

---

# API of HTTP Protocol Specification

44638 da...  
DHHUA HTTP API V3.05 for Whisper House Kft.

**V3.05**

2021-12-22

# Cybersecurity Recommendations

## **Mandatory actions to be taken towards cybersecurity**

### **1. Change Passwords and Use Strong Passwords:**

The number one reason systems get "hacked" is due to having weak or default passwords. It is recommended to change default passwords immediately and choose a strong password whenever possible. A strong password should be made up of at least 8 characters and a combination of special characters, numbers, and upper and lower case letters.

### **2. Update Firmware**

As is standard procedure in the tech-industry, we recommend keeping NVR, DVR, and IP camera firmware up-to-date to ensure the system is current with the latest security patches and fixes.

## **"Nice to have" recommendations to improve your network security**

### **1. Change Passwords Regularly**

Regularly change the credentials to your devices to help ensure that only authorized users are able to access the system.

### **2. Change Default HTTP and TCP Ports:**

- Change default HTTP and TCP ports for systems. These are the two ports used to communicate and to view video feeds remotely.
- These ports can be changed to any set of numbers between 1025-65535. Avoiding the default ports reduces the risk of outsiders being able to guess which ports you are using.

### **3. Enable HTTPS/SSL:**

Set up an SSL Certificate to enable HTTPS. This will encrypt all communication between your devices and recorder.

### **4. Enable IP Filter:**

Enabling your IP filter will prevent everyone, except those with specified IP addresses, from accessing the system.

### **5. Change ONVIF Password:**

On older IP Camera firmware, the ONVIF password does not change when you change the system's credentials. You will need to either update the camera's firmware to the latest revision or manually change the ONVIF password.

### **6. Forward Only Ports You Need:**

- Only forward the HTTP and TCP ports that you need to use. Do not forward a huge range of numbers to the device. Do not DMZ the device's IP address.
- You do not need to forward any ports for individual cameras if they are all connected to a recorder on site; just the NVR is needed.

### **7. Limit Features of Guest Accounts:**

If your system is set up for multiple users, ensure that each user only has rights to features and functions they need to use to perform their job.

### **8. UPnP:**

- UPnP will automatically try to forward ports in your router or modem. Normally this would be a good thing. However, if your system automatically forwards the ports and you leave the credentials defaulted, you may end up with unwanted visitors.
- If you manually forwarded the HTTP and TCP ports in your router/modem, this feature should be turned off regardless. Disabling UPnP is recommended when the function is not used in real applications.

#### **9. SNMP:**

Disable SNMP if you are not using it. If you are using SNMP, you should do so only temporarily, for tracing and testing purposes only.

#### **10. Multicast:**

Multicast is used to share video streams between two recorders. Currently there are no known issues involving Multicast, but if you are not using this feature, deactivation can enhance your network security.

#### **11. Check the Log:**

If you suspect that someone has gained unauthorized access to your system, you can check the system log. The system log will show you which IP addresses were used to login to your system and what was accessed.

#### **12. Physically Lock Down the Device:**

Ideally, you want to prevent any unauthorized physical access to your system. The best way to achieve this is to install the recorder in a lockbox, locking server rack, or in a room that is behind a lock and key.

#### **13. Connect IP Cameras to the PoE Ports on the Back of an NVR:**

Cameras connected to the PoE ports on the back of an NVR are isolated from the outside world and cannot be accessed directly.

#### **14. Isolate NVR and IP Camera Network**

The network your NVR and IP camera resides on should not be the same network as your public computer network. This will prevent any visitors or unwanted guests from getting access to the same network the security system needs in order to function properly.

## Purpose






Welcome to use API of HTTP protocol specification.

## Reader

- API software development engineers
- Project managers
- Product managers

## Safety Instructions

The following categorized signal words with defined meaning might appear in the Manual.

Signal Words	Meaning
 <b>DANGER</b>	Indicates a high potential hazard which, if not avoided, will result in death or serious injury.
 <b>WARNING</b>	Indicates a medium or low potential hazard which, if not avoided, could result in slight or moderate injury.
 <b>CAUTION</b>	Indicates a potential risk which, if not avoided, could result in property damage, data loss, lower performance, or unpredictable result.
 <b>TIPS</b>	Provides methods to help you solve a problem or save you time.
 <b>NOTE</b>	Provides additional information as the emphasis and supplement to the text.

## Revision History

No.	Version	Revision Content	Release Time
1	V2.40	Based on the old version.	August 8, 2018
2	V2.60	Add find media file with TrafficCar info.	September 12, 2018
3	V2.61	Add the "16 AI APIs " chapter.	October 24, 2018
4	V2.62	Add find media files with NonMotor info	November 06, 2018
5	V2.63	Add close door and access control custom password	November 27, 2018
6	V2.64	Add access control and video talk settings.	December 12, 2018
7	V2.65	Add intelligent caps and resource usage info	January 17, 2019

No.	Version	Revision Content	Release Time
8	V2.66	Add people heat map, realtime trace, history trace info	February 13, 2019
9	V2.67	Add subscribe heat map raw data	April 18, 2019
10	V2.68	Add get traffic snap event info record	May 15, 2019
11	V2.69	Add export traffic blacklist/redlist record	May 20, 2019
12	V2.70	Add privacy masking	May 29, 2019
13	V2.71	Add get the max and min temperature values	June 3, 2019
14	V2.72	Add smart motion detection config	June 27, 2019
15	V2.73	Add ecrypted download media file	July 16, 2019
16	V2.74	Adjust Camera Image, Exposure, Backlight, White Balance, Day-Night, Zoom and Focus, Lighting, Video in Options config.	July 25, 2019
17	V2.75	Add adjust angle and depth field	August 13, 2019
18	V2.76	Add subscribe vehicles distribution data.	September 10, 2019
19	V2.77	Add Seek Find Logs	October 18, 2019
20	V2.78	Add face recognition event handler config	October 30, 2019
21	V2.79	Add RemoveEx Traffic BlackList/RedList record .	December 9, 2019
22	V2.80	Add Get GPS Status	December 12, 2019
23	V2.81	Add Get Last Event Info	February 28, 2020
24	V2.82	Add Radar Adaptor	March 23, 2020
25	V2.83	Add Radar Adaptor getCapsEx Delete Radar Adaptor getCaps	May 20, 2020
26	V2.84	Add AccessControl getLockStatus	June 12, 2020
27	V2.85	Add GlobalDeviceParam SupportGlobalDeviceParam	July 2, 2020
28	V2.86	Add ObjectPlacementDetection ObjectRemovalDetection	November 5,2020
29	V2.87	Add Radar Adaptor addRadarLinkSD, delRadarLinkSD Add VideoInAnalyse getTemplateRule	November 6,2020
30	V2.88	Add LensManager, resetAngle	November 26,2020
31	V2.89	Add config to Radar Adaptor	December 3, 2020
32	V2.90	Add Get Heat Map Info	January 7, 2021
33	V2.91	Add export Encrypted Log	January 12, 2021
34	V2.92	Add controller related	February 23, 2021
35	V2.93	Add config to objectdetect DirectionStats	March 1,2021
36	V2.94	Add Camera to a Specific NVR Channel	March 11, 2021
37	V2.95	Add add and remove configure	March 26, 2021
38	V2.96	Add Add WorkSuitCompareServer	April 22,2021

No.	Version	Revision Content	Release Time
39	V2.97	Add VideoAnalyseRule <i>head</i> . Type add ObjectPlacement and ObjectRemoval	April 25,2021
40	V2.98	Modify WorkSuitCompareServer	May 11,2021
41	V2.99	Modify WorkSuitCompareServer doFind	May 11,2021
42	V3.00	Add trafficFlowStat stopFind, trafficFlowStat doFind and trafficFlowStat startFind	May 27,2021
43	V3.01	Add configManager setConfig IntelliSchemeTour.Enable	June 3,2021
44	V3.02	Add configManager getConfig IntelliSchemeTour	June 11,2021
45	V3.03	Add method cancel upgrade	June 21, 2021
46	V3.04	doFindHistoryByPic Add Time	Sept 29, 2021
47	V3.05	Add Cloud Upgrade, add find smd data	December 22, 2021

## Privacy Protection Notice

As the device user or data controller, you might collect personal data of other such as face, fingerprints, car plate number, Email address, phone number, GPS and so on. You need to be in compliance with the local privacy protection laws and regulations to protect the legitimate rights and interests of other people by implementing measures include but not limited to: providing clear and visible identification to inform data subject the existence of surveillance area and providing related contact.

## About the Manual

- The Manual is for reference only. If there is inconsistency between the Manual and the actual product, the actual product shall prevail.
- We are not liable for any loss caused by the operations that do not comply with the Manual.
- The Manual would be updated according to the latest laws and regulations of related regions. For detailed information, see the paper manual, CD-ROM, QR code or our official website. If there is inconsistency between paper manual and the electronic version, the electronic version shall prevail.
- All the designs and software are subject to change without prior written notice. The product updates might cause some differences between the actual product and the Manual. Please contact the customer service for the latest program and supplementary documentation.
- There still might be deviation in technical data, functions and operations description, or errors in print. If there is any doubt or dispute, please refer to our final explanation.
- Upgrade the reader software or try other mainstream reader software if the Manual (in PDF format) cannot be opened.
- All trademarks, registered trademarks and the company names in the Manual are the properties of their respective owners.
- Please visit our website, contact the supplier or customer service if there is any problem occurred when using the device.
- If there is any uncertainty or controversy, please refer to our final explanation.

# Table of Contents

<b>Cybersecurity Recommendations</b> .....	<b>I</b>
<b>Foreword</b> .....	<b>III</b>
<b>1 Overview</b> .....	<b>1</b>
<b>2 References</b> .....	<b>2</b>
<b>3 Definitions</b> .....	<b>3</b>
3.1 Abbreviations.....	3
3.2 Syntax Convention .....	3
3.3 API Request Format.....	3
3.4 Server Response Format .....	4
3.5 Authentication.....	5
3.6 Conventions .....	6
3.6.1 Channels.....	6
<b>4 General APIs</b> .....	<b>7</b>
4.1 APIs of RTSP .....	7
4.1.1 Get Real-Time Stream .....	7
4.1.2 Get Playback Stream .....	7
4.1.3 Get File Stream .....	8
4.1.4 Get MJPG Stream.....	8
4.1.5 Configuring Local Playback Status.....	9
4.1.6 Acquiring Local Playback Status .....	9
4.2 Common.....	10
4.2.1 Get and Set Configure .....	10
4.2.2 Add and Remove Configure .....	11
4.3 Audio .....	12
4.3.1 Get Audio Input Channel Numbers.....	12
4.3.2 Get Audio Output Channel Numbers.....	12
4.3.3 Post Audio Stream .....	13
4.3.4 Get Audio Stream.....	14
4.4 Snapshot.....	15
4.4.1 Snap.....	15
4.4.2 Get a Snapshot .....	17
4.4.3 Subscribe to Snapshot.....	17
4.5 Video Attributes .....	19
4.5.1 Get Max Extra Stream Numbers .....	19
4.5.2 Get Encode Capability .....	20
4.5.3 Get Encode Config Capability .....	20
4.5.4 Encode of Media .....	23
4.5.5 Encode of Region Interested.....	28
4.5.6 Channel Title .....	29
4.5.7 Get Video Input Channels Device Supported.....	30
4.5.8 Get Video Output Channels Device Supported .....	30

4.5.9	Get Max Remote Input Channels	30
4.5.10	Video Standard	31
4.5.11	Video Widget	31
4.5.12	Get Video Input Capability	34
4.5.13	Get Coordinates of Current Window	37
4.5.14	Set Coordinates of Current Window	38
4.5.15	Video Out	38
4.5.16	Smart Code	39
4.5.17	Get Decoder Caps	40
4.5.18	[Config] PrivacyMasking	41
4.5.19	Get Privacy Masking	43
4.5.20	Set Privacy Masking	44
4.5.21	Goto Privacy Masking	45
4.5.22	Delete Privacy Masking	45
4.5.23	Clear Privacy Masking	46
4.5.24	Get Privacy Masking Rect	46
4.5.25	[Config] SmartMotionDetect	46
4.5.26	[Config] LAEConfig	47
4.5.27	Enable/Disable all privacy masking covers	49
4.5.28	Get enable/disable state of all privacy masking covers	49
4.6	System	50
4.6.1	General	50
4.6.2	Get Current Time	51
4.6.3	Set Current Time	51
4.6.4	Locales	51
4.6.5	Get Language Capability	53
4.6.6	Language	53
4.6.7	Get Client Access Filter	54
4.6.8	Auto Maintain	55
4.6.9	Holiday Management	56
4.6.10	Get Device Type	57
4.6.11	Get Hardware Version	57
4.6.12	Get Serial Number of Device	57
4.6.13	Get Machine Name	58
4.6.14	Get System Information1	58
4.6.15	Get System Information2	58
4.6.16	Get Vendor Information	59
4.6.17	Get Software Information	59
4.6.18	Get Version of Onvif	59
4.6.19	Get Version of HTTP API	60
4.6.20	Get Device Class	60
4.6.21	Onvif Service Authorization	60
4.6.22	Backup of Config	61
4.6.23	Restore the Config	61
4.6.24	Restore except the Config	62
4.6.25	Reboot	62
4.6.26	Shutdown	62

4.6.27 FactoryReset.....	62
4.6.28 Get Tracing Code of Device .....	63
4.6.29 Add Camera to Specified Channel .....	63
4.6.30 Illuminator.....	64
4.6.31 Volume Control.....	65
4.6.32 Acquiring All Available Resources .....	65
4.6.33 Subscribing for Device Online/Offline Status.....	73
4.7 Network .....	75
4.7.1 Get Network Interfaces .....	75
4.7.2 Network Basic Config.....	75
4.7.3 PPPoE .....	77
4.7.4 DDNS .....	78
4.7.5 Email .....	80
4.7.6 Wlan.....	81
4.7.7 Scan Wlan Devices .....	82
4.7.8 UPnP.....	83
4.7.9 Get UPnP Status.....	84
4.7.10 NTP.....	84
4.7.11 RTSP.....	85
4.7.12 Alarm Server .....	86
4.7.13 Device Network Packet Capture .....	87
4.8 Motion Detection .....	90
4.8.1 Motion Detection Settings .....	90
4.9 Event.....	96
4.9.1 Event Handler .....	96
4.9.2 Alarm Event.....	99
4.9.3 Alarm Out .....	100
4.9.4 Get Alarm Input Channels .....	101
4.9.5 Get Alarm Output Channels .....	101
4.9.6 Get States of Alarm Input Channels .....	102
4.9.7 Get States of Alarm Output Channels .....	102
4.9.8 Video Blind Event.....	102
4.9.9 Video Loss Event .....	103
4.9.10 Login Failure Event .....	104
4.9.11 Storage Not Exist Event .....	105
4.9.12 Storage Access Failure Event .....	106
4.9.13 Storage Low Space Event.....	106
4.9.14 Net Abort Event .....	107
4.9.15 IP Conflict Event.....	108
4.9.16 Get Channels Event Happened.....	108
4.9.17 Subscribe to Event Message.....	109
4.9.18 Get Capability of Event Management.....	112
4.9.19 Net Alarm Event .....	113
4.9.20 Set Net Alarm State.....	114
4.9.21 GetSupportedEvents .....	114
4.10 PTZ .....	115
4.10.1 PTZ Config.....	115

4.10.2	Get PTZ Protocol List.....	116
4.10.3	Get PTZ Capability of Current Protocol.....	116
4.10.4	Get PTZ Status .....	118
4.10.5	PTZ Control.....	119
4.10.6	Preset.....	122
4.10.7	Tour .....	124
4.10.8	Scan.....	126
4.10.9	Pattern.....	127
4.10.10	Pan.....	128
4.10.11	PTZ Auto Movement.....	129
4.10.12	PTZ Restart.....	130
4.10.13	PTZ Reset.....	130
4.10.14	OSD Menu .....	131
4.11	Record.....	132
4.11.1	Get Capability of Recording.....	132
4.11.2	Record Config .....	133
4.11.3	Record Mode.....	134
4.11.4	Media Global.....	135
4.11.5	Find Media Files.....	136
4.11.6	Find media files with FaceDetection info .....	138
4.11.7	Find media files with FaceRecognition info .....	141
4.11.8	Find media files with HumanTrait info.....	145
4.11.9	Find media files with TrafficCar info.....	148
4.11.10	Find media files with IVS info .....	151
4.11.11	Find media files with NonMotor info.....	153
4.11.12	Download Media File with the File Name .....	155
4.11.13	Download Media File between Times .....	156
4.11.14	Encrypted Download Media File with the File Name .....	156
4.12	User management.....	157
4.12.1	Get Information of a Particular User .....	157
4.12.2	Get Information of All Users .....	157
4.12.3	Get Information of All Active Users .....	158
4.12.4	Get Information of a Particular Group .....	158
4.12.5	Get Information of All Groups.....	158
4.12.6	Add a New User .....	159
4.12.7	Delete a User .....	159
4.12.8	Modify User Information .....	160
4.12.9	Modify User's Password.....	160
4.12.10	Manager Modify Common User's Password .....	161
4.13	Log .....	161
4.13.1	Find Logs .....	161
4.13.2	Clear All the Logs.....	163
4.13.3	Backup Logs .....	163
4.13.4	Seek Find Logs .....	164
4.13.5	Export Encrypted Log.....	164
4.14	Upgrader .....	165
4.14.1	Strat to Upgrade.....	165

4.14.2	Get Upgrade State .....	165
4.14.3	Set upgrader url .....	166
4.14.4	Cancel Upgrade .....	166
4.14.5	Checking Cloud Update Version .....	167
4.14.6	Performing Online Update.....	169
4.14.7	Canceling Online Update .....	170
4.15	Wiper.....	171
4.15.1	Move Continuously.....	171
4.15.2	Stop Move .....	171
4.15.3	Move Once.....	171
4.16	Security .....	172
4.16.1	Set User Login Authentication Policy .....	172
<b>5</b>	<b>Camera APIs .....</b>	<b>173</b>
5.1	Image .....	173
5.1.1	Brightness, Contrast and Saturation .....	173
5.1.2	Sharpness.....	174
5.1.3	Flip, Mirror and Rotate90 .....	175
5.2	Exposure .....	176
5.2.1	Exposure Config.....	176
5.3	Backlight.....	178
5.3.1	Backlight Config .....	178
5.4	White Balance .....	179
5.4.1	White Balance Config.....	179
5.5	Day-Night .....	180
5.5.1	Day-Night Config.....	180
5.6	Zoom and Focus .....	181
5.6.1	Adjust Focus.....	181
5.6.2	Adjust Focus Continuously .....	182
5.6.3	Auto Focus .....	182
5.6.4	Get Focus Status .....	182
5.6.5	Zoom Config.....	183
5.6.6	Focus Config .....	184
5.7	Lighting.....	185
5.7.1	Lighting Config .....	185
5.8	Video in Options .....	187
5.8.1	Video in Options Config.....	187
<b>6</b>	<b>Storage APIs .....</b>	<b>196</b>
6.1	Storage Devices .....	196
6.1.1	Get Hard Disk Information.....	196
6.1.2	Get All the Names of Storage Devices .....	196
6.1.3	Get Storage Device Information .....	196
6.1.4	Get Storage Capability .....	197
6.1.5	Format Camera SD-Card .....	197
6.2	NAS.....	198
6.2.1	NAS Information.....	198
6.3	Storage Point .....	199
6.3.1	Record Storage Point.....	199

6.3.2 Storage Group.....	200
6.4 SDEncrypt .....	201
6.4.1 Encrypt SD Card .....	201
6.4.2 Decrypt SD Card .....	202
6.4.3 Clear SD Card Password .....	202
6.4.4 Modify SD Card Password .....	203
6.4.5 Get SD Card Operate Error Policy .....	203
6.4.6 Storage Health Alarm Settings .....	203
<b>7 Display APIs .....</b>	<b>205</b>
7.1 GUI.....	205
7.1.1 GUISet .....	205
7.2 Split Screen.....	206
7.2.1 Split Screen Mode .....	206
7.3 Monitor Tour .....	207
7.3.1 Monitor Tour .....	207
7.3.2 Enable Tour .....	208
7.3.3 Monitor Collection .....	208
<b>8 Video Analyse APIs .....</b>	<b>210</b>
8.1 Video Analyse.....	210
8.1.1 Get Video Analyse Capability .....	210
8.1.2 Video Analyse Global .....	211
8.1.3 Video Analyse Rule .....	212
8.1.4 Get Last Event Info .....	217
8.1.5 [Config] GlobalDeviceParam .....	217
8.1.6 getTemplateRule .....	218
8.1.7 [Config] IntelliSchemeTourEnableSetting .....	219
8.1.8 Getting Intelligent Tour Plan .....	220
8.1.9 Exporting Intelligent Diagnosis, Allowlist, and Blocklist Information .....	225
8.1.10 Importing Intelligent O&M, Allowlist, and Blocklist Information .....	226
8.2 Number of People .....	227
8.2.1 Video Widget Number Status .....	227
8.2.2 Get Heat Map Information .....	228
8.2.3 Get People Heat Map Information.....	229
8.2.4 Subscribe People Realtime Trace Information .....	231
8.2.5 Get People Histry Trace Information .....	232
8.2.6 Subscribe Heat Map Raw Data .....	233
8.2.7 Clear statistics in time section .....	234
8.3 FishEye .....	235
8.3.1 Get FishEye Capability.....	235
8.3.2 FishEye Setting .....	235
8.4 CrowdDistriMap.....	236
8.4.1 Get Channel Caps.....	236
8.4.2 Subscribe to Realtime Crowd Stat .....	237
8.4.3 Get Current Crowd Stat.....	238
8.5 Intelligent.....	239
8.5.1 Get Intelligent Caps.....	239
8.5.2 Subscribe Resource Usage Info.....	240

8.6 WorkSuitCompareServer .....	242
8.6.1 Creating Compliance Library .....	242
8.6.2 Deleting Compliance Library .....	242
8.6.3 Searching for Compliance Library .....	242
8.6.4 Getting Compliance Library Arming Information of Channels .....	243
8.6.5 Modifying Compliance Group Information .....	244
8.6.6 Arming Compliance Library by Video Channel .....	244
8.6.7 Searching for Workwear Information in Compliance Library .....	245
8.6.8 Getting Search Result .....	245
8.6.9 Stopping Search.....	246
8.6.10 Deleting Workwear Information .....	246
8.6.11 Re-extracting Features by Workwear .....	247
8.6.12 Stopping Re-extracting Workwear Features.....	247
8.7 SMD Query .....	247
8.7.1 Start SMD Query.....	247
8.7.2 Searching for SMD Data .....	249
8.7.3 End SMD Search .....	250
<b>9 Intelligent Traffic APIs .....</b>	<b>251</b>
9.1 Traffic Snap .....	251
9.1.1 Get the Specific Parking Space Status .....	251
9.1.2 Open Strobe.....	251
9.1.3 Open/Close Unlicensed Vehicle Detection .....	252
9.1.4 Snap.....	252
9.2 Traffic Parking .....	253
9.2.1 Get All Status of Parking Spaces .....	253
9.2.2 Parking Space Light State.....	253
9.2.3 Set Order State .....	255
9.2.4 Set Light State.....	255
9.2.5 [Config] Parking Space Access Filter Setting .....	256
9.2.6 Set OverLine State.....	257
9.3 Traffic Record Import Export.....	257
9.3.1 Traffic BlackList / RedList.....	257
9.3.2 Traffic Flow.....	259
9.3.3 Traffic Snap Event Info .....	260
9.4 Vehicles Distribution .....	261
9.4.1 Subscribe Vehicles Distribution Data .....	261
<b>10 Thermography and Radiometry APIs .....</b>	<b>264</b>
10.1 Thermography Manager.....	264
10.1.1 Get Capability of Thermography .....	264
10.1.2 Thermography Options.....	265
10.1.3 Get ExternSystem Information .....	267
10.1.4 Get Information of Preset Mode .....	267
10.1.5 Get Optimized Region Information .....	268
10.1.6 Enable Shutter .....	268
10.1.7 Fix Focus.....	269
10.1.8 Do Flat Field Correction .....	269
10.2 Radiometry.....	269

10.2.1 Get Capability of Radiometry .....	269
10.2.2 Heat Image Thermometry .....	270
10.2.3 Thermometry Rule.....	272
10.2.4 Heat Image Temper Event.....	274
10.2.5 Get Temperature of Particular Point .....	275
10.2.6 Get Temperature of Particular Condition .....	276
10.2.7 Query Temperature Information .....	276
10.2.8 Subscribe to Temperature Information .....	278
10.2.9 Subscribe to Radiometry Data .....	279
10.2.10 To Fetch Radiometry Data.....	280
10.2.11 Get FireWarning Config.....	280
10.2.12 Set FireWarning Config .....	281
10.2.13 Get FireWarningMode Config.....	282
10.2.14 Set FireWarningMode Config .....	282
10.2.15 Get Current Hot Cold Spot .....	282
10.2.16 Heat Image Temper PreAlarm Event.....	283
10.2.17 Get Heat Map Info.....	284
10.3 TemperCustom.....	286
10.3.1 Set Environment Temperature.....	286
<b>11 Access Control APIs .....</b>	<b>287</b>
11.1 Door.....	287
11.1.1 Open Door.....	287
11.1.2 Get Door Status.....	287
11.1.3 Close Door .....	287
11.1.4 Capture Fingerprint.....	288
11.1.5 Capture face picture .....	288
11.2 Access Control.....	289
11.2.1 Add Access Control Custom Password .....	289
11.2.2 Modify Access Control Custom Password .....	290
11.2.3 Delete Access Control Custom Password .....	290
11.2.4 Find Access Control Custom Password.....	291
11.2.5 Get the Total Number of Records of Access Control Custom Password .....	294
11.2.6 Get Access Control Caps .....	294
11.2.7 Get Lock Status.....	295
11.2.8 [Config] Access Control General Setting .....	296
11.2.9 [Config] Access Control Setting .....	298
11.2.10 [Config] Wiegand Setting.....	302
11.2.11 [Config] Access Time Schedule Setting .....	303
11.2.12 [Config] Special Day Group Setting .....	304
11.2.13 [Config] Special Days Schedule Setting .....	305
11.2.14 [Config] MeasureTemperature Setting .....	307
11.2.15 [Config] CitizenPictureCompare Setting .....	310
11.2.16 Sending User Information.....	311
11.2.17 Updating User Information.....	313
11.2.18 Deleting All User Information .....	315
11.2.19 Deleting Information of Multiple Users.....	315
11.2.20 Searching for Information of Multiple Users.....	316

11.2.21	Sending Information of Multiple Card Numbers .....	318
11.2.22	Updating Information of Multiple Card Numbers .....	319
11.2.23	Removing All Card Number Information .....	320
11.2.24	Removing Information of Multiple Card Numbers .....	320
11.2.25	Searching for Information of Multiple Card Numbers .....	321
11.2.26	Sending Face Information .....	322
11.2.27	Updating Face Information .....	323
11.2.28	Deleting All Face Information.....	323
11.2.29	Deleting Multiple Face Information .....	324
11.2.30	Searching for Multiple Face Information .....	324
11.2.31	Sending Fingerprint Information .....	325
11.2.32	Updating Fingerprint Information .....	326
11.2.33	Deleting All Fingerprint Information.....	327
11.2.34	Deleting Fingerprint Information .....	328
11.2.35	Searching for Fingerprint Information .....	328
11.2.36	Starting Searching for User Related Information .....	329
11.2.37	Getting User Related Information .....	331
11.2.38	Stopping Searching for User Related Information .....	332
11.2.39	Starting Searching for Card Number Related Information .....	332
11.2.40	Getting Card Number Related Information .....	333
11.2.41	Stopping Searching for Card Number Related Information .....	334
11.2.42	Starting Searching for Face Related Information.....	335
11.2.43	Getting Face Related Information.....	336
11.2.44	Stopping Searching for Face Related Information.....	336
11.2.45	Get access control capability.....	337
11.2.46	Add SubController .....	337
11.2.47	Modify SubController .....	338
11.2.48	Remove SubController .....	338
11.2.49	Get SubController Info.....	339
11.2.50	Get SubController States.....	340
11.2.51	Set RepeatEnter Route .....	340
11.2.52	Get RepeatEnter Route .....	341
11.2.53	Set ABLock Route .....	341
11.2.54	Get ABLock Route .....	342
11.2.55	Get Log Status .....	343
11.2.56	Sync Offline Log .....	343
11.2.57	Sync SubController Time.....	344
<b>12</b>	<b>Intelligent Building APIs .....</b>	<b>345</b>
12.1	Video Talk.....	345
12.1.1	Subscribe Video Talk Status.....	345
12.1.2	Unsubscribe Video Talk Status.....	345
12.1.3	Invite Server on Video Talk.....	346
12.1.4	Cancel the Video Talk.....	346
12.1.5	Answer the Invitation .....	346
12.1.6	Refuse to Answer the Video Talk Invitation .....	347
12.1.7	Hang Up .....	347
12.2	Video Talk Log.....	347

12.2.1 Query Video Talk Log .....	347
12.3 Announcement Record .....	348
12.3.1 Insert Record .....	348
12.4 Alarm Record .....	349
12.4.1 Query Alarm Record .....	349
12.4.2 Query Access Control Alarm Record .....	349
12.5 SIP .....	350
12.5.1 Get SIP Configuration .....	350
12.5.2 Modifying SIP Configuration .....	352
12.6 Registrar .....	355
12.6.1 Get Registrar Configuration .....	355
12.6.2 Modifying Registrar Configuration .....	356
12.7 Room Number Database Management .....	356
12.7.1 Adding Room Number .....	356
12.7.2 Getting Records by recno .....	358
12.7.3 Updating Room Number Records .....	360
12.7.4 Deleting Records by recno .....	361
12.7.5 Clearing All Room Numbers .....	362
12.7.6 Getting Total Quantity of Room Number .....	362
12.8 ElevatorFloorCounter .....	362
1.1.1 setElevatorFloorInfo .....	362
12.8.1 getElevatorWorkInfo .....	363
12.8.2 getCaps .....	363
<b>13 DVR Custom APIs .....</b>	<b>365</b>
13.1 File Finder .....	365
13.1.1 Create a File Finder .....	365
13.1.2 Create a Motion File Finder .....	366
13.1.3 Get the File Information Found by the Finder .....	367
13.1.4 Stop the Finder .....	368
13.1.5 Get Bound Files .....	368
13.2 BandLimit .....	369
13.2.1 Get Bandwidth Limit State .....	369
13.3 Record Files Protection .....	369
13.3.1 Add Protection .....	369
13.3.2 Cancel Protection .....	370
13.3.3 Remove Protection .....	370
13.3.4 DownloadFile .....	370
13.3.5 UploadFile .....	371
13.4 Get Daylight .....	372
<b>14 Comm .....</b>	<b>373</b>
14.1 Coaxial Control IO .....	373
14.1.1 Control White Light or Speaker .....	373
14.2 Pir Alarm .....	374
14.2.1 Configure Pir Parameter .....	374
<b>15 Other APIs .....</b>	<b>378</b>
15.1 Discover Devices .....	378
15.1.1 Discover Devices on Internet .....	378

15.2 Flashlight.....	379
15.2.1 Flashlight Config .....	379
15.3 Open Platform .....	380
15.3.1 Application Start and Stop.....	380
15.3.2 Install Application .....	381
15.3.3 Update Application and License .....	381
15.3.4 Uninstall Application .....	383
15.3.5 Download Application Log .....	383
15.4 LensFunc .....	384
15.4.1 getCaps.....	384
15.4.2 adjustAngleContinuously.....	384
15.4.3 stopAdjustingAngle .....	385
15.4.4 adjustDepthField .....	385
15.4.5 adjustDepthFieldContinuously.....	385
15.4.6 getDepthFieldStatus.....	386
15.4.7 autoAdjustDepthField.....	386
15.5 positionManager.....	387
15.5.1 getStatus.....	387
15.6 Radar Adaptor .....	387
15.6.1 getCaps.....	387
15.6.2 getCapsEx.....	388
15.6.3 getStatus.....	388
15.6.4 calculateRealSize.....	388
15.6.5 attachAlarmPointInfo.....	389
15.6.6 getLinkSDState .....	390
15.6.7 manualLocate.....	391
15.6.8 startRadarCalibration .....	391
15.6.9 addRadarLinkSD.....	392
15.6.10 delRadarLinkSD.....	392
15.6.11 [Config]MapPara.....	393
15.6.12 [Config]RadarAnalyseRule.....	394
15.6.13 [Config]RadarCalibration .....	396
15.6.14 [Config]RadarGuardLine .....	397
15.6.15 [Config]RadarLink .....	398
15.6.16 [Config]RadarLinkDevice.....	398
15.6.17 [Config]RadarPara .....	398
15.6.18 [Config]RadarTrackGlobal.....	399
15.6.19 [Config]RemoteSDLink.....	400
15.7 GpsControl .....	400
15.7.1 getCaps.....	400
15.7.2 GPS config.....	400
15.8 LensManager .....	401
15.8.1 resetAngle .....	401
15.9 Advertisement.....	402
15.9.1 Getting Ad Placement Capability.....	402
15.9.2 Acquiring Advertisement Resource List.....	406
15.9.3 Deleting Advertisement Files .....	407

15.9.4 Sending Advertisement Resources .....	408
15.9.5 Changing Stay Time of Advertisement Image .....	411
15.9.6 Acquiring Video and Image Files Played by the Device .....	411
<b>16 AI APIs .....</b>	<b>415</b>
16.1 Video Analyse APIs .....	415
16.1.1 People Counting.....	415
16.1.2 FaceRecognitionServer.....	423
16.1.3 Video Analyse Event .....	452
16.2 Intelligent Traffic APIs.....	460
16.2.1 Traffic Record.....	460
16.2.2 Intelligent Traffic Event.....	463
16.2.3 Traffic Flow.....	468
16.3 Access Control APIs.....	473
16.3.1 Access User .....	473
16.3.2 Access control.....	487

This document specifies the HTTP-based application programming interface of video products.

The HTTP-based interface provides the functionality for requesting snapshot and media stream, controlling camera functions (for example, PTZ and focus), and getting and setting internal parameter values.

The video products serve as a server. The client sends requests to server, and then server handles requests and returns resources accordingly.

DAHUA HTTP API V3.05 for Whisper House Kft.

- [1] RFC 2326 Real Time Streaming Protocol (RTSP)
- [2] RFC 2616 Hypertext Transfer Protocol-HTTP/1.1
- [3] RFC 3986 Uniform Resource Identifiers (URI) Generic Syntax
- [4] RFC 7616 HTTP Digest Access Authentication
- [5] RFC 8259 The JavaScript Object Notation (JSON) Data Interchange Format

DAHUA HTTP API V3.05 for Whisper House Kft.

## 3.1 Abbreviations

The following abbreviations are used throughout this document.

API	Application Programming Interface. In the document, it especially presents application programming interface of video products.
-----	---

## 3.2 Syntax Convention

- In URL syntax and in descriptions of API parameters, text in italic within angle brackets denotes content that should be replaced with either a value or a string. When replacing the text string, the angle brackets must also be replaced. For example, *<server>* in URL syntax is replaced with the IP number of server, e.g., 192.168.1.108.
- String shown in bold face denotes a brief explanatory note of the string close to it.
- Name-value pair in square brackets denotes content that is optional. For example, "http ://<server>/cgi-bin/snapshot.cgi[?channel=1]" can be like this "http ://<server>/cgi-bin/snapshot.cgi".
- The API syntax must follow the standard of URI. (RFC 3986: Uniform Resource Identifiers (URI) Generic Syntax); that is, spaces and other reserved characters (e.g, ":", "/", "?", "@", ";", "=", "+", ",", "\$", "&") within a name-value pair should be replaced with %< ASCII hex>. For example, the blank should be replaced with %20.
- To describe the range of a variable, we use some symbols such as "[]" and "{}". For example, "[0-100]" denotes an integer not less than 0 and not larger than 100. "{0, 1, 2, 3}" denotes the valid value of an integer among 0, 1, 2 and 3.
- "[]" following a string denotes an array. The index is an integer and starts from 0. For example, "Snap[*channel*]" may be "Snap[0]" "Snap[0]" etc.
- The variable may be different types: string, integer, bool or float. Integer is 32 bits. The range of bool is "true" and "false".
- "R/O" in parameters means this parameter is required or not, "R" means required, "O" means optional.

## 3.3 API Request Format

This section defines the syntax and semantics for APIs.

```
<protocol>://<server><abs_path>[?query]
```

**protocol**: URL scheme for the particular request. The http and https protocols are both supported in this specification. So "http", as most of the APIs' default protocol except several RTSP APIs, can be replaced by "https".

**server**: Server could be "**hostname[:port]**". The **hostname** can be IP address or the fully qualified domain name of an IP device. The **port** is the port number of **server** listening for TCP connections. If the port is not given, the default port is assumed. For HTTP, the default port is 80. For HTTPS, the default port is 443.

**abs\_path:** The Request-URI for the resources is abs\_path. The abs\_path in this specification is most often of the form "/cgi-bin/\*.cgi".

**query:** The query field is a string of information to be interpreted by the resource. It consists of resource-related parameters. And it must be listed in name-value pair syntax (p1=v1&p2=v2&...&pn=vn).

For example: <http://192.168.1.108/cgi-bin/snapshot.cgi?channel=1>

### 3.4 Server Response Format

The server uses the standard HTTP status codes.

Return:

```
HTTP/1.1 <HTTP code> <HTTP text>
```

With the following HTTP code and meanings

Table 3-1

HTTP code	HTTP text	Description
200	OK	The request has succeeded. The requested resource will be returned in the HTTP text.
400	Bad Request	The request had bad syntax or was inherently impossible to be satisfied.
401	Unauthorized	The request requires user authentication or the authorization has been refused.
403	Forbidden	The user does not have the right to access the service.
404	Not Found	The server has not found anything matching the request.
500	Internal Server Error	The server encountered an unexpected condition that prevented it from fulfilling the request.
501	Not Implemented	The server has not implemented the service.

If the HTTP code is 200, means the API execute success, and the response data in http body ( maybe multipart ) can be a multiline key=value data, or a json object, or just a line with a word "OK".

**Example:** success response with multiline key=value

```
HTTP/1.1 200 OK
Server: xxx
Content-Length: <length>

status.Focus=0.5
status.Zoom=0.5
...
```

**Example:** success response with a word "OK"

```
HTTP/1.1 200 OK
Server: xxx
Content-Length: <length>
```

OK

If the HTTP code is not 200, means the API execute failed, and the response data in http body maybe empty, or just two line, first line is a word "Error" to indicate error happened, the second line contain error detail.

**Example:** request does not fit with syntax.

```
HTTP/1.1 404 Not Found
Server: xxxx
```

**Example:** Request spells wrong.

```
HTTP/1.1 400 Bad Request
Server: xxx
Content-Length: <length>

Error
Bad Request!
```

**Example:** If the request fits with syntax but an error occurs while the server handles it, the response would like this:

```
HTTP/1.1 500 Internal Server Error
Server: xxx
Content-Length: <length>

Error
Internal Server Error!
```

## 3.5 Authentication

Video products support digest authentication, see RFC 7616 for detail. If the http request sent by client does not provide valid "Authorization" header information, video products would return HTTP status code 401 and some information for authentication, then client should calculate authentication information according to RFC 7616, and sent request again with authentication information using "Authorization" header. Video products return the required resource only if authorization information correct.

For example:

1. When the client does not providing digest authentication or the client calculates digest authentication with expired "nonce" data, video product's response is :

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Digest realm="DH_00408CA5EA04", qop="auth"
nonce="000562fdY631973ef04f77a3ede7c1832ff48720ef95ad",
opaque="5ccc069c403ebaf9f0171e9517f40e41", stale=TRUE
```

The "stale=TRUE" indicates that the "nonce" data in the request is expired and the client should use the new "nonce" data to calculate the digest authorization and request again.

2. Then the client generate "nc" and "cnonce", calculates the digest authorization using "username" "password" "HTTP method" "URI" "nc" "cnonce" "realm" "nonce" , according to RFC 7616, and then sends it to video product :

```
GET /cgi-bin/magicBox.cgi?action=getLanguageCaps HTTP/1.1
Authorization: Digest username="admin", realm="DH_00408CA5EA04", qop="auth",
    nonce="000562fdY631973ef04f77a3ede7c1832ff48720ef95ad",
    opaque="5ccc069c403ebaf9f0171e9517f40e41"
nc=00000001, cnonce="0a4f113b",
uri="/cgi-bin/magicBox.cgi?action=getLanguageCaps",
response="65002de02df697e946b750590b44f8bf"
```

If the digest authorization is correct, the video product will reply HTTP status code 200 with the response data. If digest authorization is not correct, the video product will reply HTTP status code 403 . If the "nonce" data used to calculates digest authentication is expired, the video product will reply HTTP status code 401 with stale=TRUE and new "nonce" data, then the client should use the new "nonce" data to calculate the digest authorization and request again .

## 3.6 Conventions

### 3.6.1 Channels

For requests containing the param 'channel' like :

http://<server>/cgi-bin/mjpg/video.cgi?channel=<ChannelNo>

ChannelNo: integer, must starts from 1. Default is 1 if not present. But for response, the 'ChannelNo' should start from 0. In other words, the request channel 1 equals response channel 0.

The APIs specified in this section are supported by all video products.

## 4.1 APIs of RTSP

### 4.1.1 Get Real-Time Stream

Get real-time media stream APIs use RTSP protocol, please refer to RFC 2326 for detail. The rtsp service default port is 554. The IP Camera supports both TCP and UDP transmission forms. It also supplies basic authentication and digest authentication ways. The authentication process is similar with "3.5 Authentication".

The rtsp url parameters format is in following table. If the stream does not exist or not enabled, it will response with error result.

<b>URL</b>	rtsp://<server>:[port]/cam/realmonitor			
<b>Method</b>	DESCRIBE, SETUP, PLAY, PAUSE, TEARDOWN, ...			
<b>URL Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	R	The video channel index which starts from 1	1
subtype	int	R	The stream type, include main stream and extra stream. The extra stream counts can be obtained in "4.5.1 Get Max Extra Stream Numbers". It can be : 0 : main stream 1 : extra stream 1 2 : extra stream 2	0
<b>URL Example</b>				
rtsp://192.168.1.108:554/cam/realmonitor?channel=1&subtype=0				

### 4.1.2 Get Playback Stream

It's similar with "4.1.1 Get Real-Time Stream". Except there are parameters "starttime" and "endtime".

<b>URL</b>	rtsp://<server>:[port]/cam/playback			
<b>Method</b>	DESCRIBE, SETUP, PLAY, PAUSE, TEARDOWN, ...			
<b>URL Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	R	The video channel index which starts from 1	1
starttime	string	R	The playback begin time	2012_09_15_12_37_05
endtime	string	O	The playback end time	2012_09_15_18_34_14
<b>URL Example</b>				
rtsp://192.168.1.108:554/cam/playback?channel=1&starttime=2012_09_15_12_37_05&endtime=2012_				

### 4.1.3 Get File Stream

It's similar with "4.1.1 Get Real-Time Stream". The filename with absolute path of file stream to get is place in rtsp url.

<b>URL</b>	rtsp://<server>:[port]/<path to filename>			
<b>Method</b>	DESCRIBE, SETUP, PLAY, PAUSE, TEARDOWN, ...			
<b>URL Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
<b>URL Example</b>				
rtsp://192.168.1.108:554//mnt/sd/2015-09-16/001/dav/20/20.32.08-20.32.28[M][0@0][0].dav				

### 4.1.4 Get MJPG Stream

Get a video stream encoded by mjpg ( motion jpeg ).

<b>Request URL</b>	http://<server>/cgi-bin/mjpg/video.cgi			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	R	The video channel index which starts from 1	1
subtype	int	R	The stream type, It can be the following value: 0-Main Stream 1-Extra Stream 1 2-Extra Stream 2	0
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/mjpg/video.cgi?channel=1&subtype=0				

#### Response Params ( multipart, binary in body )

<binary data> : JPEG image data

#### Response Example

```
HTTP/1.1 200 OK
Server: Device/1.0
Content-Type: multipart/x-mixed-replace; boundary=<boundary>
```

```
--<boundary>
```

```
Content-Type: image/jpeg
```

```
Content-Length:<image size>
```

```
<JPEG image data>
```

```
--<boundary>
```

```
Content-Type: image/jpeg
```

```
Content-Length:<image size>
```

```
<JPEG image data>
```

```
--<boundary>
```

## 4.1.5 Configuring Local Playback Status

Configure the local playback status when accessing the platform. The value of action can be Play, Pause, Stop, Faster, Slower, Back, NextFile, PreFile, NetFrame, PrevFrame, Up, Down, Left, Right, Enter, and Escape.

<b>Request URL</b>	http://<server>/cgi-bin/api/localPlayback/set			
<b>Method</b>	POST			
<b>Request Params</b> (JSON format in body)				
Name	Type	R/O	Description	Example
action	char[32]	O	Configure the status of playback, fast play and slow play. Available statuses: Play, Pause, Stop, Faster, Slower, Back, NextFile, PreFile, NetFrame, PrevFrame, Up, Down, Left, Right, Enter, and Escape.	"Play"
<b>Request Example</b>				
<pre>{   "action": "Play" }</pre>				

<b>Response Params</b> ( JSON format in body )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
<pre>{ }</pre>				

## 4.1.6 Acquiring Local Playback Status

Configure the local playback status when accessing the platform.

<b>Request URL</b>	http://<server>/cgi-bin/api/localPlayback/get			
<b>Method</b>	POST			
<b>Request Params</b> (JSON format in body)				
Name	Type	R/O	Description	Example
action	char[32]	O	Configure the status of playback, fast play, and slow play. The value can be Play (acquire playback status), and Speed (acquire the speed of fast and slow play).	"Slower"
<b>Request Example</b>				
<pre>{   "action": "Speed" }</pre>				

Response Params ( JSON format in body )				
Name	Type	R/O	Description	Example
result	char[64]	O	Returns the result of getAction. Play corresponds to three results: Stopped, Pasued, and Playing. Speed has the following situations: Faster 1×, Slower 1×, Faster 2×, Slower 2×, Faster 3×, Slower 3×, Faster 4×, Slower 4×, and Normal.	"Slower 1x"
Response Example				
<pre>{   "result": "Slower 1x" }</pre>				

## 4.2 Common

### 4.2.1 Get and Set Configure

- **Get Configure**

Get configure detail by name. Each configure name has it's specific configure detail. Please refer to configure detail APIs in later chapters.

<b>Request URL</b>	http://<server>/cgi-bin/configManager.cgi?action=getConfig			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
Name	Type	R/O	Description	Sample
name	string	R	The name of the configure	SmartEncode
Request Example				
http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=SmartEncode				

Response Params ( key=value format in body )				
Name	Type	R/O	Description	Sample
table	object	R	The configure table object.	
+<config name>	string/object/object[]	R	The configure detail of the requested configure name. Each configure name has it's specific configure detail. Please refer to configure detail apis in later chapters.	
Response Example				

```
table.SmartEncode.Enable=true
table.SmartEncode.Extra[0]=true
table.SmartEncode.Extra[1]=false
```

- **Set Configure**

Set configure detail by name. Each configure name has it's specific configure detail. Please refer to configure detail APIs in later chapters.

<b>Request URL</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Sample</b>
<config name>	string/object/object[]	R	The configure detail of the configure name. Each configure name has it's specific configure detail. Please refer to configure detail apis in later chapters.	
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&SmartEncode.Enable=true&SmartEncode.Extra[0]=true&SmartEncode.Extra[1]=false				
<b>Response Params ( OK in body )</b>				
<b>Response Example</b>				
OK				

## 4.2.2 Add and Remove Configure

- **Add configure**

The configuration name is put in the URL in the form of key-value; the specific configuration field can be obtained through getConfig. Each configure name has it's specific configure detail. Please refer to configure detail APIs in later chapters.

<b>Request URL</b>	http://<server>/cgi-bin/configManager.cgi?action=addConfig			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Sample</b>
<config name>	string/object/object[]	R	The configure detail of the configure name. Each configure name has it's specific configure detail. Please refer to configure detail apis in later chapters.	
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=addConfig&VSP_CGI.ServiceStart=true				
<b>Response Params ( OK in body )</b>				
<b>Response Example</b>				
OK				

- **Remove configure**

Specific configuration fields can be obtained through getConfig. Only configuration fields can be deleted, not configuration items. Each configure name has its specific configure detail. Please refer to configure detail APIs in later chapters.

<b>Request URL</b>	http://<server>/cgi-bin/configManager.cgi?action=removeConfig			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Sample</b>
name	string	R	configuration param name	VSP_CGI.ServiceStart
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=removeConfig&name=VSP_CGI.ServiceStart				

<b>Response Params ( OK in body )</b>				
<b>Response Example</b>				
OK				

## 4.3 Audio

### 4.3.1 Get Audio Input Channel Numbers

Get audio input channel number.

<b>Request URL</b>	http://<server>/cgi-bin/devAudioInput.cgi?action=getCollect			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/devAudioInput.cgi?action=getCollect				

<b>Response Params ( key=value format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
result	int	R	audio input channel number.	1
<b>Response Example</b>				
result=1				

### 4.3.2 Get Audio Output Channel Numbers

Get audio output channel number.

<b>Request URL</b>	http://<server>/cgi-bin/devAudioOutput.cgi?action=getCollect			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/devAudioOutput.cgi?action=getCollect				

Response Params ( key=value format in body )				
Name	Type	R/O	Description	Example
result	int	R	audio output channel number.	2
Response Example				
result=2				

### 4.3.3 Post Audio Stream

Post audio stream to device, request is very long and client continues to send audio data. If client want to stop, just close the connection.

<b>Request URL</b>	http://<server>/cgi-bin/audio.cgi?action=postAudio			
<b>Method</b>	POST			
<b>Request Params ( key=value format in URL )</b>				
Name	Type	R/O	Description	Example
httpstype	string	R	audio http transmit format, can be : singlepart: HTTP content is a continuous flow of audio packets multipart: HTTP content type is multipart/x-mixed-replace, and each audio packet ends with a boundary string	singlepart
channel	int	R	The audio channel index which starts from 1.	1
<b>Request Example ( singlepart )</b>				
POST http://192.168.1.108/cgi-bin/audio.cgi?action=postAudio&httpstype=singlepart&channel=1 HTTP/1.1 User-Agent: client/1.0 Content-Type: Audio/G.711A Content-Length: 9999999  <Audio data> <Audio data> ...				
<b>Request Example ( multipart )</b>				
POST http://192.168.1.108/cgi-bin/audio.cgi?action=postAudio&httpstype=multipart&channel=1 HTTP/1.1 User-Agent: client/1.0 Content-Type: multipart/x-mixed-replace; boundary=<boundary>  --<boundary> Content-Type: Audio/G.711A Content-Length: 800  <Audio data> --<boundary> Content-Type: Audio/G.711A Content-Length: 800				

```
<Audio data>
--<boundary>
...
```

#### Response Params ( N/A )

Name	Type	R/O	Description	Example
<b>Response Example</b>				
( N/A )				

#### Appendix A: Audio Encode Type

MIME ( Content-Type )	Description
Audio/PCM	PCM
Audio/ADPCM	ADPCM
Audio/G.711A	G.711 A Law
Audio/G.711Mu	G.711 Mu Law
Audio/G.726	G.726
Audio/G.729	G.729
Audio/MPEG2	MPEG2
Audio/AMR	AMR
Audio/AAC	AAC

### 4.3.4 Get Audio Stream

Get audio stream from device, response is very long and client continues to recv audio data. If client want to stop, just close the connection.

<b>Request URL</b>	http://<server>/cgi-bin/audio.cgi?action=getAudio			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
Name	Type	R/O	Description	Example
httpstype	string	R	audio http transmit format, can be : singlepart: HTTP content is a continuous flow of audio packets multipart: HTTP content type is multipart/x-mixed-replace, and each audio packet ends with a boundary string	singlepart
channel	int	R	The audio channel index which starts from 1.	1
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/audio.cgi?action=getAudio&httpstype=singlepart&channel=1				

#### Response Params ( binary in body )

Name	Type	R/O	Description	Example
<b>Response Example ( singlepart )</b>				
200 OK HTTP/1.1				
Server: device/1.0				
Content-Type: Audio/G.711A				

```

Content-Length: 9999999

<Audio data>
<Audio data>
...
Response Example ( multipart )
200 OK HTTP/1.1
Server: device/1.0
Content-Type: multipart/x-mixed-replace; boundary=<boundary>

--<boundary>
Content-Type: Audio/G.711A
Content-Length: 800

<Audio data>
--<boundary>
Content-Type: Audio/G.711A
Content-Length: 800

<Audio data>
--<boundary>
...

```

## 4.4 Snapshot

### 4.4.1 Snap

- Get snap config

Table 4-1

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Snap
Method	GET
Description	Get snap config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Snap
Success Return	<pre> table.Snap[0].HolidayEnable=false table.Snap[0].TimeSection[0][0]=6 00:00:00-23:59:59 table.Snap[0].TimeSection[0][1]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[0][2]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[0][3]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[0][4]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[0][5]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[1][0]=6 00:00:00-23:59:59 table.Snap[0].TimeSection[1][1]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[1][2]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[1][3]=0 00:00:00-23:59:59 </pre>

	<pre>table.Snap[0].TimeSection[1][4]=0 00:00:00-23:59:59 table.Snap[0].TimeSection[1][5]=0 00:00:00-23:59:59 ...</pre>
Comment	<p>Response format:</p> <pre>table. Snap[<b>channelNo</b>].TimeSection[<b>weekday</b>][<b>configNo</b>]=1 00:00:00-23:59:59</pre> <p><b>channelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>weekday</b>: range is [0—7] (Sunday — Saturday, Holiday).</p> <p><b>configNo</b>: the index of time section config. There are many time sections each day.</p>

- Set snap config

Table 4-2

Syntax	<pre>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;[&amp;&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;...]</pre>
Method	GET
Description	Set snap config.
Example	<pre>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;Snap[0].TimeSection[0][0]=1%2012:00:00-18:00:00</pre>
Success Return	OK
Comment	<p>In table below,</p> <p><b>ch</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>wd</b>: week day index, range is [0—6] (Sunday — Saturday).</p> <p><b>ts</b>: time section index, starts from 0.</p>

**Appendix:**

ParamName	ParamValue type	Description
Snap[ <b>ch</b> ].TimeSection[ <b>wd</b> ][ <b>ts</b> ]	string	<p>wd (week day) range is [0—7] (Sunday—Saturday, Holiday)</p> <p>ts (time section) range is [0—23], it's time section table index.</p> <p>Format: mask hh:mm:ss-hh:mm:ss</p> <p>Mask: [0—4294967295], hh: [0—24], mm: [0—59], ss: [0—59]</p> <p>Mask indicates record type by bits:</p> <p>Bit0: regular snapshot</p> <p>Bit1: motion detection snapshot</p> <p>Bit2: alarm snapshot</p> <p>Bit3: card snapshot</p> <p>Bit6- POS record</p> <p>Bit4, Bit4, Bit7~Bit31-reserved。</p>

## 4.4.2 Get a Snapshot

Table 4-3

Syntax	http://<server>/cgi-bin/ snapshot.cgi[?channel=< <b>ChannelNo</b> >]
Method	GET
Description	Get a snapshot of a video channel.
Example	To get a snapshot of video channel 1, the URL can be http://192.168.1.108/cgi-bin/snapshot.cgi or http://192.168.1.108/cgi-bin/snapshot.cgi?channel=1
Success Return	Image of jpg format.
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1, default 1 if not specified.

## 4.4.3 Subscribe to Snapshot

Table 4-4

Syntax	http://<server>/cgi-bin/snapManager.cgi?action=attachFileProc&Flags[0]=Event&Events=[< <b>eventCode</b> >,< <b>eventCode</b> >...][&channel=< <b>ChannelNo</b> >][&heartbeat=< <b>Heartbeat</b> >]
Method	GET
Description	Subscribe pictures when event of code <b>eventCode</b> happens.
Example	http://192.168.1.108/cgi-bin/snapManager.cgi?action=attachFileProc&Flags[0]=Event&Events=[VideoMotion%2CVideoLoss]&heartbeat=5

<p>Success Return</p>	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;data length&gt;  Events[0].EventBaseInfo.Code=TrafficJunction Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].Code=TrafficJunction Events[0].CountInGroup=1 Events[0].IndexInGroup=1 Events[0].Lane=1 Events[0].PTS= 42949485818.0 Events[0].TrafficCar.PlateNumber=Z A12345 Events[0].TrafficCar.DeviceAddress=Hangzhou ..... Events[1].EventBaseInfo.Code=TrafficJunction Events[1].EventBaseInfo.Action=Pulse Events[1].EventBaseInfo.Index=1 Events[1].Code=TrafficJunction ..... --&lt;boundary&gt; Content-Type: image/jpeg Content-Length:&lt;image size&gt;  &lt;JPEG image data&gt; --&lt;boundary&gt; Content-Type: text/plain Content-Length:&lt;data length&gt;  Heartbeat --&lt;boundary&gt; ..... </pre>
-----------------------	---

Comment	<p><b>ChannelNo:</b> integer, video channel index which starts from 1, default 1 if not specified.</p> <p><b>Heartbeat:</b> integer, range is [1,60], unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat message is the string "Heartbeat".</p> <p><b>eventCode:</b> it can be any one of the standard codes defined in DHIF.</p> <p><b>eventCode</b> includes:</p> <p>VideoMotion: motion detection event</p> <p>VideoLoss: video loss detection event</p> <p>VideoBlind: video blind detection event.</p> <p>AlarmLocal: alarm detection event.</p> <p>TrafficJunction: vehicle passing event</p> <p>TrafficRetrograde: retrograde event</p> <p>TrafficParking: illegal parking event</p> <p>TrafficJam: traffic congestion event</p> <p>TrafficThrow: throw something out of the car event</p> <p>TrafficPedestrian: pedestrian enter traffic lane(driverway) event</p> <p>TrafficParkingSpaceParking: a vehicle in the parking space event</p> <p>TrafficParkingSpaceNoParking: no vehicle in the parking space event</p> <p>TrafficParkingSpaceOverLine: a vehicle over the parking line event</p> <p>TrafficManualSnap: manual snapshot event</p> <p>TrafficSuspiciousCar: when the vehicle is in the blacklist, it will produce this event.</p> <p>FaceRecognition: face recognition event.</p> <p>TrafficParkingStatistics: Parking statistics event</p> <p>AccessControl                      Access control event</p> <p>CitizenPictureCompare              citizen picture compare event</p> <p>.....</p> <p>Also, the <b>eventCode</b> can be <b>All</b></p> <p><b>All</b> means all kinds of the eventcode.</p> <p>Parameters in Response:</p> <p><b>GroupID:</b> string, the ID of the snapshot event</p> <p><b>CountInGroup:</b> integer, the number of the snapshots</p> <p><b>IndexInGroup:</b> integer, the index of the snapshots. For example, the CountInGroup is 3, the IndexInGroup is 1, it means there are 3 pictures in this snapshot event, and this is the first picture.</p>
---------	---

## 4.5 Video Attributes

### 4.5.1 Get Max Extra Stream Numbers

Table 4-5

Syntax	<code>http://&lt;server&gt;/cgi-bin/magicBox.cgi?action=getProductDefinition&amp;name=MaxExtraStre</code>
--------	---

	am
Method	GET
Description	Get max extra stream count.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getProductDefinition&name=MaxExtraStream
Success Return	table. <b>MaxExtraStream</b> =1
Comment	<b>MaxExtraStream</b> : max extra stream numbers. It can be 1, 2 or 3.

## 4.5.2 Get Encode Capability

Table 4-6

Syntax	http://<server>/cgi-bin/encode.cgi?action=getCaps
Method	GET
Description	Get encode capabilities.
Example	http://192.168.1.108/cgi-bin/encode.cgi?action=getCaps
Success Return	caps.PlaybackCompressSplitNumList[0]=1 caps.PlaybackCompressSplitNumList[1]=2 caps.PlaybackCompressSplitNumList[2]=4 caps.PreviewMode=SplitSnap caps.VideoEncodeDevices[0].CoverAreaPercent=100 caps.VideoEncodeDevices[0].CoverCount=4 caps.VideoEncodeDevices[0].LadenBitrate=162201600 caps.VideoEncodeDevices[0].MaxCIFPFrameSize=40 caps.VideoEncodeDevices[0].MaxExtraStream=1 caps.VideoEncodeDevices[0].MinCIFPFrameSize=7 caps.VideoEncodeDevices[0].RecordIndividualResolution=true caps.VideoEncodeDevices[0].SupportIndividualResolution=true caps.VideoEncodeDevices[0].TitleCount=4 caps.CoriaDisplay[0].CoriaDisplaySwitchs[0]=0 caps.CoriaDisplay[0].CoriaDisplaySwitchs[1]=1 caps.CoriaDisplay[0].CoriaDisplaySwitchs[2]=2
Comment	—

## 4.5.3 Get Encode Config Capability

Table 4-7

Syntax	http://<server>/cgi-bin/encode.cgi?action=getConfigCaps[&channel=<ChannelNo>&stream=<StreamType>&<paramName>=<paramValue>&<paramName>=<paramValue>...]
Method	GET
Description	Get encode config capabilities.
Example	http://192.168.1.108/cgi-bin/encode.cgi?action=getConfigCaps&channel=1&Encode[0].MainFormat[0].Video.Width=1920&Encode[0].MainFormat[0].Video.Height=1080

Success Return	<p><b>headMain.Video.BitRateOptions</b>=448,2560  <b>headMain.Video.CompressionTypes</b>=H.264,MJPG  <b>headMain.Video.FPSMax</b>=25  <b>headMain.Video.ResolutionTypes</b>=2048 x 1536,1080,SXGA, 1280 x 960,720,D1,CIF  <b>headExtra.Video.BitRateOptions</b>=80,448  <b>headExtra.Video.CompressionTypes</b>=H.264,MJPG  <b>headExtra.Video.FPSMax</b>=25  <b>headExtra.Video.ResolutionTypes</b>=D1,CIF  <b>headSnap.Video.CompressionTypes</b>=H.264,MJPG  <b>headSnap.Video.ResolutionTypes</b>=2048 x 1536,1080,SXGA, 1280 x 960,720,D1,CIF</p>
Comment	<p>Parameters in URL:  <b>ChannelNo</b>: integer,video channel index which starts from 1.it is a optional parameter, if it is not present in the URL, the value of the channelno will be regarded as 1.  The description of <b>&lt;paramName&gt;</b> and <b>&lt;paramValue&gt;</b> is the same as 4.5.5(Set encode config).  <b>StreamType</b>:  "Main"-main stream  "Extra1"-extra stream 1  "Extra2"-extra stream 2  "Extra3"-extra stream 3</p> <p>Parameters in Response:  <b>headMain</b>= caps[<b>Channel</b>].MainFormat[<b>RecordType</b>]  <b>headExtra</b> = caps[<b>Channel</b>].ExtraFormat[<b>ExtraStream</b>]  <b>headSnap</b> = caps[<b>Channel</b>].SnapFormat[<b>SnapType</b>]</p> <p>Channel: integer, array index starts from 0, which means video channel (equals to video channel index ChannelNo -1, and so 0 means channel 1).  <b>RecordType</b>:  0 = regular record  1 = motion detection record  2 = alarm record  <b>ExtraStream</b>:  0 = extra stream 1  1 = extra stream 2  2 = extra stream 3  <b>SnapType</b>:  0 = regular snapshot  1 = motion detection snapshot  2 = alarm snapshot</p>

#### Appendix A: Encode Config Capabilities

Field in response	Value range	Description
BitRateOptions	string	Before comma is minimum bit rate. (kbps), after comma is maximum bit rate.(kbps) BitRateOptions=80,448 80 is the minimum bitrates, 448 is maximum.

Field in response	Value range	Description
CompressionTypes	string	To video, it contains all supported video compression types, separated by comma. Range is {MPEG4, MPEG2, MPEG1, MJPG, H.263, H.264, H.265, SVAC2}. To audio, it contains all supported audio compression types, separated by comma. Range is {PCM, ADPCM, G.711A, G.711Mu, G.726, G.729, MPEG2, AMR, SVAC}
FPSMax	integer	Maximum FPS.
ResolutionTypes	string	It contains all supported video resolutions. Range is in below Resolution list.

## Appendix B: Video Resolution

Fixed Resolution Name	Size in PAL	Size in NTSC
"D1"	704 x 576	704 x 480
"HD1"	352 x 576	352 x 480
"BCIF"/"2CIF"	704 x 288	704 x 240
"CIF"	352 x 288	352 x 240
"QCIF"	176 x 144	176 x 120
"NHD"	640 x 360	—
"VGA"	640 x 480	—
"QVGA"	320 x 240	—
"SVCD"	480 x 480	—
"QQVGA"	160 x 128	—
"SVGA"	800 x 592	—
"SVGA1"	800 x 600	—
"WVGA"	800 x 480	—
"FWVGA"	854 x 480	—
"DVGA"	960 x 640	—
"XVGA"	1024 x 768	—
"WXGA"	1280 x 800	—
"WXGA2"	1280 x 768	—
"WXGA3"	1280 x 854	—
"WXGA4"	1366 x 768	—
"SXGA"	1280 x 1024	—
"SXGA+"	1400 x 1050	—
"WSXGA"	1600 x 1024	—
"UXGA"	1600 x 1200	—
"WUXGA"	1920 x 1200	—
"ND1"	240 x 192	—
"720P"	1280 x 720	—
"1080P"	1920 x 1080	—
"QFHD"	3840 x 2160	—
"1_3M", "1280x960"	1280 x 960 (1.3 Mega Pixels)	—
"2_5M", "1872x1408"	1872 x 1408 (2.5 Mega Pixels)	—
"5M", "3744x1408"	3744 x 1408 (5 Mega Pixels)	—

Fixed Resolution Name	Size in PAL	Size in NTSC
"3M", "2048x1536"	2048 x 1536 (3 Mega Pixels)	—
"5_0M", "2432x2048"	2432 x 2048 (5 Mega Pixels)	—
"1_2M", "1216x1024"	1216 x 1024 (1.2 Mega Pixels)	—
"1408x1024"	1408 x 1024 (1.5 Mega Pixels)	—
"3296x2472"	3296 x 2472 (8 Mega Pixels)	—
"5_1M", "2560x1920"	2560 x 1920 (5 Mega Pixels)	—
"960H",	960 x 576	960 x 480
"DV720P"	960 x 720	—
"2560x1600"	2560 x 1600 (4 Mega Pixels)	—
"2336x1752"	2336 x 1752 (4 Mega Pixels)	—
"2592x2048"	2592 x 2048	—
"2448x2048"	2448 x 2048	—
"1920x1440"	1920x1440	—
"2752x2208"	2752x2208	—
"3840x2160"	3840x2160	—
"4096x2160"	4096x2160	—
"3072x2048"	3072x2048	—

### Appendix C: Audio Compression Type

Field in response	Value range	Description
CompressionTypes	string	It contains all supported audio compression types, separated by comma. Range is {PCM, ADPCM, G.711A, G.711Mu, G.726, G.729, MPEG2, AMR, SVAC}

## 4.5.4 Encode of Media

- Get encode config

Table 4-8

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Encode
Method	GET
Description	Get video encode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Encode
Success Return	<pre>table.Encode[0].MainFormat[0].Audio.Bitrate=64 table.Encode[0].MainFormat[0].Audio.Channels[0]=0 table.Encode[0].MainFormat[0].Audio.Compression=G.711A table.Encode[0].MainFormat[0].Audio.Depth=16 table.Encode[0].MainFormat[0].Audio.Frequency=8000 table.Encode[0].MainFormat[0].Audio.Mode=0 table.Encode[0].MainFormat[0].Audio.Pack=DHAV table.Encode[0].MainFormat[0].Audio.Enable=true table.Encode[0].MainFormat[0].Video.resolution=1920x1080 table.Encode[0].MainFormat[0].Video.BitRate=4096 table.Encode[0].MainFormat[0].Video.BitRateControl=CBR table.Encode[0].MainFormat[0].Video.Compression=H.264</pre>

table.Encode[0].MainFormat[0].Video.CustomResolutionName=1080P  
table.Encode[0].MainFormat[0].Video.FPS=18  
table.Encode[0].MainFormat[0].Video.GOP=36  
table.Encode[0].MainFormat[0].Video.Height=1080  
table.Encode[0].MainFormat[0].Video.Pack=DHAV  
table.Encode[0].MainFormat[0].Video.Profile=High  
table.Encode[0].MainFormat[0].Video.Quality=4  
table.Encode[0].MainFormat[0].Video.QualityRange=6  
table.Encode[0].MainFormat[0].Video.SVCTLayer=1  
table.Encode[0].MainFormat[0].Video.Width=1920  
table.Encode[0].MainFormat[0].Video.Priority=0  
table.Encode[0].MainFormat[0].VideoEnable=true  
table.Encode[0].MainFormat[1].Audio.Bitrate=64  
table.Encode[0].MainFormat[1].Audio.Channels[0]=0  
table.Encode[0].MainFormat[1].Audio.Compression=G.711A  
table.Encode[0].MainFormat[1].Audio.Depth=16  
table.Encode[0].MainFormat[1].Audio.Frequency=8000  
table.Encode[0].MainFormat[1].Audio.Mode=0  
table.Encode[0].MainFormat[1].Audio.Pack=DHAV  
table.Encode[0].MainFormat[1].AudioEnable=true  
table.Encode[0].MainFormat[1].Video.resolution=1920x1080  
table.Encode[0].MainFormat[1].Video.BitRate=4096  
table.Encode[0].MainFormat[1].Video.BitRateControl=CBR  
table.Encode[0].MainFormat[1].Video.Compression=H.264  
table.Encode[0].MainFormat[1].Video.CustomResolutionName=1080P  
table.Encode[0].MainFormat[1].Video.FPS=18  
table.Encode[0].MainFormat[1].Video.GOP=36  
table.Encode[0].MainFormat[1].Video.Height=1080  
table.Encode[0].MainFormat[1].Video.Pack=DHAV  
table.Encode[0].MainFormat[1].Video.Profile=High  
table.Encode[0].MainFormat[1].Video.Quality=4  
table.Encode[0].MainFormat[1].Video.QualityRange=6  
table.Encode[0].MainFormat[1].Video.SVCTLayer=1  
table.Encode[0].MainFormat[1].Video.Width=1920  
table.Encode[0].MainFormat[1].Video.Priority=0  
table.Encode[0].MainFormat[1].VideoEnable=true  
table.Encode[0].MainFormat[2].Audio.Bitrate=64  
table.Encode[0].MainFormat[2].Audio.Channels[0]=0  
table.Encode[0].MainFormat[2].Audio.Compression=G.711A  
table.Encode[0].MainFormat[2].Audio.Depth=16  
table.Encode[0].MainFormat[2].Audio.Frequency=8000  
table.Encode[0].MainFormat[2].Audio.Mode=0  
table.Encode[0].MainFormat[2].Audio.Pack=DHAV  
table.Encode[0].MainFormat[2].AudioEnable=true  
table.Encode[0].MainFormat[2].Video.resolution=1920x1080  
table.Encode[0].MainFormat[2].Video.BitRate=4096  
table.Encode[0].MainFormat[2].Video.BitRateControl=CBR

table.Encode[0].MainFormat[2].Video.Compression=H.264  
table.Encode[0].MainFormat[2].Video.CustomResolutionName=1080P  
table.Encode[0].MainFormat[2].Video.FPS=18  
table.Encode[0].MainFormat[2].Video.GOP=36  
table.Encode[0].MainFormat[2].Video.Height=1080  
table.Encode[0].MainFormat[2].Video.Pack=DHAV  
table.Encode[0].MainFormat[2].Video.Profile=High  
table.Encode[0].MainFormat[2].Video.Quality=4  
table.Encode[0].MainFormat[2].Video.QualityRange=6  
table.Encode[0].MainFormat[2].Video.SVCTLayer=1  
table.Encode[0].MainFormat[2].Video.Width=1920  
table.Encode[0].MainFormat[2].Video.Priority=0  
table.Encode[0].MainFormat[2].VideoEnable=true  
table.Encode[0].MainFormat[3].Audio.Bitrate=64  
table.Encode[0].MainFormat[3].Audio.Channels[0]=0  
table.Encode[0].MainFormat[3].Audio.Compression=G.711A  
table.Encode[0].MainFormat[3].Audio.Depth=16  
table.Encode[0].MainFormat[3].Audio.Frequency=8000  
table.Encode[0].MainFormat[3].Audio.Mode=0  
table.Encode[0].MainFormat[3].Audio.Pack=DHAV  
table.Encode[0].MainFormat[3].AudioEnable=true  
table.Encode[0].MainFormat[3].Video.resolution=704x576  
table.Encode[0].MainFormat[3].Video.BitRate=2048  
table.Encode[0].MainFormat[3].Video.BitRateControl=VBR  
table.Encode[0].MainFormat[3].Video.Compression=H.264  
table.Encode[0].MainFormat[3].Video.FPS=25  
table.Encode[0].MainFormat[3].Video.GOP=50  
table.Encode[0].MainFormat[3].Video.Height=576  
table.Encode[0].MainFormat[3].Video.Pack=DHAV  
table.Encode[0].MainFormat[3].Video.Profile=Main  
table.Encode[0].MainFormat[3].Video.Quality=4  
table.Encode[0].MainFormat[3].Video.QualityRange=6  
table.Encode[0].MainFormat[3].Video.SVCTLayer=1  
table.Encode[0].MainFormat[3].Video.Width=704  
table.Encode[0].MainFormat[3].Video.Priority=0  
table.Encode[0].MainFormat[3].VideoEnable=true  
table.Encode[0].ExtraFormat[0].Audio.Bitrate=64  
...  
table.Encode[0].SnapFormat[0].Audio.Bitrate=64  
...

Comment	<p>Parameters in Response:</p> <p>The format of the config is <i>head.configItems</i>. The <i>head</i> can be:  <i>headMain</i>= table. Encode[<b>Channel</b>].MainFormat[<b>Type</b>]  <i>headSnap</i> = table. Encode[<b>Channel</b>].SnapFormat[<b>Type</b>]  <i>headExtra</i> =table. Encode[<b>Channel</b>].ExtraFormat[<b>ExtraStream</b>]</p> <p><b>Channel</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>Type</b>:</p> <p>0 = regular encode  1 = motion detection encode  2 = alarm encode  3= emergency encode</p> <p><b>ExtraStream</b>:</p> <p>0 = extra stream 1  1 = extra stream 2  2 = extra stream 3</p> <p>The <i>configItems</i> are list as bellow.</p>
---------	--

- Set encode config

Table 4-9

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set encode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Encode[1].MainFormat[0].Video.Compression=MPEG4&
Success Return	OK
Comment	<p>Parameters in URL:</p> <p>In table below,  <b>head</b>=Encode[<b>Channel</b>].MainFormat[<b>RecordType</b>] (or)  Encode[<b>Channel</b>].ExtraFormat[<b>ExtraStream</b>]</p> <p><b>Channel</b>: integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).</p> <p><b>RecordType</b>:</p> <p>0 = regular record  1 = motion detection record  2 = alarm record</p> <p><b>ExtraStream</b>:</p> <p>0 = extra stream 1  1 = extra stream 2  2 = extra stream 3</p>

## Appendix A: Video Encode Config

ParamName	ParamValue type	Description
<b>head.Video.BitRate</b>	integer	Unit is Kbps Range depends on capability in <a href="#">GetVideoConfigCaps</a>
<b>head.Video.BitRateControl</b>	string	Range is {CBR, VBR} CBR: constant bitrates VBR: variable bitrates
<b>head.Video.Compression</b>	String	Range is {MPEG4, MPEG2, MPEG1, MJPG, H.263, H.264, H.265, SVAC2} Depends on capacity in <a href="#">GetVideoConfigCaps</a>
<b>head.Video.FPS</b>	float	Frames per second. <1.0: several seconds/frame, FPS=0.3333: 3 seconds per frame. >1.0: several frames/second. FPS=3: 3 frames per second. Range depends on capability in <a href="#">GetVideoConfigCaps</a>
<b>head.Video.GOP</b>	integer	Range is [1—100]. Group of picture, it's the interval of I Frame, Example: GOP=50, means there is one I frame every 49 P or B frames
<b>head.Video.Height</b>	integer	Video height
<b>head.Video.Width</b>	integer	Video Width
<b>head.Video.Profile</b>	String	Range is { Baseline, Main , Extended , High } Only when video compression is H.264, it's effective.
<b>head.Video.Quality</b>	integer	Range is [1—6]. Image Quality, available when Video.BitRateControl=VBR 1: worst quality 6: best quality
<b>head.Video.Priority</b>	integer	Range is 0 or 1 0 for Image Quality first 1 for Bitstream first
<b>head.VideoEnable</b>	bool	True: enable video

#### Appendix B: Audio Encode Config

ParamName	ParamValue type	Description
<b>head.Audio.Bitrate</b>	integer	Unit is kbps Range depends on capacity in <a href="#">GetAudioConfigCaps</a>
<b>head.Audio.Compression</b>	string	Range depends on capacity in <a href="#">GetAudioConfigCaps</a>
<b>head.Audio.Depth</b>	integer	Audio sampling depth
<b>head.Audio.Frequency</b>	integer	Audio sampling frequency

ParamName	ParamValue type	Description
<i>head</i> .Audio.Mode	integer	Range is {0,1,2,3,4,5,6,7} Audio encode mode. 0: 4.75kbps, 1: 5.15 kbps, 2: 5.9 kbps, 3: 6.7 kbps, 4: 7.4 kbps, 5: 7.95 kbps, 6: 10.2 kbps, 7: 12.2 kbps,
<i>head</i> .AudioEnable	bool	Enable/Disable audio

## 4.5.5 Encode of Region Interested

- Get encode config of region interested

Table 4-10

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoEncodeROI
Method	GET
Description	Get video encode config of region interested.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoEncodeROI
Success Return	<i>head</i> .Regions[0][0]=0 <i>head</i> .Regions[0][1]=0 <i>head</i> .Regions[0][2]=0 <i>head</i> .Regions[0][3]=0 <i>head</i> .Regions[1][0]=0 ... <i>head</i> .Quality=4, <i>head</i> .Main=true, <i>head</i> .Extra1=true, <i>head</i> .Extra2=true, <i>head</i> .Extra3=true, <i>head</i> .Snapshot=true <i>head</i> .DynamicTrack=true, <i>head</i> .DynamicDelayTime=60
Comment	Parameters in Response: <i>head</i> =table.VideoEncodeROI[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set encode config of region interested

Table 4-11

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
--------	---

Method	GET
Description	Set video encode config of region interested.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoEncodeROI[0].DynamicTrack=true
Success Return	OK
Comment	Parameters in URL: <b>paramName</b> and <b>paramValue</b> are as table below. In table below, <b>head</b> = VideoEncodeROI[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).

Appendix:

ParamName	ParamValue type	Description
<b>head</b> . DynamicTrack	bool	Enable/Disable

## 4.5.6 Channel Title

- Get channel title

Table 4-12

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=ChannelTitle
Method	GET
Description	Get the title of the video channel.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=ChannelTitle
Success Return	table.ChannelTitle[ <b>Channel</b> ].Name=CAM1 123
Comment	Parameters in Response: <b>Channel</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>Name</b> : string, which means title content. character ' ' means newlines.

- Set channel title

Table 4-13

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>
Method	GET
Description	Set the title of the channel.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&ChannelTitle[1].Name= test1 test2
Success Return	OK
Comment	If VideoWidget[ <b>Channel</b> ].ChannelTitle.EncodeBlend is true, this title is blended to the video frames. Please refer to SetVideoWidgetConfig.

	Parameters in URL: Channel Name Format: ChannelTitle[ <b>Channel</b> ].Name <b>Channel</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1). Name: string, which means title content. Character ' ' means newlines.  Limit: Support 2 lines at most.
--	---

## 4.5.7 Get Video Input Channels Device Supported

Table 4-14

Syntax	http://<server>/cgi-bin/devVideoInput.cgi?action=getCollect
Method	GET
Description	Get the video input channel numbers.
Example	http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getCollect
Success Return	result=1
Comment	—

## 4.5.8 Get Video Output Channels Device Supported

Table 4-15

Syntax	http://<server>/cgi-bin/devVideoOutput.cgi?action=getCollect
Method	GET
Description	Get the video output channel numbers.
Example	http://192.168.1.108/cgi-bin/devVideoOutput.cgi?action=getCollect
Success Return	result=2
Comment	—

## 4.5.9 Get Max Remote Input Channels

Table 4-16

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getProductDefinition&name=MaxRemoteInputChannels
Method	GET
Description	Get max remote input channels.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getProductDefinition&name=MaxRemoteInputChannels
Success Return	table.MaxRemoteInputChannels=16

Comment	MaxRemoteInputChannels: max remote input channels.
---------	--

## 4.5.10 Video Standard

- Get video standard

Table 4-17

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoStandard
Method	GET
Description	Get video standard config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoStandard
Success Return	table.VideoStandard=PAL
Comment	—

- Set video standard

Table 4-18

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&VideoStandard=<paramValue>
Method	GET
Description	Set video standard config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoStandard=PAL
Success Return	OK
Comment	VideoStandard: string, range is {PAL, NTSC}; Video Standard.

## 4.5.11 Video Widget

- Get video widget config

Table 4-19

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoWidget
Method	GET
Description	Video widget config contains Channel Title, User-defined Title, Covers and Time Title parameters, defines the background color, front color and positions of channel title and time title, User-defined title and defines the regions which are not visible (cover).
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoWidget
Success Return	<pre> <b>head</b>.BackColor[0]=0 <b>head</b>.BackColor[1]=0 <b>head</b>.BackColor[2]=0 <b>head</b>.BackColor[3]=128 <b>head</b>.EncodeBlend=true <b>head</b>.FrontColor[0]=255 <b>head</b>.FrontColor[1]=255 </pre>

	<pre> <b>head</b>.FrontColor[2]=255 <b>head</b>.FrontColor[3]=0 <b>head</b>.Rect[0]=0 <b>head</b>.Rect[1]=8191 <b>head</b>.Rect[2]=0 <b>head</b>.Rect[3]=8191 ... </pre>
Comment	<p>Parameters in Response:</p> <pre> <b>head</b>=table.VideoWidget[<b>Channel</b>].ChannelTitle (or) table.VideoWidget[<b>Channel</b>].Covers[<b>CoReg</b>] (or) table.VideoWidget[<b>Channel</b>].TimeTitle table.VideoWidget[<b>Channel</b>].UserDefinedTitle[<b>Index</b>] </pre> <p><b>Channel</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>CoReg</b>: Cover Region, Covers is an array which sustains multi- Cover regions</p> <ul style="list-style-type: none"> <li>0 = region 1</li> <li>1 = region 2</li> <li>2 = region 3</li> <li>3 = region 4</li> </ul> <p><b>Index</b>: integer, User-defined title index, now only index 0 is valid, others are reserved.</p>

- Set video widget config

Table 4-20

Syntax	<pre> http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;[&amp;&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;...] </pre>
Method	GET
Description	Set video widget config.
Example	<pre> http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoWidget[1].Covers[0].BackColor[0]=128&amp;VideoWidget[1].Covers[0].BackColor[1]=128&amp;VideoWidget[1].Covers[0].BackColor[2]=128&amp;VideoWidget[1].Covers[0].BackColor[3]=0 </pre>
Success Return	OK
Comment	<p>Parameters in URL:</p> <p>In table below,</p> <pre> <b>headChannelTitle</b> = VideoWidget[<b>Channel</b>].ChannelTitle <b>headCover</b>       = VideoWidget[<b>Channel</b>].Covers[<b>CoReg</b>] <b>headTimeTitle</b>  = VideoWidget[<b>Channel</b>].TimeTitle <b>headUserDefinedTitle</b> = VideoWidget[<b>Channel</b>].UserDefinedTitle [ <b>Index</b> ] </pre> <p><b>Channel</b>: integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).</p> <p><b>CoReg</b>: Cover region index. Covers is an array which contains multiple cover regions</p> <ul style="list-style-type: none"> <li>0 = region 1</li> <li>1 = region 2</li> <li>2 = region 3</li> </ul>

3 = region 4

**Index:** integer, User-defined title index, now only index 0 is valid, others are reserved.

**Appendix:**

ParamName	ParamValue type	Description
<i>headCover.BackColor</i> [0] <i>headCover.BackColor</i> [1] <i>headCover.BackColor</i> [2] <i>headCover.BackColor</i> [3]	integer	Range is [0—255]. BackColor[0]:red value BackColor[1]:green value BackColor[2]:blue value BackColor[3]: alpha value
<i>headCover.EncodeBlend</i>	bool	false - widget blend is disabled.
<i>headCover.FrontColor</i> [0] <i>headCover.FrontColor</i> [1] <i>headCover.FrontColor</i> [2] <i>headCover.FrontColor</i> [3]	integer	Range is [0—255]. FrontColor[0]:red value FrontColor[1]:green value FrontColor[2]:blue value FrontColor[3]: alpha value
<i>headCover.Rect</i> [0] <i>headCover.Rect</i> [1] <i>headCover.Rect</i> [2] <i>headCover.Rect</i> [3]	integer	Range is [0—8191]. Rect[0]: top left corner x coordinate (left) Rect[1]: top left corner y coordinate (top) Rect[2]: bottom right x coordinate (right) Rect[3]: bottom right y coordinate (bottom)
<i>headChannelTitle.BackColor</i> [0] <i>headChannelTitle.BackColor</i> [1] <i>headChannelTitle.BackColor</i> [2] <i>headChannelTitle.BackColor</i> [3]	integer	Range is the same with <i>headCover</i> .
<i>headChannelTitle.EncodeBlend</i>	bool	
<i>headChannelTitle.FrontColor</i> [0] <i>headChannelTitle.FrontColor</i> [1] <i>headChannelTitle.FrontColor</i> [2] <i>headChannelTitle.FrontColor</i> [3]	integer	
<i>headChannelTitle.Rect</i> [0] <i>headChannelTitle.Rect</i> [1] <i>headChannelTitle.Rect</i> [2] <i>headChannelTitle.Rect</i> [3]	integer	Only use the value of (left, top),the value of (right, bottom) is the same as (left, top) Rect[0], Rect[1] are used, and Rect[2] must be same with Rect[0], Rect[3] must be same with Rect[1].
<i>headUserDefinedTitle.Text</i>	string	Title content. Character '\n' means newlines. Limit: Support 2 lines at most.
<i>headUserDefinedTitle.TextAlign</i>	integer	Range is the same with <i>headChannelTitle</i>
<i>headUserDefinedTitle.BackColor</i> [0] <i>headUserDefinedTitle.BackColor</i> [1] <i>headUserDefinedTitle.BackColor</i> [2] <i>headUserDefinedTitle.BackColor</i> [3]	integer	These are configs about User-defined title.
<i>headUserDefinedTitle.EncodeBlend</i>	bool	
<i>headUserDefinedTitle.FrontColor</i> [0] <i>headUserDefinedTitle.FrontColor</i> [1]	integer	

ParamName	ParamValue type	Description
<i>headUserDefinedTitle.FrontColor[2]</i> <i>headUserDefinedTitle.FrontColor[3]</i>		
<i>headUserDefinedTitle.Rect[0]</i> <i>headUserDefinedTitle.Rect[1]</i> <i>headUserDefinedTitle.Rect[2]</i> <i>headUserDefinedTitle.Rect[3]</i>	integer	
<i>headTimeTitle.BackColor[0]</i> <i>headTimeTitle.BackColor[1]</i> <i>headTimeTitle.BackColor[2]</i> <i>headTimeTitle.BackColor[3]</i>	integer	
<i>headTimeTitle.EncodeBlend</i>	bool	
<i>headTimeTitle.FrontColor[0]</i> <i>headTimeTitle.FrontColor[1]</i> <i>headTimeTitle.FrontColor[2]</i> <i>headTimeTitle.FrontColor[3]</i>	integer	Range is the same with <i>headChannelTitle</i> . These are configs about time title.
<i>headTimeTitle.Rect[0]</i> <i>headTimeTitle.Rect[1]</i> <i>headTimeTitle.Rect[2]</i> <i>headTimeTitle.Rect[3]</i>	integer	
<i>headTimeTitle.ShowWeek</i>	bool	True: Display week within the time title.

## 4.5.12 Get Video Input Capability

Table 4-21

Syntax	<code>http://&lt;server&gt;/cgi-bin/devVideoInput.cgi?action=getCaps&amp;channel=&lt;ChannelNo&gt;</code>
Method	GET
Description	Get video input capabilities.
Example	<code>http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getCaps&amp;channel=1</code>
Success Return	<pre> caps.AutoSyncPhase=false caps.Backlight=2 caps.BrightnessCompensation=true caps.ChipID=0 caps.CoverCount=4 caps.CoverType=1 caps.CustomManualExposure=true caps.DayNightColor=true caps.DayNightColorIO=false caps.DoubleExposure=0 caps.DownScaling=false caps.EEModeRange=100 caps.ElectricFocus=false caps.Exposure=16 caps.ExposureMode=31 caps.ExternalSyncInput=false caps.FishEye=false </pre>

caps.FlashAdjust=false  
caps.Flip=true  
caps.FormatCount=5  
caps.Gain=true  
caps.GainAuto=true  
caps.Gamma=true  
caps.GammaModeRange=100  
caps.GlareInhibition=1  
caps.HorizontalBinning=0  
caps.IRCUT=true  
caps.ImageEnhancement.LevelRange[0]=0  
caps.ImageEnhancement.LevelRange[1]=100  
caps.ImageEnhancement.Support=true  
caps.InfraRed=true  
caps.Iris=true  
caps.IrisAuto=true  
caps.LadenBitrate=972000  
caps.LimitedAutoExposure=true  
caps.MaxExposureTime=300.0  
caps.MaxExposureTime1=0.0  
caps.MaxHeight=1080  
caps.MaxMultiProfile=3  
caps.MaxWidth=1920  
caps.MeteringRegionCount=0  
caps.MinExposureTime=1.0  
caps.MinExposureTime1=0.0  
caps.Mirror=true  
caps.MultiOptions=false  
caps.NightOptions=true  
caps.ReferenceLevel=false  
caps.Rotate90=true  
caps.SetColor=true  
caps.SignalFormats=Inside  
caps.SignalType[0]=VGA  
caps.SnapshotExposure=false  
caps.SupportProfile=false  
caps.SupportWhiteLevel=true  
caps.SupportWriteLevel=false  
caps.SyncChipChannels=false  
caps.SyncFocus=false  
caps.TitleCount=4  
caps.TridimDenoise=2  
caps.TridimDenoiseDetails=0  
caps.UTC=0  
caps.UpScaling=false  
caps.Version=0  
caps.VerticalBinning=0

	caps.VideoInDenoise.2D.LevelRange[0]=0 caps.VideoInDenoise.2D.LevelRange[1]=100 caps.VideoInDenoise.2D.Support=true caps.VideoInDenoise.3D.3DAutoType.ModRange[0]=0 caps.VideoInDenoise.3D.3DAutoType.ModRange[1]=100 caps.VideoInDenoise.3D.Support=true caps.VideoInDenoise.Support=true caps.WhiteBalance=3 caps.WideDynamicRange=1
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. Parameters in Response: see table below

### Appendix:

Field in response	Value type	Description
Backlight	integer	0: not support backlight ; 1: support backlight; 2: support regional regulation
ChipID	String	ID of chips in this channel
CoverCount	integer	The maximum cover region count.
CoverType	integer	0: don't support cover 1: support realtime cover 2: support non-realtime cover
CustomManualExposure	bool,	true: support use defined manual exposure time
DayNightColor	bool	true: support color alternate between day and night.
DownScaling	bool	true: support down scaling, binning mode not included.
Exposure	integer	Exposure grade. 0 — don't support exposure control.
ExternalSyncInput	bool	true: support HD signal external synchronization.
FlashAdjust	bool	true: support flash adjust
Flip	bool	true: support picture flip.
Gain	bool	true: support gain control.
GainAuto	bool	true: support auto gain.
HorizontalBinning	integer	Horizontal/Vertical pixel binning mask, 1 — support 2 pixel binning, 2 — support 3 pixel binning 4 — support 4 pixel binning ...
VerticalBinning	integer	2^n — support n+2 pixel binning
InfraRed	bool	true: support Infra compensation
Iris	bool	true: support Iris adjust
IrisAuto	bool	true: support auto Iris adjust
LadenBitrate	integer	Unit is Kbps. Maximum value of video stream bitrates, 16bpp, not in binning mode.

Field in response	Value type	Description
LimitedAutoExposure	bool	true: support auto exposure with time limit.
MaxExposureTime	double	upper limit of exposure time, unit: ms
MaxExposureTime1	double	upper limit of exposure time, unit: ms
MaxHeight	integer	Maximum video height
MaxWidth	integer	Maximum video width
MinExposureTime	double	lower limit of exposure time, unit: ms
MinExposureTime1	double	lower limit of exposure time, unit: ms
Mirror	bool	true: support picture mirror.
NightOptions	bool	true: support night options.
ReferenceLevel	bool	true: support reference level.
Rotate90	bool	true: support clockwise/anticlockwise 90° rotate
SetColor	bool	true: support color set.
SignalFormats	string	It's a string contains supported video input signal formats for this channel. Signal formats are separated by comma. Range is {Inside, BT656, 720p, 1080p, 1080i, 1080sF, 1_3M}  Inside — inside input. 1_3M — 1280*960
SyncChipChannels	bool	True: channels in same chip should be synchronized. Synchronized means video resolution of these channels should be the same.
TitleCount	integer	Maximum count of blending titles.
UpScaling	bool	true: support up scaling.
WhiteBalance	integer	Range is {0, 1, 2, 3} 0: don't support white balance. 1: support auto white balance 2: support auto and pre defined white balance. 3: support auto, pre defined and user defined white balance

### 4.5.13 Get Coordinates of Current Window

Table 4-22

Syntax	http://<server>/cgi-bin/devVideoInput.cgi?action=getCurrentWindow&channel=<ChannelNo>
Method	GET
Description	Get the coordinates of the current window.
Example	http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getCurrentWindow&channel=1
Success Return	<b>rect</b> [0]=500 <b>rect</b> [1]=500 <b>rect</b> [2]=5000 <b>rect</b> [3]=5000
Comment	Parameters in URL:

	<p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p>Parameters in Response:</p> <p><b>rect[n]</b> : relative coordinates, range is 0-8192. {0,0,0,0} top-left, {8192,0,0,0} top-right, {0,8192,0,0} bottom-left, {8192,8192,0,0} bottom-right</p>
--	---

## 4.5.14 Set Coordinates of Current Window

Table 4-23

Syntax	http://<server>/cgi-bin/devVideoInput.cgi?action=setCurrentWindow&channel=< <b>ChannelNo</b> >&rect[0]=< <b>rect0</b> >&rect[1]=< <b>rect1</b> >&rect[2]=< <b>rect2</b> >&rect[3]=< <b>rect3</b> >
Method	GET
Description	Set the coordinates of the current window.
Example	http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=setCurrentWindow&channel=1&rect[0]=0&rect[1]=0&rect[2]=5000&rect[3]=5000
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p><b>rect0 &amp; rect1 &amp; rect2 &amp; rect3</b>: relative coordinates, range is 0-8192. {0,0,0,0} top-left, {8192,0,0,0} top-right, {0,8192,0,0} bottom-left, {8192,8192,0,0} bottom-right</p>

## 4.5.15 Video Out

- Get video out config

Table 4-24

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoOut
Method	GET
Description	Get video out config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoOut
Success Return	<p><b>head.Margin</b>[0]=0</p> <p><b>head.Margin</b>[1]=0</p> <p><b>head.Margin</b>[2]=0</p> <p><b>head.Margin</b>[3]=0</p> <p><b>head.Color.Brightness</b>=50</p> <p><b>head.Color.Contrast</b> =50</p> <p><b>head.Color.Saturation</b> =50</p> <p><b>head.Color.Hue</b> =50</p> <p><b>head.Mode.Width</b> =800</p> <p><b>head.Mode.Height</b>=600</p> <p><b>head.Mode.BPP</b> =16</p> <p><b>head.Mode.Format</b> ="Auto"</p> <p><b>head.Mode.RefreshRate</b> =60</p> <p>...</p>
Comment	Parameters in Response:

	<p><b>head</b> = table.VideoOut[<b>ChannelNo</b>].</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p>
--	---

- Set video out config

Table 4-25

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set video out config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&table.VideoOut[1].Color.Brightness=50
Success Return	OK
Comment	<p>Parameters in URL:</p> <p>In table below,</p> <p><b>head</b> = table.VideoOut[<b>ChannelNo</b>].</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p>

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> . Margin[0]	integer	Margin.
<b>head</b> . Margin[1]		
<b>head</b> . Margin[2]		
<b>head</b> . Margin[3]		
<b>head</b> . Color.Brightness	integer	Brightness.
<b>head</b> . Color.Contrast =50	integer	Contrast.
<b>head</b> .Color.Satuation =50	integer	Satuation.
<b>head</b> . Color.Hue =50	integer	Hue.
<b>head</b> . Mode.Width =800	integer	Resolution.
<b>head</b> . Mode.Height=600		
<b>head</b> . Mode.BPP =16	integer	Bits per pixel.
<b>head</b> .Mode.Format="Auto"	string	The range is {"Auto","TV","VGA","DVI","HDMI","SPOT","SDI","HDMI/VGA"}.
<b>head</b> .Mode.RefreshRate=60	integer	Refresh rate.

### 4.5.16 Smart Code

- Get Smart Code config

Table 4-26

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=SmartEncode
Method	GET
Description	Get smart code config.

Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=SmartEncode
Success Return	head.Enable=false head.Extra[0]=false
Comment	Parameters in Response: <b>head</b> = table.SmartEncode[ <b>ChannelNo</b> ]. <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set Smart Code config

Table 4-27

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set smart code config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&SmartEncode[0].Enable=true
Success Return	OK
Comment	Parameters in URL: In table below, <b>head</b> = SmartEncode[ <b>ChannelNo</b> ]. <b>ChannelNo</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).

#### Appendix:

ParamName	ParamValue type	Description
<b>head</b> .Enbale	bool	Open or close the smart code of the main stream.
<b>head</b> .Extra[0]	bool	Open or close the smart code of the extra stream.

### 4.5.17 Get Decoder Caps

URL	http://<server>/cgi-bin/DevVideoDec.cgi?action=getCaps		
Method	GET		
Description	Get capability of decoder		
[ Request Params ] (None )			
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
Video	object	R	The video caps
+StreamType	array<string>	R	The stream types that the device supports, ex : h264, h265 etc.
+LadenBitrate	int	R	The maximum decoding capability that the device supports (w * h * FPS * 16)/1024, uint: kbps, ex : 7680*4320*30*8/1024 = 7776000
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/DevVideoDec.cgi?action=getCaps		

Response	Video.StreamType[0]=h264 Video.StreamType[1]=h265 Video.LadenBitrate=7776000
----------	--

## 4.5.18 [Config] PrivacyMasking

- Get privacy masking config

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=PrivacyMasking		
Method	GET		
Description	Get privacy masking setting		
[ Request Params ] (None )			
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
table	object	R	Config info table.
+PrivacyMasking	Array<Array<object>>	R	PrivacyMasking config, each channel has several masking region
++Enable	bool	R	Enable/Disable
++Postion	Array<double>	O	Size is 3 Postion[0] is horizontal angle, range is [-1—1] Postion[1] is vertical angel, range is [-1—1] Postion[2] is zoom, range is [ 0—1]
++ShapeType	string	O	Shape type,can be: "Rect", "Polygon"
++Rect	Array<int>	O	It is valid if ShapeType is Rect , top left and bottom right point, each point has x and y value, oordinate remap to 0 — 8192.
++Polygon	Array<Array<int>>	O	It is valid if ShapeType is Polygon the first array is point list, minimum item is 3, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
++Color	Array<int>	O	Mask color, RGBA color, 4 integer range is 0-255
++Mosaic	int	O	Mosaic type, Range is {0, 8, 16, 24, 32} 0: no mosaic 8: mosaic size 8*8 16: mosaic size 16*16 24: mosaic size 24*24 32: mosaic size 32*32
++ViewAngle	double	O	View angle, range is [0.0, 360.0], unit: degree
[ Example ]			
Request	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=PrivacyMasking		

Response	<pre> table.PrivacyMasking[0][0].Enable=true table.PrivacyMasking[0][0].Postion[0]=0.0 table.PrivacyMasking[0][0].Postion[1]=0.0 table.PrivacyMasking[0][0].Postion[2]=1.0 table.PrivacyMasking[0][0].ShapeType=Rect table.PrivacyMasking[0][0].Rect[0]=0 table.PrivacyMasking[0][0].Rect[1]=0 table.PrivacyMasking[0][0].Rect[2]=50 table.PrivacyMasking[0][0].Rect[3]=100 table.PrivacyMasking[0][0].Color[0]=128 table.PrivacyMasking[0][0].Color[1]=128 table.PrivacyMasking[0][0].Color[2]=128 table.PrivacyMasking[0][0].Color[3]=255 table.PrivacyMasking[0][0].Mosaic=8 table.PrivacyMasking[0][0].ViewAngle=30.0 table.PrivacyMasking[0][1].Enable=true ... </pre>
----------	---

- Set privacy masking config

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set privacy masking setting		
[ Request Params ] (key=value format )			
Name	Type	R/O	Param Description
PrivacyMasking	Array<Array<object>>	R	privacy masking config, each channel has several masking region
+Enable	bool	R	Enable/Disable
+Postion	Array<double>	O	Size is 3 Postion[0] is horizontal angle, range is [-1—1] Postion[1] is vertical angel, range is [-1—1] Postion[2] is zoom, range is [ 0—1]
+ShapeType	string	O	Shape type,can be: "Rect", "Polygon"
+Rect	Array<int>	O	It is valid if ShapeType is Rect , top left and bottom right point, each point has x and y value, ordinate remap to 0 — 8192.
+Polygon	Array<Array<int>>	O	It is valid if ShapeType is Polygon the first array is point list, minimum item is 3, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
+Color	Array<int>	O	Mask color, RGBA color, 4 integer range is 0-255
+Mosaic	int	O	Mosaic type, Range is {0, 8, 16, 24, 32} 0: no mosaic 8: mosaic size 8*8 16: mosaic size 16*16 24: mosaic size 24*24 32: mosaic size 32*32
+ViewAngle	double	O	View angle, range is [0.0, 360.0], unit: degree

[ Response Params ] ( OK )	
[ Example ]	
Request	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&PrivacyMasking[0][0].Enable=true&PrivacyMasking[0][0].Postion[0]=0.0&PrivacyMasking[0][0].Postion[1]=0.0&PrivacyMasking[0][0].Postion[2]=1.0&PrivacyMasking[0][0].ShapeType=Rect&PrivacyMasking[0][0].Rect[0]=0&PrivacyMasking[0][0].Rect[1]=0&PrivacyMasking[0][0].Rect[2]=50&PrivacyMasking[0][0].Rect[3]=100&PrivacyMasking[0][0].Mosaic=8&PrivacyMasking[0][0].ViewAngle=30.0
Response	OK

## 4.5.19 Get Privacy Masking

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=getPrivacyMasking		
Method	GET		
Description	Get privacy masking		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
offset	int	R	Offset in the result record set, range is [0, Total – 1].
limit	int	R	Count of result to get.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
total	int	R	Count of result
PrivacyMasking	Array<object >	R	PrivacyMasking config
+Index	int	R	Index of PrivacyMasking which starts from 0
+Name	string	O	PrivacyMasking name
+Enable	int	R	1: enable; 0: disable
+ShapeType	string	O	Shape type, can be: "Rect", "Polygon"
+Rect	Array<int>	O	It is valid if ShapeType is Rect , top left and bottom right point, each point has x and y value, oordinate remap to 0 — 8192.
+Polygon	Array<Array<int>>	O	It is valid if ShapeType is Polygon the first array is point list, minimum item is 3, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
+Color	Array<int>	O	Mask color, RGBA color, 4 integer range is 0-255
+Mosaic	int	O	Mosaic type, Range is {0, 8, 16, 24, 32} 0: no mosaic 8: mosaic size 8*8 16: mosaic size 16*16 24: mosaic size 24*24 32: mosaic size 32*32
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=getPrivacyMasking&channel=1&offset=2&limit =5		

Response	<pre> total=5 PrivacyMasking[0].Index=0 PrivacyMasking[0].Name=Privacy Masking1 PrivacyMasking[0].Enable=1 PrivacyMasking[0].ShapeType=Polygon PrivacyMasking[0].Polygon[0][0]= 0 PrivacyMasking[0].Polygon[0][1]=0 PrivacyMasking[0].Polygon[1][0]=128 PrivacyMasking[0].Polygon[1][1]=128 PrivacyMasking[0].Polygon[2][0]=256 PrivacyMasking[0].Polygon[2][1]=200 ... PrivacyMasking[0].Color[0]=128, PrivacyMasking[0].Color[1]=128, PrivacyMasking[0].Color[2]=128, PrivacyMasking[0].Color[3]=255, PrivacyMasking[0].Mosaic=8, PrivacyMasking[1].Index =1, PrivacyMasking[1].Enable=1, ... </pre>
----------	--

## 4.5.20 Set Privacy Masking

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=setPrivacyMasking		
Method	GET		
Description	Set privacy masking		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
PrivacyMasking	object	R	Privacy masking config
+Index	int	R	Index of PrivacyMasking which starts from 0
+Name	string	O	PrivacyMasking name
+Enable	int	R	1: enable; 0: disable
+ShapeType	string	R	Shape type, can be: "Rect", "Polygon"
+Rect	Array<int>	RO	It is valid if ShapeType is Rect , top left and bottom right point, each point has x and y value, ordinate remap to 0 — 8192. mandatory when ShapeType is "Rect"
+Polygon	Array<Array<int>>	RO	It is valid if ShapeType is Polygon the first array is point list, minimum item is 3, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192. mandatory when ShapeType is "Polygon"
+Color	Array<int>	RO	Mask color, RGBA color, 4 integer range is 0-255. mandatory when Mosaic is 0

+Mosaic	int	R	Mosaic type, Range is {0, 8, 16, 24, 32} 0: no mosaic 8: mosaic size 8*8 16: mosaic size 16*16 24: mosaic size 24*24 32: mosaic size 32*32
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=setPrivacyMasking&channel=2&PrivacyMasking.Index=3&PrivacyMasking.Name=Privacy%20Masking1&PrivacyMasking.Enable=1&PrivacyMasking.ShapeType=Rect&PrivacyMasking.Rect[0]=0&PrivacyMasking.Rect[1]=0&PrivacyMasking.Rect[2]=50&PrivacyMasking.Rect[3]=100&PrivacyMasking.Mosaic=8		
Response	OK		

### 4.5.21 Goto Privacy Masking

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=gotoPrivacyMasking		
Method	GET		
Description	Goto privacy masking		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
index	int	R	Index of PrivacyMasking which starts from 0
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=gotoPrivacyMasking&channel=2&index=3		
Response	OK		

### 4.5.22 Delete Privacy Masking

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=deletePrivacyMasking		
Method	GET		
Description	Delete privacy masking		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
index	int	R	Index of PrivacyMasking which starts from 0
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=deletePrivacyMasking&channel=2&index=3		
Response	OK		

### 4.5.23 Clear Privacy Masking

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=clearPrivacyMasking		
Method	GET		
Description	Clear privacy masking		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=clearPrivacyMasking&channel=2		
Response	OK		

### 4.5.24 Get Privacy Masking Rect

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=getRealRect		
Method	GET		
Description	Get privacy masking rect		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
index	int	R	Index of PrivacyMasking which starts from 0
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
Rect	Array<int>	R	Rect, top left and bottom right point, each point has x and y value, ordinate remap to 0 – 8192.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=getRealRect&channel=2&index=1		
Response	Rect[0]=0 Rect[1]=0 Rect[2]=50 Rect[3]=100		

### 4.5.25 [Config] SmartMotionDetect

- Get smart motion detection setting

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=SmartMotionDetect		
Method	GET		
Description	Get smart motion detection setting		
[ Request Params ] (None)			
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description

table	object	R	Config info table.
+SmartMotionDetect	Array<object>	R	PrivacyMasking config, each channel has one config object
++Enable	bool	R	Enable/Disable
++Sensitivity	string	R	Detection sensitivity, can be: "Low", "Middle", "High"
++ObjectTypes	object	R	Detection object type
+++Human	bool	R	Whether detect motion of human
+++Vehicle	bool	R	Whether detect motion of vehicle
[ Example ]			
Request	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=SmartMotionDetect		
Response	table.SmartMotionDetect[0].Enable=true table.SmartMotionDetect[0].Sensitivity=Middle table.SmartMotionDetect[0].ObjectTypes.Human=true table.SmartMotionDetect[0].ObjectTypes.Vehicle=true ...		

- Set smart motion detection setting

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set smart motion detection setting		
[ Request Params ] (key=value format )			
Name	Type	R/O	Param Description
SmartMotionDetect	Array<object>	R	Smart motion detection config, each channel has one config object
+Enable	bool	R	Enable/Disable
+Sensitivity	string	R	Detection sensitivity, can be: "Low", "Middle", "High"
+ObjectTypes	object	R	Detection object type
++Human	bool	R	Whether detect motion of human
++Vehicle	bool	R	Whether detect motion of vehicle
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&SmartMotionDetect[0].Enable=true&SmartMotionDetect[0].Sensitivity=Middle&SmartMotionDetect[0].ObjectTypes.Human=true&SmartMotionDetect[0].ObjectTypes.Vehicle=true		
Response	OK		

## 4.5.26 [Config] LAEConfig

- Get LAEConfig setting

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=LAEConfig		
Method	GET		
Description	Get LAE Configs.		

Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=LAEConfig
Success Return	table.LAEConfig.Enable=true table.LAEConfig.ManulEn=1 table.LAEConfig.YSIGMA =1 table.LAEConfig.Dethr =1 table.LAEConfig.STS =1 table.LAEConfig.GauFilter0=1 table.LAEConfig.GauFilter1=1 table.LAEConfig.GauFilter2=1 table.LAEConfig.UVGauFilter0=1 table.LAEConfig.UVGauFilter1=1 table.LAEConfig.UVGauFilter2=1 table.LAEConfig.UVSigma =1 table.LAEConfig.EEGauFilter0=1 table.LAEConfig.EEGauFilter1=1 table.LAEConfig.EEGauFilter2=1 table.LAEConfig.EEratio =1 table.LAEConfig.GEnable =1 table.LAEConfig.GainLevel =1 table.LAEConfig.DebugPin =1
Comment	

- Set LAEConfig setting

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set LAE Configs.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&LAEConfig.Enable=true&LAEConfig.ManulEn=1
Success Return	OK
Comment	

#### Appendix:

ParamName	ParamValue type	Description
table.LAEConfig.Enable	bool	turn on/off LAE fuction The following parameters will take effects when it's on
table.LAEConfig.ManulEn	integer	whether use manual mode, 1 for manual mode, 0 for alternative plan
table.LAEConfig.YSIGMA	integer	absolute strength of Y channel Grinding, The smaller the stronger(fuzzier)
table.LAEConfig.Dethr	integer	relative strength of Y channel Grinding, The smaller the stronger(fuzzier)
table.LAEConfig.STS	integer	Y channel STS

ParamName	ParamValue type	Description
table.LAEConfig.GauFilter0	integer	Y channel pre filter0, meet the conditions: GauFilter0 * 2 + GauFilter1 * 2 + GauFilter2 = 256
table.LAEConfig.GauFilter1	integer	Y channel pre filter1
table.LAEConfig.GauFilter2	integer	Y channel pre filter2
table.LAEConfig.UVGauFilter0	integer	UV channel pre filter0, meet the conditions: GauFilter0 * 2 + GauFilter1 * 2 + GauFilter2 = 256
table.LAEConfig.UVGauFilter1	integer	UV channel pre filter1
table.LAEConfig.UVGauFilter2	integer	UV channel pre filter2
table.LAEConfig.UVSigma	integer	absolute strength of UV channel Grinding, The smaller the stronger
table.LAEConfig.EEGauFilter0	integer	EE pre filter0, meet the conditions: GauFilter0 * 2 + GauFilter1 * 2 + GauFilter2 = 256
table.LAEConfig.EEGauFilter1	integer	EE pre filter1, extracts low-frequenc
table.LAEConfig.EEGauFilter2	integer	EE pre filter2
table.LAEConfig.EERatio	integer	strength of enhancement, the bigger the stronger
table.LAEConfig.GEnable	integer	Brightening switch, default false
table.LAEConfig.GainLevel	integer	it is used to choose paramas, as an alternative
table.LAEConfig.DebugPin	integer	debug option

#### 4.5.27 Enable/Disable all privacy masking covers

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=setPrivacyMaskingEnable		
Method	GET		
Description	Enable/Disable all privacy masking covers		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1
Enable	bool	R	true: Enable, false: Disable
[ Response Params ] ( OK )			
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=setPrivacyMaskingEnable&channel=2&Enable=true		
Response	OK		

#### 4.5.28 Get enable/disable state of all privacy masking covers

Syntax	http://<server>/cgi-bin/PrivacyMasking.cgi?action=getPrivacyMaskingEnable		
Method	GET		
Description	Get enable/disable state of all privacy masking covers		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
channel	int	R	video channel index which starts from 1

[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
Enable	bool	R	true: Enable, false: Disable
[ Example ]			
Request	http://192.168.1.108/cgi-bin/PrivacyMasking.cgi?action=getPrivacyMaskingEnable&channel=2		
Response	Enable=false		

## 4.6 System

### 4.6.1 General

- Get general config

Table 4-28

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=General
Method	GET
Description	Get general config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=General
Success Return	table.General.LocalNo=8, table.General.MachineID=20832748927, table.General.MachineName=Dahua001, table.General.MachineAddress=滨江区.江南大道.伟业路口, table.General.MachineGroup=交警一队, table.General.LockLoginEnable=true, table.General.CheckDuration=30, table.General.LockLoginTimes=3, table.General.LoginFailLockTime=1800, table.General.MaxOnlineTime=1800, table.General.LocalPolicy.LockLoginEnable=true, table.General.LocalPolicy.CheckDuration=30, table.General.LocalPolicy.LockLoginTimes=3, table.General.LocalPolicy.LoginFailLockTime=1800 table.General.ActivationTime=2000-01-01 00:00:00
Comment	—

- Set general config

Table 4-29

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue> ...]
Method	GET
Description	Set general config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&General.MachineName=MyIPC

Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
General.MachineName	string	Device name or serial number.
General.LocalNo	integer	Local number for remote controller
General.MachineAddress	string	Address machine places in
General.MachineGroup	string	Group machine belongs to

### 4.6.2 Get Current Time

Table 4-30

Syntax	http://<server>/cgi-bin/global.cgi?action=getCurrentTime
Method	GET
Description	Get current time.
Example	http://192.168.1.108/cgi-bin/global.cgi?action=getCurrentTime
Success Return	result = 2011-7-3 21:02:32
Comment	The time format is "Y-M-D H-m-S". It's not be effected by Locales. TimeFormat in SetLocalesConfig.

### 4.6.3 Set Current Time

Table 4-31

Syntax	http://<server>/cgi-bin/global.cgi?action=setCurrentTime&time=2011-7-3%2021:02:32
Method	GET
Description	Set current time.
Example	http://192.168.1.108/cgi-bin/global.cgi?action=setCurrentTime&time=2016-01-01%2021:02:32
Success Return	OK
Comment	The time format is "Y-M-D H-m-S". It's not be effected by Locales. TimeFormat in SetLocalesConfig.

### 4.6.4 Locales

- Get locales config

Table 4-32

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Locales
Method	GET
Description	Get locales config

Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Locales
Success Return	<pre>table.Locales.DSTEnable=false table.Locales.DSTEnd.Day=1 table.Locales.DSTEnd.Hour=0 table.Locales.DSTEnd.Minute=0 table.Locales.DSTEnd.Month=1 table.Locales.DSTEnd.Week=2 table.Locales.DSTEnd.Year=2011 table.Locales.DSTStart.Day=0 table.Locales.DSTStart.Hour=0 table.Locales.DSTStart.Minute=0 table.Locales.DSTStart.Month=1 table.Locales.DSTStart.Week=1 table.Locales.DSTStart.Year=2011 table.Locales.TimeFormat=yyyy-MM-dd HH:mm:ss</pre>
Comment	—

- Set locales config

Table 4-33

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set Locales config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Locales.DSTEnable=false
Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
Locales.DSTEnable	bool	Enable/Disable DST (daylight saving time)
Locales.DSTEnd.Day	integer	Range is [0—6] or [1—31] [0—6]: week day, 0 = Sunday, 6 = Saturday [1—31]: month day If Locales.DSTEnd.Week is 0, use month day, otherwise, use week day.
Locales.DSTEnd.Hour	integer	Range is [0—23]
Locales.DSTEnd.Minute	integer	Range is [0—59]
Locales.DSTEnd.Month	integer	Range is [1—12]
Locales.DSTEnd.Week	Integer	Range is {1, 2, 3, 4, -1, 0}. 0 = Use month day [1, 2, 3, 4, -1]: use week day. 1 = first week, 2 = second, 3 = third, 4 = fourth, -1 = last.
Locales.DSTEnd.Year	Integer	Range is [2000-2038]

ParamName	ParamValue type	Description
Locales.DSTStart.Day	Integer	Range is the same with items in Locales.DSTEnd. Locales.DSTStart table and Locales.DSTEnd table define the time range of DST.
Locales.DSTStart.Hour		
Locales.DSTStart.Minute		
Locales.DSTStart.Month		
Locales.DSTStart.Week		
Locales.DSTStart.Year		
Locales.TimeFormat	string	<p>Defines time format displayed in video time title. String form is: <i>year-month-day hour:mm:ss</i>. Position of <i>year</i>, <i>month</i> and <i>day</i> can be exchanged.</p> <p>Range of <i>year</i> is {yy, yyyy}  yy = year without century, yyyy = year with century.  Range of <i>month</i> is {M, MM, MMMM}  M = 1 for January, MM = 01 for January, MMMM = Jan for January  Range of <i>day</i> is {d, dd}  d = 1 for first day, dd = 01 for first day  Range of <i>hour</i> is {H, HH, h, hh}  H = 1 for 1:00, HH = 01 for 1:00, range is 0-23  h = 1 for 1:00, hh = 01 for 1:00, time range is 1-12</p> <p>Example:  yyyy-MM-dd HH:mm:ss or  MM-dd-yyyy HH:mm:ss or  dd-M-yy hh:mm:ss</p>

## 4.6.5 Get Language Capability

Table 4-34

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getLanguageCaps
Method	GET
Description	Get the list of supported languages.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getLanguageCaps
Success Return	Languages=SimpChinese,English,French
Comment	response is a string contains languages with comma separated. Languages include { English, SimpChinese, TradChinese, Italian, Spanish, Japanese, Russian, French, German }

## 4.6.6 Language

- Get language config

Table 4-35

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Language
Method	GET
Description	Get system language config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Language
Success Return	table.Language=SimpChinese
Comment	—

- Set language config

Table 4-36

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set system language config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Language=SimpChinese
Success Return	OK
Comment	NOTE: After changing language setting, system will automatically reboot!

**Appendix:**

ParamName	ParamValue type	Description
Language	string	The language range is get from interface in <u>GetLanguageCaps</u>

## 4.6.7 Get Client Access Filter

Table 4-37

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AccessFilter
Method	GET
Description	Get access filter config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=AccessFilter
Success Return	table.AccessFilter.Enable=false table.AccessFilter.TrustList[0]=10.6.10.23 table.AccessFilter.TrustList[1]=10.6.10.62 table.AccessFilter. <b>Type</b> =TrustList
Comment	Parameters in Response: <b>Type</b> : Range is {TrustList, BannedList}. If Trustlist is supported, Bannedlist is not supported. More information in Appendix.

**Appendix:**

ParamName	ParamValue type	Description
AccessFilter.BannedList[ <i>index</i> ]	string	Banned IP address list
AccessFilter.TrustList[ <i>index</i> ]	string	Trusted IP address list

ParamName	ParamValue type	Description
AccessFilter.Enable	bool	Enable/Disable access filter function
AccessFilter.Type	string	Range is {TrustList, BannedList}, TrustList: Trust list is used, banned list is not used. BannedList: Banned list is used, trust list is not used.

## 4.6.8 Auto Maintain

- Get auto maintain config

Table 4-38

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AutoMaintain
Method	GET
Description	Get Auto Maintain config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=AutoMaintain
Success Return	table.AutoMaintain.AutoRebootDay=3 table.AutoMaintain.AutoRebootHour=0 table.AutoMaintain.AutoRebootMinute=0 table.AutoMaintain.AutoShutdownDay=1 table.AutoMaintain.AutoShutdownHour=0 table.AutoMaintain.AutoShutdownMinute=0 table.AutoMaintain.AutoStartupDay=1 table.AutoMaintain.AutoStartupHour=2 table.AutoMaintain.AutoStartupMinute=0
Comment	—

- Set auto maintain config

Table 4-39

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set auto maintain config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&AutoMaintain.AutoRebootDay=7
Success Return	OK
Comment	—

### Appendix:

ParamName	ParamValue type	Description
-----------	-----------------	-------------

ParamName	ParamValue type	Description
AutoMaintain.AutoRebootDay	integer	Range is [-1—7]. Auto restart day. -1 = never auto restart 0—6 = Sunday—Saturday 7 = restart every day
AutoMaintain.AutoRebootHour	integer	Range is [0—23]. Auto restart hour
AutoMaintain.AutoRebootMinute	integer	Range is [0—59]. Auto restart minute
AutoMaintain.AutoShutdownDay	integer	auto reboot time
AutoMaintain.AutoShutdownHour		Range is same with AutoOpenDay, AutoOpenHour and AutoOpenMinute.
AutoMaintain.AutoShutdownMinute		
AutoMaintain.AutoStartUpDay	integer	Auto shutdown time.
AutoMaintain.AutoStartUpHour		Range is same with AutoOpenDay, AutoOpenHour, and AutoOpenMinute.
AutoMaintain.AutoStartUpMinute		

## 4.6.9 Holiday Management

- Get holiday config

Table 4-40

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Holiday
Method	GET
Description	Get holiday config for record or snap.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Holiday
Success Return	table.Holiday.MonthMask[0]=3 table.Holiday.MonthMask[1]=0 table.Holiday.MonthMask[2]=0 table.Holiday.MonthMask[3]=0 table.Holiday.MonthMask[4]=0 table.Holiday.MonthMask[5]=0 table.Holiday.MonthMask[6]=0 table.Holiday.MonthMask[7]=0 table.Holiday.MonthMask[8]=0 table.Holiday.MonthMask[9]=1610612739 table.Holiday.MonthMask[10]=0 table.Holiday.MonthMask[11]=0
Comment	—

- Set holiday config

Table 4-41

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET

Description	Set holiday config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Holiday.MonthMask[0]=3
Success Return	OK
Comment	Parameters in URL: In table below, <b>monthIndex</b> presents the index of a month. 0 presents January, 1 presents February, 11 presents December.

#### Appendix:

ParamName	ParamValue type	Description
Holiday.MonthMask[ <b>monthIndex</b> ]	integer	It is the mask of a month. Every bit present a day. For example, 0x0001 presents the first day of a month is holiday. 0x0002 presents the second day of a month is holiday, 0x0003 presents the first day and second day of a month is holiday.

### 4.6.10 Get Device Type

Table 4-42

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getDeviceType
Method	GET
Description	Get the device type displaying which is not the real type.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getDeviceType
Success Return	type=DVR
Comment	—

### 4.6.11 Get Hardware Version

Table 4-43

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getHardwareVersion
Method	GET
Description	Get the device hardware version.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getHardwareVersion
Success Return	version=1.00
Comment	—

### 4.6.12 Get Serial Number of Device

Table 4-44

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getSerialNo
Method	GET
Description	Get the device serial number.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getSerialNo
Success Return	sn=YZC0GZ05100020
Comment	—

### 4.6.13 Get Machine Name

Table 4-45

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getMachineName
Method	GET
Description	Get the device machine name.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getMachineName
Success Return	name=YZC0GZ05100020
Comment	—

### 4.6.14 Get System Information1

Table 4-46

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getSystemInfo
Method	GET
Description	Get the system information.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getSystemInfo
Success Return	serialNumber=PA1FQ15900207 deviceType=27 processor=ST7108
Comment	No longer maintained.

### 4.6.15 Get System Information2

Table 4-47

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getSystemInfoNew
Method	GET
Description	Get the system information.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getSystemInfoNew
Success Return	info.SyncTime.Strategy="PerLogin", info.2DCode="123456", info.TotalRunTime=123456 info.cameraNum=0

	info.cardReader=false info.flashID[0]=200 info.flashID[1]=209 info.flashID[2]=128 info.flashID[3]=149 info.flashID[4]=64 info.flashID[5]=127 info.flashID[6]=127 info.flashID[7]=200 info.hasRTC=true
Comment	—

## 4.6.16 Get Vendor Information

Table 4-48

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getVendor
Method	GET
Description	Get the vendor information.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getVendor
Success Return	vendor=TTT
Comment	—

## 4.6.17 Get Software Information

Table 4-49

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getSoftwareVersion
Method	GET
Description	Get the software information.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getSoftwareVersion
Success Return	version=2.212.0000.0.R,build:2013-11-14
Comment	—

## 4.6.18 Get Version of Onvif

Table 4-50

Syntax	http://<server>/cgi-bin/IntervideoManager.cgi?action=getVersion&Name=Onvif
Method	GET
Description	Get onvif version.
Example	http://192.168.1.108/cgi-bin/IntervideoManager.cgi?action=getVersion&Name=Onvif
Success	version=2.4.2

Return	
Comment	—

## 4.6.19 Get Version of HTTP API

Table 4-51

Syntax	http://<server>/cgi-bin/IntervideoManager.cgi?action=getVersion&Name=CGI
Method	GET
Description	Get CGI version.
Example	http://192.168.1.108/cgi-bin/IntervideoManager.cgi?action=getVersion&Name=CGI
Success Return	version=2.0.0
Comment	—

## 4.6.20 Get Device Class

Table 4-52

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getDeviceClass
Method	GET
Description	Get the device class.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getDeviceClass
Success Return	class=HDVR
Comment	—

## 4.6.21 Onvif Service Authorization

- Get config of Onvif service authorization

Table 4-53

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=UserGlobal
Method	GET
Description	Get user global config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=UserGlobal
Success Return	table.UserGlobal.OnvifLoginCheck=false
Comment	If "OnvifLoginCheck" is false, you can get Onvif service directly; if true, you should enter your ID/username and password.

- Set config of Onvif service authorization

Table 4-54

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&UserGlobal.OnvifLoginCheck=<flag>
--------	--

Method	GET
Description	Enable onvif login check or not.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&UserGlobal.OnvifLoginCheck=true
Success Return	OK
Comment	Parameters in URL: <b>flag</b> : range is {true, false}.

## 4.6.22 Backup of Config

Table 4-55

Syntax	http://<server>/cgi-bin/Config.backup?action=All
Method	GET
Description	Download all the settings of a device as a file named Config.Backup by default.
Example	http://192.168.1.108/cgi-bin/Config.backup?action=All
Success Return	<pre> HTTP/1.1 200 OK CONTENT-LENGTH: 743087 CONNECTION: close Content-type: application/binarytet-stream; charset=utf-8  {   "ATM" : {     "DataSource" : "RS232",     "DisplayPostion" : "lefttop",     "EncodeBlend" : true,     "PreviewBlend" : true,     "ProtocolAbility" : [ "POS" ],     "ProtocolName" : "ATM\POS",     "RecordChannels" : [ 0, 1, 2, 3 ]   }   ..... } </pre>
Comment	—

## 4.6.23 Restore the Config

Table 4-56

Syntax	http://<server>/cgi-bin/configManager.cgi?action=restore&names[0]=<xxx>&names[1]=<yyy>[&...]
Method	GET
Description	Restore config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=restore&names[0]=UPnP
Success	OK

Return	
Comment	Parameters in URL: <b>xxx</b> and <b>yyy</b> is config name which need to be restore

## 4.6.24 Restore except the Config

Table 4-57

Syntax	http://<server>/cgi-bin/configManager.cgi?action=restoreExcept&names[0]=<xxx>&names[1]=<yyy>[&...]
Method	GET
Description	Restore all config except several.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=restoreExcept&names[0]=UPnP
Success Return	OK
Comment	Parameters in URL: All the config file but <b>xxx</b> and <b>yyy</b> will be restored.

## 4.6.25 Reboot

Table 4-58

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=reboot[&delay=<paramValue>]
Method	GET
Description	Reboot the device.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=reboot
Success Return	OK
Comment	If successful, response OK. If fail, response Error.

## 4.6.26 Shutdown

Table 4-59

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=shutdown
Method	GET
Description	Shutdown the device.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=shutdown
Success Return	OK
Comment	If successful, response OK. If fail, response Error.

## 4.6.27 FactoryReset

Table 4-60

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=resetSystemEx[&type=<type>]
Method	GET
Description	Reset the configuration for the device to the factory default.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=resetSystemEx&type=0
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>Type</b>: integer, it can be 0 or 1.</p> <p>0 means all parameters are set to their factory default value;</p> <p>1 means all parameters are set to their factory default value except the specific parameters;</p> <p>The specific parameters of different device types are different. But it always contains the network settings and user settings. After resetting, the device is reachable on the same ip address as used before the reset.</p> <p>When the <b>Type</b> parameter is not present in the URL, the default value of the <b>Type</b> is 0.</p>

## 4.6.28 Get Tracing Code of Device

Table 4-61

Syntax	http://<server>/cgi-bin/magicBox.cgi?action=getTracingCode
Method	GET
Description	Get the tracing code of the device.
Example	http://192.168.1.108/cgi-bin/magicBox.cgi?action=getTracingCode
Success Return	<b>tc</b> =0001...1101
Comment	<p>Only system default admin can get it.</p> <p><b>tc</b> is a 64bit binary string.</p>

## 4.6.29 Add Camera to Specified Channel

<b>Request URL</b>	http://<server>/cgi-bin/LogicDeviceManager.cgi?action=addCameraByGroup			
<b>Method</b>	POST			
<b>Request Params</b> ( Json format in body )				
Parameter	Type	Required	Description	Example
+group	object[]	Yes	Add device groups in batches It is an array with the same elements as addCameraByDevice.	
++DeviceInfo	object	Yes	Device details	
+++RtspPort	int	No	RTSP port: Required when ProtocolType is not Private, 554 by default.	554
+++HttpPort	int	No	HTTP port: Required when ProtocolType is not	80

			Private, 80 by default.	
+++Port	int	No	TCP port: Required when ProtocolType is Private, 37777 by default.	37777
+++UserName	char[32]	Yes	Username	admin
+++Password	char[32]	Yes	Password, plain text	admin123
+++ProtocolType	char[128]	Yes	Protocol Type "Private", "Dahua2", "Dahua3", "General", "Onvif", "GB28181", "Ehome"	"Private"
+++Address	char[16]	Yes	Device address	172.11.1.161
++cameras	object[]	Yes	Video source information	
+++uniqueChannel	int	Yes	Unique channel number, starting from 1	1

#### Example

POST http://192.168.1.108/cgi-bin/LogicDeviceManager.cgi?action=addCameraByGroup

```
{
  "group":{
    "DeviceInfo":{
      "UserName":"admin",
      "Password":"admin123",
      "ProtocolType":"Private",
      "Port":37777,
      "Address":"172.11.1.161"
    },
    "cameras":{
      "uniqueChannel":1
    }
  }
}
```

Response Params ( OK in body )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
OK				

## 4.6.30 Illuminator

The following commands are applicable to non-intelligent illuminators.

### 4.6.30.1 Visible-light Illuminator

Request

<b>Template</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig&SignLight[0].onCycle=30
-----------------	--

<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format in URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
onCycle	UInt32	Yes	Range[0-100]	30
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&SignLight[0].onCycle=30				
Response				
<b>Parameter Format</b>	OK at body			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>	OK			

### 4.6.31 Volume Control

Request

<b>Template</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig&AudioOutputVolume[0]=20			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format in URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
AudioOutputVolume	UInt32[]	Yes	Each array element represents the volume value of a audio channel, and the range is [0-100]	30
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&AudioOutputVolume[0]=20				
Response				
<b>Parameter Format</b>	OK at body			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>	OK			

### 4.6.32 Acquiring All Available Resources

Customer requests to obtain camera information, such as MAC address and SN, through CGI commands.

<b>Request URL</b>	http://<server>/cgi-bin/api/LogicDeviceManager/getCameraAll
--------------------	---

<b>Method</b>	GET			
<b>Request Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
<b>Request Example</b>				
{ }				

<b>Response Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
camera	object[]	R		
+ControllID	char[128]	O	Control number. Keyboard users can use the number to locate channels to improve operation efficiency. Control number "0" is not available, which is reserved for special use.	"5001"
+Name	char[128]	O	Name	"Hangzhou"
+Enable	bool	O	Enable only when using DeviceID to add/delete devices. Do not enable it through DeviceInfo. true: Meaning trying to connect the device, but not necessarily successful. You can acquire the connection status through getCameraState. false: Do not connect the device. From user's point of view, it is the same as that the device is not added. This Enable field is only used for the case of adding a device through DeviceID. We keep this field for compatibility and recommend using the Enable in DeviceInfo. When acquiring the field, this Enable and the Enable in DeviceInfo are returned at the same time. When sending the field, you can only select one between this Enable and the Enable in DeviceInfo. We	true

			recommend sending the Enable in DeviceInfo.	
+DeviceInfo	DeviceInfo	O	Device details. Use the Enable in the structure to add/delete devices. True: Add devices. False: delete devices from the channel. A device can be added only when "DeviceID" is " " and Info is not NULL.	
++Enable	bool	O	The device is enabled or not.	true
++EncryptStream	int	O	Device stream encryption algorithm (custom field for NVR). (If other encryption algorithms are used, refer to the encryption type definition in Dahua code stream format 0x95). 0: No encryption by default. 7: Tianjin HUISUN encryption algorithm.	0
++Address	char[16]	O	Device IP address or domain name.	"10.6.5.10"
++Port	int	O	Port number.	37777
++usePreSecret	enumint8	O	Select whether to use NVR preset password to add cameras (the naming remains the same globally, so the case is not uniform here). enumint8{ 0: Use UserName/Password to connect devices (default). 1: When using a preset password to connect devices, UserName/Password is not required. 2: When using camera login password to connect devices, UserName/Password is not	1

			required. }	
++UserName	char[32]	O	Username	"admin"
++Password	char[32]	O	Password in plain text.	"admin"
++ProtocolType	enumchar[32] ]	O	The protocol type of the connected device. For the value range, see the ProtocolType field of the RemoteDevice configuration. enumchar[32]{ "Private" "Dahua2" "Dahua3" "General" "Onvif" "GB28181" "Ehome" "ICC": The special protocol for the central Intelligence platform. It has not been reviewed, therefore it is not allowed to add the protocol type in this way in the future. "DahuaDSS": The special protocol for the central Intelligence platform. It has not been reviewed, therefore it is not allowed to add the protocol type in this way in the future. }	"Private"
++VideoInput++ Channels	uint	O	Total number of video input channels, including analog and digital channels	16
++AudioInput++ Channels	uint	O	Total number of audio input channels	4
++DeviceClass	char[64]	O	Device type	"IPC"
++DeviceType	char[64]	O	Device model	"IPC-HF3300"
++HttpPort	int	O	HTTP port number	80
++HttpsPort	int	O	HTTPS port number (customized for BOSCH)	443
++RtspPort	int	O	RTSP port number	554
++Name	char[64]	O	Device name	"IPC1"
++MachineAddress	char[256]	O	Device deployment location	"Intersection of Weiye Road, Jiangnan Avenue, Binjiang

				District"
++SerialNo	char[48]	O	Device serial number	"DH12345678910"
++VendorAbbr	char[32]	O	Vendor abbreviation (optional) Vendor type: Dahua (for example).	"GY"
++HardID	char[64]	O	Hardware ID (optional) Customer's requirements without indicating the specific hardware type	"454"
++SoftwareVersion	char[64]	O	Software version (optional)	"2.420.0000006.0.R.150311"
++ActivationTime	char[24]	O	The date when starting using the device. Date format: "yyyy-MM-dd HH:mm:ss" "Year-Month-Day Hour: Minute: Second" Range {"2000-01-01 00:00:00", /*Minimum date*/ "2099-01-01 00:00:00", /*Maximum date*/}	"2000-01-01 00:00:00"
++NodeType	char[32]	O	Device node type (main or sub node). It is used when adding storage node (DataNodeManager.addDataNode) in N + M cluster mode.	
++Mac	char[]	O	MAC address	"08:00:20:0A:8C:6D"
++OEMVendor	char[]	O	OEM vendor type. During cloud update, for some vendors, the Vendor field is displayed as OEM, and OEMVendor is added to indicate the actual vendor information. OEMVendor is the same as Vendor by default. If OEMVendor is defined in the product definition, refer to the the product definition.	"SONY"
+DeviceID	char[128]	O	Device ID When DeviceID is not " ", ignore DeviceInfo and use the existing DeviceID information in the	"dev123"

			RemoteDevice configuration.	
+Type	enumchar[32] ]	O	Input channel type. enumchar[32]{ "Local": Local channel (physical channel). "Remote": Remote channel. "Reserved32": Reserve 32 channels (compatible with the device protocol of the second generation). "Output": Output channel. "Matrix": Analog matrix channel. "Compose": Composite channel. "Cascade": Optical fiber cascading channel. }	"Remote"
+VideoStream	enumchar[32] ]	O	Video stream. enumchar[32]{ "Main": Main stream "Extra1": Sub stream 1 "Extra2": Sub stream 2 "Extra3": Sub stream 3 "Auto": Automatically selects the appropriate stream. "Preview": Preview raw data stream. "None": No video stream (pure audio stream). }	"Main"
+Channel	uint	O	Channel number of the remote device (optional). DeviceID is unique.	0
+UniqueChannel	int	O	Unique channel number of uniform device number (required). You can obtain video/audio streams by specifying the channel number with 0x11/0xF4 second-generation protocol. Notes: When implementing any addressing function of DeviceID+Channel,	0

			<p>LogicDeviceManager.getCameraAll() must be implemented, and the device must support "DeviceID=Unique" unified numbering method for input channels For details, see explanation on "Unique" unified channel numbering of <b>Dahua Information Exchange Format</b>.</p> <p>For setCamera(), uniqueChannel-1 means that the device automatically assigns a channel number.</p>	
+DataNodeId	int	O	<p>Indicates the data node to which the global channel is currently allocated (N+0 cluster). Counted from 0. If it is -1 or when the field does not exist, it means that it is not allocated currently.</p>	1
+DataNodeChannel	int	O	<p>Indicates the local channel of data node to which the global channel is currently allocated (N+0 cluster). Counted from 0. If it is -1 or when the field does not exist, it means that it is not currently allocated.</p>	2
+AutoReset	bool	O	<p>Auto reset. Compatible process: If this parameter is not configured, the default value is false and the channel cannot be reset automatically. when there is no client to access the video stream (streaming client access count changes from non-zero to zero), if AutoReset=true, the device will actively enable the channel to be idle (that is,</p>	false

			the video source attribute Enable=false ).	
+VideoStandard	enumchar[32]	O	Current video standard enumchar[32]{ "PAL" "NTSC" "SECAM" }	"PAL"
+Label	enumchar[32]	O	Device label enumchar[32]{ "Indoor": Indoor "Outdoor": Outdoor }	"Indoor"
+ImageInfo	object	O	Device snapshot When adding a device, take a snapshot and save it to local computer.	
++FilePath	char[256]	R	Image path	"/var/local/ch1.jpg"
+PasswordGenType	enumchar[32]	O	Password generation method enumchar[32]{ "UserInput": Entered by users. "Auto": Automatically generated. }	"UserInput"

#### Response Example

```
{
  "camera":{
    "ControllID": "5001",
    "Name": "杭州",
    "Enable": true,
    "DeviceInfo":{
      "Enable": true,
      "EncryptStream": 0,
      "Address": "10.6.5.10",
      "Port": 37777,
      "usePreSecret": 1,
      "UserName": "admin",
      "Password": "admin",
      "ProtocolType": "Private",
      "VideoInputChannels": 16,
      "AudioInputChannels": 4,
      "DeviceClass": "IPC",
      "DeviceType": "IPC-HF3300",
      "HttpPort": 80,
      "HttpsPort": 443,
```

```

    "RtspPort": 554,
    "Name": "IPC1",
    "MachineAddress": "Intersection of Weiye Road, Jiangnan Avenue, Binjiang District"
    "SerialNo": "DH12345678910",
    "VendorAbbr": "GY",
    "HardID": "454",
    "SoftwareVersion": "2.420.0000006.0.R.150311",
    "ActivationTime": "2000-01-01 00:00:00",
    "NodeType": "",
    "Mac": "08:00:20:0A:8C:6D",
    "OEMVendor": "SONY"
  }
  "DeviceID": "dev123",
  "Type": "Remote",
  "VideoStream": "Main",
  "Channel": 0,
  "UniqueChannel": 0,
  "DataNodeid": 1,
  "DataNodeChannel": 2,
  "AutoReset": false,
  "VideoStandard": "PAL",
  "Label": "Indoor",
  "ImageInfo": {
    "FilePath": "/var/local/ch1.jpg"
  },
  "PasswordGenType": "UserInput"
},...{}
}

```

### 4.6.33 Subscribing for Device Online/Offline Status

Subscribe for device online/offline status

<b>Request URL</b>	http://<server>/cgi-bin/api/LogicDeviceManager/attachCameraState			
<b>Method</b>	POST			
<b>Request Params</b> (JSON format in body)				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int[]	R	Observed unique channel number. -1 means all Unique channels. If one of the array elements is -1, it will observe all channels.	[1, 2]
<b>Request Example</b>				
{				

```
"type": [1, 2]
}
```

**Response Params (multipart in body)**

Name	Type	R/O	Description	Example
state	object	R	New status of camera	
+channel	int32	R	Unique channel number (channel number with changed status)	1
+connectionState	enumchar[12]	R	Connection status. enumchar[12]{ "Connecting": Connecting "Connected": Connected "Unconnect": Unconnected "Empty": No information. The channel is not configured. Disable": The channel is configured but disabled (the protocol cannot be reflected, and delete it first). "Changed: Device information connected to the channel is changed. "Hibernation": Hibernation }	"Connected"

**Response Example**

```
HTTP/1.1 200 OK
Content-Type: multipart/x-mixed-replace; boundary=<boundary>
Connection: close

--<boundary>
Content-Type: application/json
Content-Length: <data length>

{
  "channel" : 1,
  "connectionState" : "Connected"
}
--<boundary>
Content-Type: text/plain
Content-Length: 11

Heartbeat
--<boundary>
```

## 4.7 Network

### 4.7.1 Get Network Interfaces

Table 4-62

Syntax	http://<server>/cgi-bin/netApp.cgi?action=getInterfaces
Method	GET
Description	Get all of the system network interfaces.
Example	http://192.168.1.108/cgi-bin/netApp.cgi?action=getInterfaces
Success Return	<pre>netInterface[0].Name=eth0 netInterface[0].Type=Normal netInterface[0].Valid=true ... netInterface[1].Name=3g netInterface[1].Type=Auto,WCDMA,TD-SCDMA,TD-LTE,FDD-LTE netInterface[1].Valid=true ...</pre>
Comment	<p>result item value:</p> <p><b>Name:</b> network interface name.  "eth0" - wired network interface  "eth2" - wireless network interface  "3G" - 3G network interface  "bond0" - bond network interface  "bond1" - bond network interface  Not listed in detail</p> <p><b>Type:</b> Network types that support settings  "Normal" — wired network  "Wireless" — wireless network  "Auto", "TD-SCDMA", "WCDMA", "CDMA1x", "EDGE", "EVDO", "TD-LTE", "FDD-LTE",  "NR" — 3G network types.</p> <p><b>Valid:</b> network interface is valid if netInterface[n].Valid is true.</p>

### 4.7.2 Network Basic Config

- Get network config

Table 4-63

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Network
Method	GET
Description	Get network basic config. The basic config contains basic network parameters (default interface, domain name, host name), and configuration of each network interface.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Network

Success Return	<pre>table.Network.DefaultInterface=eth0 table.Network.Domain=ttt table.Network.Hostname=badak table.Network.<b>interface</b>.DefaultGateway=10.7.0.1 table.Network.<b>interface</b>.DhcpEnable=false table.Network.<b>interface</b>.DnsServers[0]=221.123.33.228 table.Network.<b>interface</b>.DnsServers[1]=221.12.1.228 table.Network.<b>interface</b>.IPAddress=10.7.2.3 table.Network.<b>interface</b>.MTU=1500 table.Network.<b>interface</b>.PhysicalAddress=00:10:5c:f2:1c:b4 table.Network.<b>interface</b>.SubnetMask=255.255.0.0</pre>
Comment	<b>interface</b> in response is network interface name, such as eth0, eth2...

- Set network config

Table 4-64

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&amp;&lt;paramName&gt;=&lt;paramValue&gt;[&amp;&lt;paramName&gt;=&lt;paramValue&gt;...]</code>
Method	GET
Description	Set network basic config.
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;NetWork.Domain=ttt&amp;NetWork.eth0.DhcpEnable=true</code>
Success Return	OK
Comment	<b>interface</b> in below ParamName is network interface name, such as eth0 and eth2

#### Appendix:

ParamName	ParamValue type	Description
Network.DefaultInterface	string	Set default network interface when multiple interfaces exist. Range of interfaces is depends on <b>GetInterfaces</b> .
Network.Domain	string	Domain name.
Network.Hostname	string	Hostname and Domain compose a network address.
Network. <b>interface</b> .DefaultGateway	string	IP address.
Network. <b>interface</b> .DhcpEnable	bool	Enable/Disable DHCP.
Network. <b>interface</b> .DnsServers[0]	string	IP address of first DNS server.
Network. <b>interface</b> .DnsServers[1]	string	IP address of second DNS server.
Network. <b>interface</b> .IPAddress	string	Interface IP address.
Network. <b>interface</b> .MTU	integer	Interface MTU.
Network. <b>interface</b> .PhysicalAddress	string	MAC address of interface. HEX string in the form of: xx:xx:xx:xx:xx:xx. Range of x is [0-9, a-f, A-F] Example: 00:10:5c:f2:1c:b4

ParamName	ParamValue type	Description
		00:10:5C:F2:1C:B5
Network. <b>interface</b> .SubnetMask	string	Network mask string: In the form of x.x.x.x, range of x is [0-255] Example: 255.255.255.0
Network. <b>interface</b> .EnableDhcpReservedIP	bool	Enable/Disable when dhcp failed, it will continue sending dhcp request, if it was enabled.

### 4.7.3 PPPoE

- Get PPPoE config

Table 4-65

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=PPPoE
Method	GET
Description	Get PPPoE config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=PPPoE
Success Return	table.PPPoE.Enable=true, table.PPPoE.AuthType[0]=PAP, table.PPPoE.UserName=hZ150260, table.PPPoE.Password=Dont Know, table.PPPoE.Mode=Default, table.PPPoE.Eth=eth0, table.PPPoE.DefaultPPPoEInterface=ppp0 table.PPPoE.PPPOE1.Enable=true, table.PPPoE.PPPOE1.AuthType[0]=PAP, table.PPPoE.PPPOE1.UserName=hZ150260, table.PPPoE.PPPOE1.Password=Don't Know, table.PPPoE.PPPOE1.Mode=Custom, table.PPPoE.PPPOE1.Eth=eth1
Comment	—

- Set PPPoE config

Table 4-66

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set PPPoE config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&PPPoE.UserName=user1&PPPoE.Password=123456
Success Return	OK
Comment	—

## Appendix:

ParamName	ParamValue type	Description
PPPoE.Enable	bool	Enable/Disable PPPoE.
PPPoE.UserName	string	PPPoE user name.
PPPoE.Password	string	PPPoE user password.

### 4.7.4 DDNS

- Get DDNS config

Table 4-67

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=DDNS
Method	GET
Description	Get DDNS config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=DDNS
Success Return	table.DDNS[ <i>index</i> ].Address=www.ttt.com table.DDNS[ <i>index</i> ].Enable=true table.DDNS[ <i>index</i> ].HostName=www.ttt.com table.DDNS[ <i>index</i> ].KeepAlive=10 table.DDNS[ <i>index</i> ].Password=none table.DDNS[ <i>index</i> ].Port=5050 table.DDNS[ <i>index</i> ].Protocol= Quick DDNS table.DDNS[ <i>index</i> ].UserName=user1 table.DDNS[ <i>index</i> ].DefaultHostName.Enable=false table.DDNS[ <i>index</i> ].DefaultHostName.HostName=9002A9D77133.quickddns.com
Comment	<i>index</i> in response is the DDNS protocol table index, start from 0. The meaning of parameters can refer to SetDDNSConfig chapter.

- Set DDNS config

Table 4-68

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue> ...]
Method	GET
Description	Set DDNS config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&DDNS[0].Address=www.ttt.com&DDNS[0].Enable=true
Success Return	OK
Comment	<i>index</i> in below ParamName is the DDNS protocol table index, start from 0.

## Appendix:

ParamName	ParamValue type	Description
DDNS[ <i>index</i> ].Address	string	DDNS server IP address or name.

ParamName	ParamValue type	Description
DDNS[ <i>index</i> ].Enable	bool	Multiple DDNS hostname can be configured, but Only one hostname can be enabled, others should be disabled.
DDNS[ <i>index</i> ].HostName	String	Hostname of this device.
DDNS[ <i>index</i> ].KeepAlive	integer	Range is [1—65535]. Unit is minutes.
DDNS[ <i>index</i> ].Password	string	DDNS user password
DDNS[ <i>index</i> ].Port	integer	Range is [1—65535]. Port of DDNS server
DDNS[ <i>index</i> ].Protocol	string	DDNS protocol type. Range is { "PRIVATE DDNS": 大华一代 "Private DDNS": 大华二代(使用 quickddns 域名) "CN99 DDNS" "NO-IP DDNS" "Dyndns DDNS" "LUPUS DDNS" "Oray DDNS" "CP Plus DDNS" "Ipplus DDNS" "Q-See DDNS" "SYSDNS DDNS" "Videotrend DDNS" "CEPSA DDNS" "G4IP DDNS" "HOSS DDNS" "Intelbras DDNS" "HSY DDNS" "Flir DDNS" "Private P2P DDNS": 大华 P2P 设备注册 "Mivigilante DDNS": 西班牙 DDNS "ByDemes DDNS" "DYNDNS OLD DDNS": 针对存储 dyndns 协议 "WATASHI DDNS" "CNB DDNS": 韩国 CNB DDNS "Peoplefu DDNS": 泰国 PeopleFu DDNS "MINT DDNS" "SPECO DDNS" "SHANY DDNS" "WITHCCTV DDNS": "AMCREST DDNS" "SONIC DDNS": 印度松下 DDNS "KBVISION DDNS": 越南 kbvision "BOSCH DDNS": 博世 DDNS}.
DDNS[ <i>index</i> ].UserName	string	DDNS user name

ParamName	ParamValue type	Description
DDNS[ <i>index</i> ].DefaultHostName.Enable	bool	Only protocol is in range {"Private DDNS", "DHDDNS", "QUICK DDNS"}, it effects. true : use the DefaultHostName.HostName false: use the HostName
DDNS[ <i>index</i> ].DefaultHostName.HostName	string	The default hostname. It cannot be modified.

## 4.7.5 Email

- Get email config

Table 4-69

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Email
Method	GET
Description	Get Email config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Email
Success Return	table.Email.Address=www.ttt.com table.Email.Anonymous=true table.Email.AttachEnable=true table.Email.Enable=true table.Email.HealthReport.Enable=false table.Email.HealthReport.Interval=61 table.Email.Password=123456 table.Email.Port=26 table.Email.Receivers[0]=x@ttt.com table.Email.Receivers[1]=y@ttt.com table.Email.Receivers[2]=z@ttt.com table.Email.SendAddress=x@ttt.com table.Email.SslEnable=false table.Email.TlsEnable=false table.Email.Authentication=false table.Email.OnlyAttachment=false table.Email.SendInterv=10 table.Email.Title=DVRMessage table.Email.UserName=anonymity
Comment	—

- Set email config

Table 4-70

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set Email config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Email.Address=mail.ttt.com&Email.Anonymous=false

Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
Email.Address	string	SMTP server IP address or name.
Email.Anonymous	bool	Enable/Disable anonymous email.
Email.AttachEnable	bool	Enable/Disable email attachment
Email.AttachmentEnable	bool	Enable/Disable email attachment
Email.Enable	bool	Enable/Disable email function
Email.HealthReport.Enable	bool	Enable/Disable report device status by email.
Email.HealthReport.Interval	integer	Range is [30-1440]. Unit is minutes
Email.Password	string	User password of email account.
Email.Port	integer	Range is [1-65535]
Email.Receivers[0]	string	Email addresses of 3 receivers.
Email.Receivers[1]	string	
Email.Receivers[2]	string	
Email.SendAddress	string	Sender email address.
Email.SslEnable	bool	True: enable SSL email.
Email.Title	string	Title of email.
Email.UserName	string	User name of email account.

### 4.7.6 Wlan

- Get Wlan config

Table 4-71

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Wlan
Method	GET
Description	Get Wlan config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Wlan
Success Return	table.Wlan.eth2.Enable=true table.Wlan.eth2.Encryption=off table.Wlan.eth2.KeyFlag=false table.Wlan.eth2.KeyID=0 table.Wlan.eth2.KeyType=Hex table.Wlan.eth2.Keys[0]=password1 table.Wlan.eth2.Keys[1]=password2 table.Wlan.eth2.Keys[2]=password3 table.Wlan.eth2.Keys[3]=password4 table.Wlan.eth2.LinkMode=Auto table.Wlan.eth2.SSID=ttt
Comment	—

- Set Wlan config

Table 4-72

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set WLAN config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&WLAN.eth2.Enable=true&WLAN.eth2.KeyType=Hex
Success Return	OK
Comment	In below ParamName, <b>interface</b> is name of wireless interface.

**Appendix:**

ParamName	ParamValue type	Description
WLAN. <b>interface</b> .Enable	bool	True: Enable WLAN on this interface.
WLAN. <b>interface</b> .Encryption	string	Range is {Off, On, WEP64Bits, WEP128Bits, WPA-PSK-TKIP, WPA-PSK-CCMP} Encryption mode.
WLAN. <b>interface</b> .KeyFlag	bool	true: key is configured.
WLAN. <b>interface</b> .KeyID	integer	Range is [0—3] Indicates which key is used. 0: WLAN. <b>interface</b> .Keys[0] is used.
WLAN. <b>interface</b> .KeyType	string	Range is {Hex, ASCII}
WLAN. <b>interface</b> .Keys[0]	string	For ASCII key type: 64bits encryption key length is 5, 128bits encryption key length is 13, consists of [0—9, a—z, A—Z]
WLAN. <b>interface</b> .Keys[1]	string	
WLAN. <b>interface</b> .Keys[2]	string	
WLAN. <b>interface</b> .Keys[3]	string	For HEX key type: 64bits encryption key length is 10, 128bits encryption key length is 26, consists of [0—9, a—z, A—Z]
WLAN. <b>interface</b> .LinkMode	string	Range is {Auto, Ad-hoc, and Infrastructure}. Auto – select suitable mode automatically. Ad-hoc – Device with wireless network adapter can connect to each other without Access Point. Infrastructure – Integrate wire and wireless LAN together to share network resource, access point is need in this mode.
WLAN. <b>interface</b> .SSID	string	SSID.

**4.7.7 Scan Wlan Devices**

Table 4-73

Syntax	http://<server>/cgi-bin/wlan.cgi?action=scanWlanDevices<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Search Wi-Fi device information.

Example	http://192.168.1.108/cgi-bin/wlan.cgi?action=scanWlanDevices&SSID=xia_yuguo 13098 Internet
Success Return	found=1 wlanDevice[0].ApConnected=0 wlanDevice[0].ApMaxBitRate=54000000 wlanDevice[0].ApNetWorkType=255 wlanDevice[0].AuthMode=7 wlanDevice[0].BSSID=28:2c:b2:5c:de:36 wlanDevice[0].EncrAlgr=3 wlanDevice[0].LinkMode=0 wlanDevice[0].LinkQuality=31 wlanDevice[0].RSSIQuality=0 wlanDevice[0].SSID=xia_yuguo 13098 Internet
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
<b>SSID</b>	string	Specified SSID, if not include any SSID, all Wi-Fi information will be searched and displayed.

### 4.7.8 UPnP

- Get UPnP config

Table 4-74

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=UPnP
Method	GET
Description	Get UPnP config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=UPnP
Success Return	table.UPnP.Enable=true table.UPnP.MapTable[ <i>index</i> ].Enable=true table.UPnP.MapTable[ <i>index</i> ].InnerPort=80 table.UPnP.MapTable[ <i>index</i> ].OuterPort=8080 table.UPnP.MapTable[ <i>index</i> ].Protocol=TCP table.UPnP.MapTable[ <i>index</i> ].ServiceName=HTTP
Comment	<b>index</b> in response is the UPnP map table index, start from 0.

- Set UPnP config

Table 4-75

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set UPnP config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&UPnP.Enable=true&UPnP.MapTable[0].Protocol=TCP

Success Return	OK
Comment	index in below ParamName is UPnP map table index, range is [0-255]

#### Appendix:

ParamName	ParamValue type	Description
UPnP.Enable	bool	Enable/Disable UPnP feature.
UPnP.MapTable[ <i>index</i> ].Enable	bool	Enable/Disable this UPnP map.
UPnP.MapTable[ <i>index</i> ].InnerPort	integer	Range is [1—65535]. Inner port number
UPnP.MapTable[ <i>index</i> ].OuterPort	integer	Range is [1—65535]. Outer port number.
UPnP.MapTable[ <i>index</i> ].Protocol	string	Range is {TCP, UDP}
UPnP.MapTable[ <i>index</i> ].ServiceName	string	User defined UPnP service name.

### 4.7.9 Get UPnP Status

Table 4-76

Syntax	http://<server>/cgi-bin/netApp.cgi?action=getUPnPStatus
Method	GET
Description	Get UPnP Status.
Example	http://192.168.1.108/cgi-bin/netApp.cgi?action=getUPnPStatus
Success Return	status.InnerAddress=0.0.0.0 status.OuterAddress=0.0.0.0 status.PortMapStatus[0]=Failed status.PortMapStatus[1]=Failed status.PortMapStatus[2]=Failed status.PortMapStatus[3]=Failed status.Status=Unknown status.Working=false
Comment	—

### 4.7.10 NTP

- Get NTP config

Table 4-77

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=NTP
Method	GET
Description	Get NTP config
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=NTP
Success Return	table.NTP.Address=clock.isc.org table.NTP.Enable=false table.NTP.Port=38 table.NTP.TimeZone=9

	table.NTP.UpdatePeriod=31 table.NTP.ServerList[0].Enable=true, table.NTP.ServerList[0].Address=192.168.1.108, table.NTP.ServerList[0].Port=123 table.NTP.Tolerance=5, table.NTP.TimeZoneDesc=Beijing
Comment	—

- Set NTP config

Table 4-78

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set NTP config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&NTP.Address=time.ttt.com&NTP.Enable=true
Success Return	OK
Comment	—

#### 4.7.11 RTSP

- Get RTSP config

Table 4-79

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=RTSP
Method	GET
Description	Get RTSP config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RTSP
Success Return	table.RTSP.Enable=true table.RTSP.Port=554 table.RTSP.RTP.EndPort=40000 table.RTSP.RTP.StartPort=20000
Comment	—

- Set RTSP config

Table 4-80

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set RTSP config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&RTSP.Enable=true&RTSP.Port=554
Success Return	OK

Comment	—
---------	---

#### Appendix:

ParamName	ParamValue type	Description
RTSP.Enable	bool	Enable/Disable RTSP.
RTSP.Port	integer	RTSP port.
RTSP.RTP.StartPort	integer	RTP start port.
RTSP.RTP.EndPort	integer	RTP end port.

## 4.7.12 Alarm Server

- Get alarm server config

Table 4-81

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AlarmServer
Method	GET
Description	Get alarm server config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=AlarmServer
Success Return	table.AlarmServer.Address=10.7.8.9 table.AlarmServer.Enable=false table.AlarmServer.Password= table.AlarmServer.Port=8888 table.AlarmServer.Protocol=ttt table.AlarmServer.ReportTime=02:00:00 table.AlarmServer.ReportWeekDay=2 table.AlarmServer.UserName=admin
Comment	—

- Set alarm server config

Table 4-82

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set Alarm Server config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&AlarmServer.Address=as.ttt.com&AlarmServer.Enable=false
Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
AlarmServer.Address	string	Alarm server IP address or name.
AlarmServer.Enable	bool	Enable/Disable Alarm server.

ParamName	ParamValue type	Description
AlarmServer.Port	integer	Range is [1—65535]. Port of Alarm server.

## 4.7.13 Device Network Packet Capture

### • Enabling Device Network Packet Capture

Enabling Device Network Packet Capture

<b>Syntax</b>	http://<server>/cgi-bin/NetworkSnifferManager.cgi?action=start			
<b>Method</b>	GET			
<b>Request Params</b>	Key=Value			
<b>Parameter</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
networkCard	string	O	Network adapter name	"eth0"
path	string	O	When the packet capture storage file name is "", the file is saved in the default configuration directory (specified by NetworkSniffer. DefaultSaveDirectory and NetworkSniffer. FileNamedRule).	"/mnt/dvr/sda0/2010/8/11/dav/xxx.pcap"
saveType	string	O	Types of packet capture storage file:  1. "Wireshark/Tcpdump": See the Wireshark packet capture file type.  2. Remotecap: Streaming remote packet capture, that is, calling back data through the attachSnifferCall interface.	"Wireshark/Tcpdump"
filter	string	O	Packet capture filtering rules:  Optional rules: 1) Target IP	"host 10.9.8.2 and port 8080 and tcp"

			2) Target port number 3) Protocol type	
captureTime	integer	O	Packet capture timeout period, in seconds. If this field does not exist, no timeout by default.	1024
currentFileSize	integer	O	Specify the maximum size of this packet capture, in kb.	1024
filterEx	object	O	High priority filtering rules. If this option exists, the filtering rules in filter are not adopted	
+CapInfo	object[]	O	Filter the arrays of quintuples. The number of arrays should not exceed 8.	
++SrcIP	string	O	Source IP address	"127.0.0.1"
++SrcPort	string	O	Source port address	1234
++DstIP	string	O	Destination IP address	"127.0.0.1"
++DstPort	string	O	Destination port address	5678
++Protocol	integer	O	tcp or udp	"tcp"
+CapType		O	Filtering types:  0: Use the Or-Set of all the above quintuples as the filtering condition.  1: Use the NOR of all the above quintuples as the filtering condition	0

#### Example

GET

http://<server>/cgi-bin/NetworkSnifferManager.cgi?action=start&networkCard="eth0"&path="/mnt/dvr/sda0/2010/8/11/dav/xxx.pcap"&saveType="Wireshark/Tcpdump"&filter="host 10.9.8.2 and port 8080 and tcp"&captureTime=1024&currentFileSize=1024&filterEx.CapInfo[0].SrcIP="127.0.0.1"&filterEx.CapInfo[0].SrcPort=1234&filterEx.CapInfo[0].DstIP="127.0.0.1"&filterEx.CapInfo[0].DstPort=5678&filterEx.CapIn

fo[0].Protocol="tcp"&filterEx.CapType=0

Response Params		Key=Value		
Parameter	Type	R/O	Description	Example
networkSnifferID	integer	O	Packet capture ID, which is generated when enabling packet capture and uniquely identifies a packet capture.	1387
<b>Example</b>				
networkSnifferID=1387				

- **Disabling Device Network Packet Capture**

Disabling Device Network Packet Capture

Syntax		http://<server>/cgi-bin/NetworkSnifferManager.cgi?action=stop		
Method		GET		
Request Params		Key=Value		
Parameter	Type	R/O	Description	Example
networkSnifferID	integer	O	Packet capture ID, which is generated when enabling packet capture and uniquely identifies a packet capture.	1387
<b>Example</b>				
GET http://<server>/cgi-bin/NetworkSnifferManager.cgi?action=stop&networkSnifferID=1387				

Response Params		OK		
Parameter	Type	R/O	Description	Example
	string	O	Result of disabling packet capture	"OK"
<b>Example</b>				
OK				

## 4.8 Motion Detection

### 4.8.1 Motion Detection Settings

- Get motion detect config

Table 4-83

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=getConfig&amp;name=MotionDetect</code>
Method	GET
Description	Motion detect config of a video channel contains Enable, MotionDetectWindow and EventHandler.
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&amp;name=MotionDetect</code>
Success Return	<pre> table.MotionDetect[0].Enable=false table.MotionDetect[0].EventHandler.AlarmOut=1 table.MotionDetect[0].EventHandler.AlarmOutChannels[0]=0 table.MotionDetect[0].EventHandler.AlarmOutEnable=true table.MotionDetect[0].EventHandler.AlarmOutLatch=10 table.MotionDetect[0].EventHandler.BeepEnable=false table.MotionDetect[0].EventHandler.Dejitter=5 table.MotionDetect[0].EventHandler.Delay=0 table.MotionDetect[0].EventHandler.ExAlarmOut=1 table.MotionDetect[0].EventHandler.ExAlarmOutChannels[0]=0 table.MotionDetect[0].EventHandler.ExAlarmOutEnable=false table.MotionDetect[0].EventHandler.FlashEnable=false table.MotionDetect[0].EventHandler.FlashLatch=10 table.MotionDetect[0].EventHandler.LogEnable=true table.MotionDetect[0].EventHandler.MailEnable=false table.MotionDetect[0].EventHandler.Matrix=1 table.MotionDetect[0].EventHandler.MatrixChannels[0]=0 table.MotionDetect[0].EventHandler.MatrixEnable=false table.MotionDetect[0].EventHandler.MessageEnable=false table.MotionDetect[0].EventHandler.PtzLink[0][0]=None table.MotionDetect[0].EventHandler.PtzLink[0][1]=1 table.MotionDetect[0].EventHandler.PtzLinkEnable=false table.MotionDetect[0].EventHandler.Record=1 table.MotionDetect[0].EventHandler.RecordChannels[0]=0 table.MotionDetect[0].EventHandler.RecordEnable=true table.MotionDetect[0].EventHandler.RecordLatch=10 table.MotionDetect[0].EventHandler.Snapshot=1 table.MotionDetect[0].EventHandler.SnapshotChannels[0]=0 table.MotionDetect[0].EventHandler.SnapshotEnable=false table.MotionDetect[0].EventHandler.TimeSection[0][0]=1 00:00:00-23:59:59 table.MotionDetect[0].EventHandler.TimeSection[0][1]=0 00:00:00-23:59:59 table.MotionDetect[0].EventHandler.TimeSection[0][2]=0 00:00:00-23:59:59 table.MotionDetect[0].EventHandler.TimeSection[0][3]=0 00:00:00-23:59:59 </pre>



table.MotionDetect[0].MotionDetectWindow[0].Region[1]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[2]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[3]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[4]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[5]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[6]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[7]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[8]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[9]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[10]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[11]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[12]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[13]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[14]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[15]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[16]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Region[17]=4194303  
table.MotionDetect[0].MotionDetectWindow[0].Sensitive=60  
table.MotionDetect[0].MotionDetectWindow[0].Threshold=5  
table.MotionDetect[0].MotionDetectWindow[0].Window[0]=0  
table.MotionDetect[0].MotionDetectWindow[0].Window[1]=0  
table.MotionDetect[0].MotionDetectWindow[0].Window[2]=8191  
table.MotionDetect[0].MotionDetectWindow[0].Window[3]=8191  
table.MotionDetect[0].MotionDetectWindow[1].Id=1  
table.MotionDetect[0].MotionDetectWindow[1].Name=Region2  
table.MotionDetect[0].MotionDetectWindow[1].Region[0]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[1]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[2]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[3]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[4]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[5]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[6]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[7]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[8]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[9]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[10]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[11]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[12]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[13]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[14]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[15]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[16]=0  
table.MotionDetect[0].MotionDetectWindow[1].Region[17]=0  
table.MotionDetect[0].MotionDetectWindow[1].Sensitive=60  
table.MotionDetect[0].MotionDetectWindow[1].Threshold=5  
table.MotionDetect[0].MotionDetectWindow[1].Window[0]=0  
table.MotionDetect[0].MotionDetectWindow[1].Window[1]=0

table.MotionDetect[0].MotionDetectWindow[1].Window[2]=0  
table.MotionDetect[0].MotionDetectWindow[1].Window[3]=0  
table.MotionDetect[0].MotionDetectWindow[2].Id=2  
table.MotionDetect[0].MotionDetectWindow[2].Name=Region3  
table.MotionDetect[0].MotionDetectWindow[2].Region[0]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[1]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[2]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[3]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[4]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[5]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[6]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[7]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[8]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[9]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[10]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[11]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[12]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[13]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[14]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[15]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[16]=0  
table.MotionDetect[0].MotionDetectWindow[2].Region[17]=0  
table.MotionDetect[0].MotionDetectWindow[2].Sensitive=60  
table.MotionDetect[0].MotionDetectWindow[2].Threshold=5  
table.MotionDetect[0].MotionDetectWindow[2].Window[0]=0  
table.MotionDetect[0].MotionDetectWindow[2].Window[1]=0  
table.MotionDetect[0].MotionDetectWindow[2].Window[2]=0  
table.MotionDetect[0].MotionDetectWindow[2].Window[3]=0  
table.MotionDetect[0].MotionDetectWindow[3].Id=3  
table.MotionDetect[0].MotionDetectWindow[3].Name=Region4  
table.MotionDetect[0].MotionDetectWindow[3].Region[0]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[1]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[2]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[3]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[4]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[5]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[6]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[7]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[8]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[9]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[10]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[11]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[12]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[13]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[14]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[15]=0  
table.MotionDetect[0].MotionDetectWindow[3].Region[16]=0

	<pre>table.MotionDetect[0].MotionDetectWindow[3].Region[17]=0 table.MotionDetect[0].MotionDetectWindow[3].Sensitive=60 table.MotionDetect[0].MotionDetectWindow[3].Threshold=5 table.MotionDetect[0].MotionDetectWindow[3].Window[0]=0 table.MotionDetect[0].MotionDetectWindow[3].Window[1]=0 table.MotionDetect[0].MotionDetectWindow[3].Window[2]=0 table.MotionDetect[0].MotionDetectWindow[3].Window[3]=0 table.MotionDetect[0].OsdTwinkleEnable=false table.MotionDetect[0].PirMotionLevel=3</pre>
Comment	—

- Set motion detect config

Table 4-84

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;paramName&gt;=&lt;paramValue&gt;[&amp;&lt;paramName&gt;=&lt;paramValue&gt;...]</code>
Method	GET
Description	Set motion detection config.
Example	<p>Enable motion detection:  <code>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;MotionDetect[0].Enable=true</code></p> <p>Set motion detection regions:  <code>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;MotionDetect[0].MotionDetectWindow[0].Region[0]=1&amp;MotionDetect[0].MotionDetectWindow[0].Region[1]=1&amp;MotionDetect[0].MotionDetectWindow[0].Region[2]=1&amp;MotionDetect[0].MotionDetectWindow[0].Region[3]=1&amp;MotionDetect[0].MotionDetectWindow[0].Region[4]=1&amp;MotionDetect[0].DetectVersion=V3.0</code></p>
Success Return	OK
Comment	<p>In table below,</p> <p><b>head</b> = MotionDetect[<b>ChannelNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).</p> <p><b>LineNum</b>  Index of region, region is divided into lines and each line has several blocks, a line is described by a 32 bit integer, a bit for a block.</p> <p>0=Line 1  1=Line 2  ...</p> <p><b>WinNum</b>  Index of detect window, there are 4 detect windows at present. Each window is divided into 18 lines and 22 blocks per line.</p> <p><b>Notice</b>: When setting "MotionDetect[<b>ChannelNo</b>].MotionDetectWindow[<b>WinNum</b>].Region", you need to contain the parameter "MotionDetect[<b>ChannelNo</b>].DetectVersion=V3.0" along.</p>

**Appendix:**

ParamName	ParamValue type	Description
<b>head.Enable</b>	bool	Enable/Disable motion detect feature in a channel.
<b>head.PtzManualEnable</b>	bool	Enable/Disable manual control Ptz, only for Hahua devices
<b>head.OsdTwinkleEnable</b>	bool	Osd Twinkle Enable default : false
<b>head.Row</b>	integer	motion detect row number
<b>head.Col</b>	integer	motion detect colum number
<b>head.Level</b>	integer	sensitive ,range 1-6
<b>head.TimeDivide. TimeSchedule</b>	array<array<string>>	TimeSchedule
<b>head.TimeDivide.Level</b>	integer	sensitive in timesection, different with global sensitive.
<b>head.DetectVersion</b>	string	detect version
<b>head.PirMotionEnable</b>	bool	Customized Pir Motion Enable
<b>head.PirMotionLevel</b>	integer	Customized Pir Motion sensitive level
<b>head.VolumeRatio</b>	integer	Area occupancy ratio
<b>head.SubRatio</b>	integer	SubRatio
<b>head.Region</b>	array<integer>	Region
<b>head.EventHandler</b>	—	Setting of EventHandler is described in <b><u>SetEventHandler.</u></b>
<b>head.MotionDetectWindow [WinNum].Id</b>	integer	It is the Id of a detect window.
<b>head.MotionDetectWindow [WinNum].Name</b>	string	It is the name of a detect window.
<b>head.MotionDetectWindow[WinNum].Sensitivity</b>	integer	Range is [0—100]. Sensitivity of motion detection. It presents more sensitive if the value is larger.
<b>head.MotionDetectWindow[WinNum].Threshold</b>	integer	Range is [0—100]. It presents the threshold value when trigger motion detect.
<b>head.MotionDetectWindow[WinNum].Region[LineNum]</b>	integer	It is similar with head.Region [LineNum]. Currently, a region is divided into 18 lines and 22 blocks per line. A bit describes a block in the line. Bit = 1: motion in this block is monitored. Example: MotionDetect [0].Region [0] = 4194303 (0x3FFFFFF): the 22 blocks in channel 0 line 0 is monitored. MotionDetect [0].Region [1] =0: the 22 blocks in channel 0 line 1 is not monitored. MotionDetect [0].Region [17] = 3: the left two blocks in the last line of channel 0 is monitored.

## 4.9 Event

### 4.9.1 Event Handler

- Get event handler config

Table 4-85

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=getConfig&amp;name=&lt;handlerName&gt;</code>
Method	GET
Description	Get event handler settings.
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&amp;name=Alarm[0].EventHandler</code>
Success Return	<pre> <b>handlerName</b>.EventHandler.AlarmOutChannels[0]=1 <b>handlerName</b>.EventHandler.AlarmOutChannels[1]=1 ... <b>handlerName</b>.EventHandler.AlarmOutEnable=false <b>handlerName</b>.EventHandler.AlarmOutLatch=10 <b>handlerName</b>.EventHandler.BeepEnable=true <b>handlerName</b>.EventHandler.Dejitter=0 <b>handlerName</b>.EventHandler.Delay=30 <b>handlerName</b>.EventHandler.LogEnable=true <b>handlerName</b>.EventHandler.MailEnable=true <b>handlerName</b>.EventHandler.PtzLink[0][0]=None <b>handlerName</b>.EventHandler.PtzLink[0][1]=0 <b>handlerName</b>.EventHandler.PtzLink[1][0]=None <b>handlerName</b>.EventHandler.PtzLink[1][1]=0 ... <b>handlerName</b>.EventHandler.PtzLinkEnable=false <b>handlerName</b>.EventHandler.RecordChannels[0]=1 <b>handlerName</b>.EventHandler.RecordChannels[1]=2 ... <b>handlerName</b>.EventHandler.RecordEnable=true <b>handlerName</b>.EventHandler.RecordLatch=10 <b>handlerName</b>.EventHandler.SnapshotChannels[0]=1 <b>handlerName</b>.EventHandler.SnapshotChannels[1]=1 ... <b>handlerName</b>.EventHandler.SnapshotEnable=false <b>handlerName</b>.EventHandler.SnapshotPeriod=3 <b>handlerName</b>.EventHandler.SnapshotTimes=0 <b>handlerName</b>.EventHandler.TimeSection[0][0]=1 01:00:00-24:00:00 <b>handlerName</b>.EventHandler.TimeSection[0][1]=1 01:00:00-24:00:00... ... <b>handlerName</b>.EventHandler.TimeSection[6][5]=1 01:00:00-24:00:00 <b>handlerName</b>.EventHandler.TipEnable=true <b>handlerName</b>.EventHandler.ExAlarmOutEnable=true </pre>

	<b>handlerName</b> .EventHandler.Enable=true, <b>handlerName</b> .EventHandler.FilckerLightType=WhiteLight, <b>handlerName</b> .EventHandler.LightLinkType=Filcker, <b>handlerName</b> .EventHandler.FilckerIntevalTime=5, <b>handlerName</b> .EventHandler.FilckerTimes=5, <b>handlerName</b> .EventHandler.LightDuration=10, <b>handlerName</b> .EventHandler.WhiteLightTimeSection=, <b>handlerName</b> .EventHandler.LightBright=50 <b>handlerName</b> .ExAlarmOutChannels[0] =2 <b>handlerName</b> .ExAlarmOutChannels[1]=3 ...
Comment	Parameters in URL: <b>handlerName</b> can be one of below four formats: <b>Alarm</b> [ChannelNo].EventHandler <b>MotionDetect</b> [ChannelNo].EventHandler <b>BlindDetect</b> [ChannelNo].EventHandler <b>LossDetect</b> [ChannelNo].EventHandler

- Set event handler config

Table 4-86

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Modify event handler settings.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Alarm[0].EventHandler.AlarmOutChannels[0]=1&Alarm[0].EventHandler.AlarmOutEnable=true
Success Return	OK
Comment	In below paramName, Meaning of <b>handlerName</b> is the same with GetEventHandler.

**Appendix:**

paramName	paramValue type	Description
<b>handlerName</b> .EventHandler.AlarmOutChannels[ <b>ch</b> ]	integer	<b>ch</b> is alarm out channel index which starts from 0. Range is {0, 1}, 0 — do not output alarm at alarm out channel <b>ch</b> 1 — output alarm at alarm out channel <b>ch</b>
<b>handlerName</b> .EventHandler.AlarmOutEnable	bool	Enable/Disable alarm out function.
<b>handlerName</b> .EventHandler.AlarmOutLatch	Integer	Range is [10—300]. Unit is seconds, indicates the time to output alarm after input alarm is cleared.
<b>handlerName</b> .EventHandler.BeepEnable	bool	Enable/Disable beep.

paramName	paramValue type	Description
<b>handlerName</b> .EventHandler.Dejitter	integer	Range is [0—255]. Alarm signal dejitter seconds. Alarm signal change during this period is ignored.
<b>handlerName</b> .EventHandler.Delay	integer	Range is [0—300]. Delay seconds before setting take effect.
<b>handlerName</b> .EventHandler.LogEnable	bool	Enable/Disable log for alarm.
<b>handlerName</b> .EventHandler.MailEnable	bool	Enable/Disable mail send for alarm.
<b>handlerName</b> .EventHandler.PtzLink[ch][0]	string	Range is {None, Preset, Tour, Pattern} This is PTZ action linked with events. ch is PTZ channel index which starts from 0.
<b>handlerName</b> .EventHandler.PtzLink[ch][1]	integer	This is the parameter of PtzLink[ch][0], If PtzLink[ch][0] is Preset: this is preset point. Tour: this is tour path number. Pattern: this is pattern number.
<b>handlerName</b> .EventHandler.PtzLinkEnable	Bool	Enable/Disable PTZ link.
<b>handlerName</b> .EventHandler.RecordChannels	array<integer>	record channel list
<b>handlerName</b> .EventHandler.RecordEnable	bool	Enable/Disable record function.
<b>handlerName</b> .EventHandler.RecordLatch	integer	Range is [10—300]. Unit is seconds, indicates the time to record after input alarm is cleared.
<b>handlerName</b> .EventHandler.SnapshotChannels	array<integer>	snapshot channel list
<b>handlerName</b> .EventHandler.SnapshotEnable	bool	Enable/Disable snapshot function.
<b>handlerName</b> .EventHandler.SnapshotPeriod	integer	Range is [0—255]. Frames between snapshots. 0 means continuously snapshot for every frame.
<b>handlerName</b> .EventHandler.SnapshotTimes	integer	Range is [0—65535] Snapshot times before stop, 0 means don't stop snapshot.

paramName	paramValue type	Description
<b>handlerName</b> .EventHandler.TimeSection[wd][ts]	String	<p>It's an effective time period for eventHandler everyday.  wd (week day) range is [0—6] (Sunday—Saturday)  ts (time section) range is [0-23], it's index of time section table.</p> <p>Format: <span style="float: right;">mask</span>  hh:mm:ss-hh:mm:ss  Mask: {0,1}, hh: [0—24], mm: [00—59], ss: [00—59]  Mask 0: this time section is not used.  Mask 1: this time section is used.</p> <p>Example:  TimeSection[1][0]=1  12:00:00-18:00:00  Means EventHandler is effective between 12:00:00 and 18:00:00 at Monday.</p>
<b>handlerName</b> .EventHandler.TipEnable	bool	Enable/Disable local message box tip.
<b>handlerName</b> .EventHandler.ExAlarmOutEnable	bool	Enable/Disable extend alarm out ability
<b>handlerName</b> .EventHandler.LightingLink.Enable	bool	Enable/Disable LightingLink
<b>handlerName</b> .EventHandler.LightingLink.FlickerLightType	String	Flicker Light Type range : "WhiteLight"
<b>handlerName</b> .EventHandler.LightingLink.LightingLinkType	string	range : ["Flicker", "KeepLighting"]
<b>handlerName</b> .EventHandler.LightingLink.FlickerIntervalTime	float	Flicker Interval Time unit: 0.1s
<b>handlerName</b> .EventHandler.LightingLink.FlickerTimes	integer	Flicker Times
<b>handlerName</b> .EventHandler.LightingLink.LightingLinkDuration	integer	Light Duration unit: s
<b>handlerName</b> .EventHandler.LightingLink.WhiteLightTimeSection	array<array<string>>	
<b>handlerName</b> .EventHandler.LightingLink.LightingLinkBright	integer	brightness
<b>handlerName</b> .ExAlarmOutChannels[ch]	integer	extend alarm out channels

## 4.9.2 Alarm Event

- Get alarm config

Table 4-87

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Alarm
Method	GET
Description	Get alarm config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Alarm
Success Return	table.Alarm[0].Enable=false table.Alarm[0].EventHandler....(output of EventHandler is described in GetEventHandler) table.Alarm[0].Name=Door1 table.Alarm[0].SensorType=NC table.Alarm[1].... ...
Comment	—

- Set alarm config

Table 4-88

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set alarm config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Alarm[0].Enable=true
Success Return	OK
Comment	In below ParamName, input is external alarm input channel which starts from 0. EventHandler defines parameter of relevant actions when alarm or event happens. It's also used in following sections about events.

#### Appendix:

ParamName	ParamValue type	Description
Alarm[ <i>input</i> ].Enable	bool	Enable/Disable alarm from a input channel
Alarm[ <i>input</i> ].EventHandler	—	Setting of EventHandler is described in <b><u>SetEventHandler</u></b> .
Alarm[ <i>input</i> ].Name	string	Name of alarm input channel.
Alarm[ <i>input</i> ].SensorType	string	Range is {NC, NO}. NC: normal close NO: normal open

### 4.9.3 Alarm Out

- Get alarm out config

Table 4-89

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AlarmOut
Method	GET

Description	Get alarm out config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=AlarmOut
Success	table.AlarmOut[ <b>alarmOutChannel</b> ].Mode=0
Return	table.AlarmOut[ <b>alarmOutChannel</b> ].Name=Beep
Comment	Parameters in Response: <b>alarmOutChannel</b> : the alarm out channel index which starts from 0.

- Set alarm out config

Table 4-90

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramV</b> <b>alue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set alarm out config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&AlarmOut[0].Mode=0 &AlarmOut[0].Name=port1
Success Return	OK
Comment	<b>port</b> in below ParamName is alarm out port index, start form 0.

#### Appendix:

ParamName	ParamValue type	Description
AlarmOut[ <b>port</b> ].Mode	integer	Range is {0, 1, 2} 0: automatically alarm 1: force alarm 2: close alarm
AlarmOut[ <b>port</b> ].Name	string	Alarm out port name.

### 4.9.4 Get Alarm Input Channels

Table 4-91

Syntax	http://<server>/cgi-bin/alarm.cgi?action=getInSlots
Method	GET
Description	Get alarm input channel number.
Example	http://192.168.1.108/cgi-bin/alarm.cgi?action=getInSlots
Success Return	result=2
Comment	—

### 4.9.5 Get Alarm Output Channels

Table 4-92

Syntax	http://<server>/cgi-bin/alarm.cgi?action=getOutSlots
Method	GET
Description	Get alarm output channel number.

Example	http://192.168.1.108/cgi-bin/alarm.cgi?action=getOutSlots
Success Return	result=1
Comment	—

## 4.9.6 Get States of Alarm Input Channels

Table 4-93

Syntax	http://<server>/cgi-bin/alarm.cgi?action=getInState
Method	GET
Description	Get alarm input state for all channels.
Example	http://192.168.1.108/cgi-bin/alarm.cgi?action=getInState
Success Return	result=3
Comment	A bit in the response result indicates a channel alarm states, result 3 means alarm channel 1 and channel 2 have alarm now.

## 4.9.7 Get States of Alarm Output Channels

Table 4-94

Syntax	http://<server>/cgi-bin/alarm.cgi?action=getOutState
Method	GET
Description	Get alarm output state for all channels.
Example	http://192.168.1.108/cgi-bin/alarm.cgi?action=getOutState
Success Return	result=0
Comment	A bit in the response result indicates a channel, result 1 means alarm is present.

## 4.9.8 Video Blind Event

- Get video blind detect config

Table 4-95

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=BlindDetect
Method	GET
Description	Get blind detect config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=BlindDetect
Success Return	<b>head</b> .Enable=false <b>head</b> .EventHandler= (output of EventHandler is described in GetEventHandler) <b>head</b> .Level=3
Comment	Parameters in Response: <b>head</b> =table.BlindDetect[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index - 1, and so 0 means channel 1).

- Set video blind detect config

Table 4-96

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set blind detect config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&BlindDetect[0].Enable=true
Success Return	OK
Comment	Parameters in URL: In table below, <b>head</b> =BlindDetect[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	Enable/Disable blind detect feature.
<b>head</b> .EventHandler	—	Setting of EventHandler is described in <b>SetEventHandler</b> .
<b>head</b> .Level	integer	Range is [1—6]. Sensitivity of blind detection. 1: lowest sensitivity. 6: highest sensitivity.

### 4.9.9 Video Loss Event

- Get video loss detect config

Table 4-97

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=LossDetect
Method	GET
Description	Get video loss detect config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=LossDetect
Success Return	<b>head</b> . Enable=false <b>head</b> . EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	Parameters in Response: <b>head</b> =table.LossDetect [ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set video loss detect config

Table 4-98

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set video loss detection config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&LossDetect[0].Enable=true
Success Return	OK
Comment	Parameters in URL: In table below, <b>head</b> =LossDetect [ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).

#### Appendix:

ParamName	ParamValue type	Description
head.Enable	bool	Enable/Disable loss detect feature.
head.EventHandler	—	Setting of EventHandler is described in <u><a href="#">SetEventHandler</a></u> .

### 4.9.10 Login Failure Event

- Get login failure event config

Table 4-99

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name>LoginFailureAlarm
Method	GET
Description	Get login failure alarm config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name>LoginFailureAlarm
Success Return	<b>head</b> .Enable=false <b>head</b> .EventHandler= (output of EventHandler is described in <a href="#">GetEventHandler</a> )
Comment	Parameters in Response: <b>head</b> =table.LoginFailureAlarm

- Set login failure alarm config

Table 4-100

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set login failure alarm config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&LoginFailureAlarm.Enable=true
Success Return	OK

Comment	Parameters in URL: In table below, <b>head</b> =LoginFailureAlarm
---------	---

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	Enable/Disable to notify LoginFailure event. Now this event can be linked with send email and alarm out. The max try login times can be configured in chapter <b>SetGeneralConfig</b> .
<b>head</b> .EventHandler	—	Setting of EventHandler is described in <b>SetEventHandler</b> .

### 4.9.11 Storage Not Exist Event

- Get storage not exist event config

Table 4-101

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageNotExist
Method	GET
Description	Get storage not exist event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=StorageNotExist
Success Return	table.StorageNotExist.Enable=false table.StorageNotExist.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set storage not exist event config

Table 4-102

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set storage not exist event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&StorageNotExist.Enable=true
Success Return	OK
Comment	—

**Appendix:**

ParamName	ParamValue type	Description
StorageNotExist.Enable	bool	Enable/Disable loss detect feature.
StorageNotExist.EventHandler	—	Setting of EventHandler is described in <b>SetEventHandler</b> .

## 4.9.12 Storage Access Failure Event

- Get storage access failure event config

Table 4-103

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageFailure
Method	GET
Description	Get storage failure event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=StorageFailure
Success Return	table.StorageFailure.Enable=false table.StorageFailure.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set storage access failure event config

Table 4-104

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set storage failure event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&StorageFailure.Enable=true
Success Return	OK
Comment	—

### Appendix:

ParamName	ParamValue type	Description
StorageFailure.Enable	bool	Enable/Disable loss detect feature.
StorageFailure.EventHandler	—	Setting of EventHandler is described in <u><a href="#">SetEventHandler</a></u> .

## 4.9.13 Storage Low Space Event

- Get storage low space event config

Table 4-105

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageLowSpace
Method	GET
Description	Get storage low space event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=StorageLowSpace
Success	table.StorageLowSpace.Enable=false

Return	table.StorageLowSpace.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set storage low space event config

Table 4-106

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set storage low space event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&StorageLowSpace.Enable=true
Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
StorageLowSpace.Enable	bool	Enable/Disable loss detect feature.
StorageLowSpace.EventHandler	—	Setting of EventHandler is described in <u>SetEventHandler</u> .

### 4.9.14 Net Abort Event

- Get net abort event config

Table 4-107

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=NetAbort
Method	GET
Description	Get net abort event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=NetAbort
Success Return	table.NetAbort.Enable=false table.NetAbort.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set net abort event config

Table 4-108

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set net abort event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&NetAbort.Enable=true
Success	OK

Return	
Comment	—

**Appendix:**

ParamName	ParamValue type	Description
NetAbort.Enable	bool	Enable/Disable loss detect feature.
NetAbort.EventHandler	—	Setting of EventHandler is described in <u><a href="#">SetEventHandler</a></u> .

## 4.9.15 IP Conflict Event

- Get IP conflict event config

Table 4-109

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=IPConflict
Method	GET
Description	Get IP conflict event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=IPConflict
Success	table.IPConflict.Enable=false
Return	table.IPConflict.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set IP conflict event config

Table 4-110

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set IP Conflict event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&IPConflict.Enable=true
Success	OK
Return	
Comment	—

**Appendix:**

ParamName	ParamValue type	Description
IPConflict.Enable	bool	Enable/Disable loss detect feature.
IPConflict.EventHandler	—	Setting of EventHandler is described in <u><a href="#">SetEventHandler</a></u> .

## 4.9.16 Get Channels Event Happened

Table 4-111

Syntax	http://<server>/cgi-bin/eventManager.cgi?action=getEventIndexes&code=<eventCode>
Method	GET
Description	Get channels indexes that event of code eventCode happens. Not all events support this command. Do not recommend to use it, use Attach command instead.
Example	http://192.168.1.108/cgi-bin/eventManager.cgi?action=getEventIndexes&code=AlarmLocal
Success Return	channels[0]=0 channels[1]=2 channels[2]=3 ... (This response means event happened on channel 0, channel 2 and channel 3 while video channel index starts from 0)
Comment	Parameters in URL: eventCode includes: VideoMotion: motion detection event VideoLoss: video loss detection event VideoBlind: video blind detection event. AlarmLocal: alarm detection event. StorageNotExist: storage not exist event. StorageFailure: storage failure event. StorageLowSpace: storage low space event. AlarmOutput: alarm output event.

## 4.9.17 Subscribe to Event Message

Table 4-112

Syntax	http://<server>/cgi-bin/eventManager.cgi?action=attach&codes=[<eventCode>,<eventCode>,...][&keepalive = 20][&heartbeat=<Heartbeat>]
Method	GET
Description	Subscribe to messages that event of code eventCode happens.
Example	http://192.168.1.108/cgi-bin/eventManager.cgi?action=attach&codes=[AlarmLocal%2CVideoMotion%2CVideoLoss%2CVideoBlind]  http://192.168.1.108/cgi-bin/eventManager.cgi?action=attach&codes=[All]&heartbeat=5
Success Return	HTTP/1.1 200 OK Cache-Control: no-cache Pragma: no-cache Expires: Thu, 01 Dec 2099 16:00:00 GMT Connection: close Content-Type: multipart/x-mixed-replace; boundary=<boundary>  --<boundary> Content-Type: text/plain Content-Length: <data length>

	<pre> &lt;eventInfo&gt; --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;data length&gt;  &lt;eventInfo&gt;  For example:  HTTP Code: 200 OK Cache-Control: no-cache Pragma: no-cache Expires: Thu, 01 Dec 2099 16:00:00 GMT Connection: close Content-Type: multipart/x-mixed-replace; boundary=myboundary  --myboundary Content-Type: text/plain Content-Length: 39  Code=VideoMotion;action=Start;index=0  --myboundary Content-Type: text/plain Content-Length: 38  Code=VideoBlind;action=Start;index=0  --myboundary Content-Type: text/plain Content-Length: 9  Heartbeat  --myboundary ... </pre>
<p>Comment</p>	<p><b>eventCode</b> can be any one of the standard codes defined in DHIF, or "<b>All</b>".  <b>All</b> means all kinds of the eventcode.  <b>eventcode</b> include:  VideoMotion: motion detection event  SmartMotionHuman: human smart motion detection  SmartMotionVehicle: Vehicle smart motion detection  VideoLoss: video loss detection event  VideoBlind: video blind detection event.  AlarmLocal: alarm detection event.  CrossLineDetection: tripwire event</p>

CrossRegionDetection:	intrusion event
LeftDetection:	abandoned object detection
TakenAwayDetection:	missing object detection
VideoAbnormalDetection:	scene change event
FaceDetection:	face detect event
AudioMutation:	intensity change
AudioAnomaly:	input abnormal
VideoUnFocus:	defocus detect event
WanderDetection:	loitering detection event
RioterDetection:	People Gathering event
ParkingDetection:	parking detection event
MoveDetection:	fast moving event
StorageNotExist:	storage not exist event.
StorageFailure:	storage failure event.
StorageLowSpace:	storage low space event.
AlarmOutput:	alarm output event.
MDResult:	motion detection data reporting event. The motion detect window contains 18 rows and 22 columns. The event info contains motion detect data with mask of every row.
HeatImagingTemper:	temperature alarm event
CrowdDetection:	crowd density overrun event
FireWarning:	fire warning event
FireWarningInfo:	fire warning specific data info
ObjectPlacementDetection:	placement object detection
ObjectRemovalDetection:	removal object detection
AccessControl	Access control event
CitizenPictureCompare	citizen picture compare event
.....	

In the example, you can see most event info is like "Code=**eventcode**; action=Start; index=0", but for some specific events, they will contain an another parameter named "data", the event info is like "Code=**eventcode**; action=Start; index=0; data=**datainfo**", the **datainfo**'s format is JSON(JavaScript Object Notation). The detail information about the specific events and **datainfo** are listed in the appendix below this table.

**keepalive:** If this param exist, the client must send any data to device by this connection in cycle. The keepalive is in range of [1,60] second.

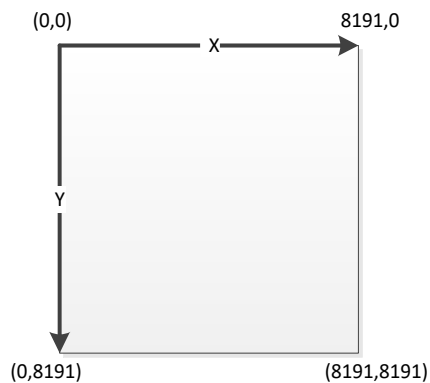
For example:

The keeplive data can be the string "keep alive".

Notice: the keepalive parameter is not recommended to use. It is recommended to use the Heartbeat parameter.

**Heartbeat:** integer, range is [1,60],unit is second.If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client,the heartbeat message are "Heartbeat".

**Note:** Heartbeat message must be sent before heartbeat timeout.



## 4.9.18 Get Capability of Event Management

Table 4-113

Syntax	http://<server>/cgi-bin/eventManager.cgi?action=getCaps
Method	GET
Description	Get event manager capabilities.
Example	http://192.168.1.108/cgi-bin/eventManager.cgi?action=getCaps
Success Return	<pre> caps.AlarmOutEnable=true caps.BeepEnable=true caps.DejitterEnable=true caps.MMSEnable=true caps.MailEnable=true caps.MonitorTourEnable=true caps.PtzLinkEnable=true caps.RecordEnable=true caps.SnapshotEnable=true caps.TimeSectionEnable=true caps.TipEnable=true caps.RecordCloudEnable=true, caps.SnapshotCloudEnable=false, caps.SnapshotTimes[0]=3, caps.SnapshotTimes[1]=10, caps.SupportAlarmBell=false, caps.SupportAccessControl=false, caps.SipCallEnable=false, caps.SupportAlarmServer=false, caps.SupportPtzLinkDelay=false, caps.SupportPSTNAlarmServer=false, caps.SupportICR=false, caps.BeepTime=false, caps.DejitterRange.Min=0, caps.DejitterRange.Max=60 caps.AlarmOutLatch[0]=1, caps.AlarmOutLatch[1]=300, caps.RecordLatch[0]=0, </pre>

	<pre> caps.RecordLatch[1]=300, caps.VoiceEnable=true, caps.VoiceLinkTimeRange[0]=10, caps.VoiceLinkTimeRange[1]=30, caps.VoicePlayTimesRange[0]=1, caps.VoicePlayTimesRange[1]=10, caps.VoiceLinkFileOptional=true, caps.LogEnable=true, caps.SupportLightControl=false, caps.LinkLightBrightRange[0]=0, caps.LinkLightBrightRange[1]=100, caps.SupportVideoMatrix=false, caps.LinkDetailCameraEnable=false, caps.SupportDisableLinkage[0]=0 caps.SupportDisableLinkage[1]=0 caps.SupportDisableLinkage[2]=0 caps.SupportDisableLinkage[3]=0 caps.SupportHTTPUpload.PictureHttpUploadEnable=false caps.SupportHTTPUpload.EventHttpUploadEnable=false </pre>
Comment	—

## 4.9.19 Net Alarm Event

- Get net alarm event config

Table 4-114

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=NetAlarm
Method	GET
Description	Get net alarm event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=NetAlarm
Success Return	<pre> table.NetAlarm[0].Enable=false table.NetAlarm[0].Name=channel1 NetAlarm[0].SensorType=NO table.NetAlarm[0].EventHandler= (output of EventHandler is described in GetEventHandler) </pre>
Comment	—

- Set net alarm event config

Table 4-115

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue> ...]
Method	GET
Description	Set net alarm event config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&NetAlarm[0].Enable=true
Success	OK

Return	
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
NetAlarm[ <i>channel</i> ].Enable	bool	Enable/Disable alarm input.
NetAlarm[ <i>channel</i> ].DefenceAreaType	string	Defence Area Type, it can be { "Intime", "Delay", "Fullday" }
NetAlarm[ <i>channel</i> ].DisableDelay	integer	delay time of removal, unit: s
NetAlarm[ <i>channel</i> ].EnableDelay	integer	delay time of protection, unit: s
NetAlarm[ <i>channel</i> ].Name	string	alarm channel name
NetAlarm[ <i>channel</i> ].SensorType	string	Sensor type, Normally open or normally closed it can be {"NO", "NC"}
NetAlarm[ <i>channel</i> ].EventHandler	—	Setting of EventHandler is described in <a href="#">SetEventHandler</a> .

### 4.9.20 Set Net Alarm State

Table 4-116

Syntax	http://<server>/cgi-bin/netAlarm.cgi?action=setState&channel=< <b>ChannelNo</b> >&alarm=< <b>Alarm</b> >[&name=< <b>Name</b> >&trigger=< <b>Trigger</b> >&desc=< <b>Desc</b> >]
Method	GET
Description	Set the netalarm state.
Example	http://192.168.1.108/cgi-bin/netAlarm.cgi?action=setState&channel=1&alarm=true&name=somke&trigger=SmokingSensor&desc=Zone8
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, alarm channel index which starts from 1</p> <p><b>Alarm</b>: bool, the state of the alarm, the value can be true and false, true means the alarm is start.</p> <p><b>Name</b>: string, the name of the alarm, optional parameter, the length of this value should less than 32.</p> <p><b>Trigger</b>:string, the trigger of the alarm,for example, it can be Infrared, WaterSensor or Manual and so on.</p> <p>Optional parameter, the length of this value should less than 32.</p> <p><b>Desc</b>: string, the description of the alarm, optional parameter, the length of this value should less than 128.</p>

### 4.9.21 GetSupportedEvents

Table 4-117

Syntax	http://<server>/cgi-bin/eventManager.cgi?action=getExposureEvents
Method	GET
Description	Get the event list which the device supports.
Example	http://192.168.1.108/cgi-bin/eventManager.cgi?action=getExposureEvents
Success Return	events[0]=VideoMotion events[1]=AlarmLocal events[2]=FaceDetection events[3]=VideoMotion ...
Comment	—

## 4.10 PTZ

### 4.10.1 PTZ Config

- Get PTZ config

Table 4-118

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Ptz
Method	GET
Description	Get PTZ config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Ptz
Success Return	table.Ptz[ <b>port</b> ].Address=8 table.Ptz[ <b>port</b> ].Attribute[0]=115200 table.Ptz[ <b>port</b> ].Attribute[1]=8 table.Ptz[ <b>port</b> ].Attribute[2]=Even table.Ptz[ <b>port</b> ].Attribute[3]=1 table.Ptz[ <b>port</b> ].Homing[0]=0 table.Ptz[ <b>port</b> ].Homing[1]=30 table.Ptz[ <b>port</b> ].NumberInMatrixs=0 table.Ptz[ <b>port</b> ].ProtocolName=NONE
Comment	Parameters in Response: <b>port</b> is PTZ port index, start form 0.

- Set PTZ config

Table 4-119

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set PTZ config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Ptz[0].Address=192.168.0.1&Ptz[0].Attribute[0]=9600
Success Return	OK
Comment	<b>port</b> in below ParamName is PTZ port index, start form 0.

## Appendix:

ParamName	ParamValue type	Description
Ptz[ <i>port</i> ].Address	integer	Range is [0—255]. Device address, if there are more than one device connected to this port, distinguish them by this address.
Ptz[ <i>port</i> ].Attribute[0]	integer	The baud rate. Range is {1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200}.
Ptz[ <i>port</i> ].Attribute[1]	integer	Range is {4, 5, 6, 7, 8}. Data bit.
Ptz[ <i>port</i> ].Attribute[2]	string	Range is {Even, Mark, None, Odd, Space}. Parity verification mode.
Ptz[ <i>port</i> ].Attribute[3]	float	Range is {1, 1.5, 2}. Stop bit.
Ptz[ <i>port</i> ].Homing[0]	integer	Range is {-1,0—255} -1: homing is disabled. [0—255]: preset point number
Ptz[ <i>port</i> ].Homing[1]	integer	Range is [0—65535]. No operation timeout, unit is seconds. After no operation timeout, PTZ go to preset point set in Ptz[ <i>port</i> ].Homing[0].
Ptz[ <i>port</i> ].ProtocolName	string	PTZ protocol name depends on PTZ capability. Refer to <a href="#">GetProtocolList</a> to get the protocol list.
Ptz[ <i>port</i> ].ControlDelayTime	integer	PTZ control delay time, unit: second
Ptz[ <i>port</i> ].ControlPriority	string	PTZ control priority,the range is "RS485", "Net". default : "Net"

### 4.10.2 Get PTZ Protocol List

Table 4-120

Syntax	http://<server>/cgi-bin/ptz.cgi?action=getProtocolList[&channel=<ChannelNo>]
Method	GET
Description	Get the protocol list that PTZ can support. Unsupported now.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=getProtocolList&channel=1
Success Return	info.RS[0]=Pelco info.RS[1]=DH-SD1 info.Coaxial[0]=HD-CVI info.Coaxial[1]=HD-CVI2.0
Comment	Response contains all support PTZ protocols of the server. <b>ChannelNo</b> : integer, video channel index which starts from 1.

### 4.10.3 Get PTZ Capability of Current Protocol

Table 4-121

Syntax	http://<server>/cgi-bin/ptz.cgi?action=getCurrentProtocolCaps[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Get PTZ channel protocol capabilities.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=getCurrentProtocolCaps&channel=1
Success Return	caps. <b>AlarmLen</b> =0 caps. <b>AuxMax</b> =8 caps. <b>AuxMin</b> =1 caps. <b>CamAddrMax</b> =255 caps. <b>CamAddrMin</b> =1 caps. <b>Flip</b> =false caps. <b>Focus</b> =false caps. <b>Interval</b> =200 caps. <b>Iris</b> =false caps. <b>Menu</b> =false caps. <b>MonAddrMax</b> =255 caps. <b>MonAddrMin</b> =0 caps. <b>Name</b> =DH-SD1 caps. <b>Pan</b> =false caps. <b>PanSpeedMax</b> =255 caps. <b>PanSpeedMin</b> =1 caps. <b>PatternMax</b> =5 caps. <b>PatternMin</b> =1 caps. <b>PresetMax</b> =80 caps. <b>PresetMin</b> =1 caps. <b>Tile</b> =false caps. <b>TileSpeedMax</b> =255 caps. <b>TileSpeedMin</b> =1 caps. <b>TourMax</b> =7 caps. <b>TourMin</b> =0 caps. <b>Type</b> =1 caps. <b>Zoom</b> =false caps.PtzMotionRange.HorizontalAngle[0]=0 caps.PtzMotionRange.HorizontalAngle[1]=360 caps.PtzMotionRange.VerticalAngle[0]=-20 caps.PtzMotionRange.VerticalAngle[1]=90 caps. <b>ZoomMax</b> =30 caps. <b>ZoomMin</b> =1
Comment	Parameters in URL: <b>ChannelNo</b> : PTZ channel index which starts from 1.

#### Appendix:

Field in response	Description
AlarmLen	Alarm length in protocol.
AuxMax	Maximum/Minimum number for auxiliary functions.
AuxMin	
CamAddrMax	Maximum/Minimum channel address.
CamAddrMin	

Field in response	Description
Flip	True or false, support picture flip or not.
Focus	True or false, support focus or not.
Iris	True or false, support Iris adjusts or not.
Menu	True or false, support internal menu of the PTZ or not.
MonAddrMax	Maximum/Minimum monitor address.
MonAddrMin	
Name	Name of the operation protocol.
Pan	True or false, support pan or not.
PanSpeedMax	Maximum/Minimum pan speed.
PanSpeedMin	
PatternMax	Maximum/Minimum pattern path number.
PatternMin	
PresetMax	Maximum/Minimum preset point number.
PresetMin	
Tile	True or false, support tilt or not.
Zoom	True or false, support zoom or not.
TileSpeedMax	Maximum/Minimum tile speed.
TileSpeedMin	
TourMax	Maximum/Minimum tour path number.
TourMin	
Type	Type of PTZ protocol.
PtzMotionRange.HorizontalAngle	Horizontal angle range, [0] for minimum angle, [1] for maximum angle it only when Pan was true
PtzMotionRange.VerticalAngle	Vertical angle range, [0] for minimum angle, [1] for maximum angle it only when Tile was true
ZoomMax	Maximum/Minimum Zoom.
ZoomMin	it only when Zoom was true

#### 4.10.4 Get PTZ Status

Table 4-122

Syntax	http://<server>/cgi-bin/ptz.cgi?action=getStatus[&channel=<ChannelNo>]
Method	GET
Description	Get PTZ status.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=getStatus&channel=1
Success Return	status.UTC=6538920 status.MoveStatus=Idle status.ZoomStatus=Idle status.PresetID=10 status.Position=120,12,2
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1.

## 4.10.5 PTZ Control

- Basic movement

Table 4-123

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=&lt;action&gt;&amp;channel=&lt;ch&gt;&amp;code=&lt;code&gt;&amp;arg1=&lt;arg1&gt;&amp;arg2=&lt;arg2&gt;&amp;arg3=&lt;arg3&gt;</code>
Method	GET
Description	PTZ basic movement.
Example	<code>http://192.168.1.108/cgi-bin/ptz.cgi?action=start&amp;channel=1&amp;code=Up&amp;arg1=0&amp;arg2=1&amp;arg3=0</code> <code>http://192.168.1.108/cgi-bin/ptz.cgi?action=stop&amp;channel=1&amp;code=Up&amp;arg1=0&amp;arg2=1&amp;arg3=0</code>
Success Return	OK
Comment	Parameters in URL: <b>action</b> is PTZ control command, it can be start or stop. <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>code</b> is PTZ operation, and <b>arg1</b> , <b>arg2</b> , <b>arg3</b> are arguments of operation. <b>code</b> and <b>argN</b> values are listed in table below.
Comment	The default exposure mode of PTZ is auto-exposure, aperture adjustment (Aperture larger 、 Aperture smaller) would take effects temporarily, PTZ will auto—adjust the aperture after the auto—exposure recovery time (15 mins) The default focus mode of PTZ is semi—auto, PTZ control or other operations will trigger the auto focus

### Appendix:

Code	Code description	arg1	arg2	arg3
Up	Tile up	0	Vertical speed, range is [1—8]	0
Down	Tile down	0	Vertical speed, range is [1—8]	0
Left	Pan left	0	Horizontal speed, range is [1—8]	0
Right	Pan right	0	Horizontal speed, range is [1—8]	0
LeftUp	Pan left and tile up	Vertical speed, range is [1—8]	Horizontal speed, range is [1—8]	0
RightUp	Pan right and tile up	Vertical speed, range is [1—8]	Horizontal speed, range is [1—8]	0
LeftDown	Pan left and tile down	Vertical speed, range is [1—8]	Horizontal speed, range is [1—8]	0
RightDown	Pan right and tile down	Vertical speed, range is [1—8]	Horizontal speed, range is [1—8]	0
ZoomWide	Zoom out	0	0	0

Code	Code description	arg1	arg2	arg3
ZoomTele	Zoom in	0	0	0
FocusNear	Focus near	0	0	0
FocusFar	Focus far	0	0	0
IrisLarge	Aperture larger	0	0	0
IrisSmall	Aperture smaller	0	0	0

- Start the continuous movement

Table 4-124

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=start&amp;channel=&lt;ch&gt;&amp;code=Continuously&amp;arg1=&lt;arg1&gt;&amp;arg2=&lt;arg2&gt;&amp;arg3=&lt;arg3&gt;&amp;arg4=&lt;arg4&gt;</code>
Method	GET
Description	Start the continuous movement.
Example	<code>http://192.168.1.108/cgi-bin/ptz.cgi?action=start&amp;channel=1&amp;code=Continuously&amp;arg1=5&amp;arg2=5&amp;arg3=5&amp;arg4=60</code>
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg3</b> is zoom speed, range is [-100—100]; <b>arg4</b> is timeout value, the measurement unit is second, it's an integer greater than 0, the PTZ will stop moving automatically if timeout value reached and no"stop"command received. The maximum timeout value is 3600 seconds.

#### Appendix:

Move description	arg1	arg2
Pan left Continuously	Less than -4	0
Pan right Continuously	Geater than 4	0
Tile up Continuously	0	Geater than 4
Tile down Continuously	0	Less than -4
Pan left and tile up Continuously	Less than -4	Geater than 4
Pan right and tile up Continuously	Geater than 4	Geater than 4
Pan left and tile down Continuously	Less than -4	Less than -4
Pan right and tile down Continuously	Geater than 4	Less than -4

- Stop the continuous movement

Table 4-125

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=stop&amp;channel=&lt;ch&gt;&amp;code=Continuously&amp;arg1=0&amp;arg2=0&amp;arg3=0&amp;arg4=0</code>
Method	GET

Description	Immediately stop the continuous movement.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=stop&channel=1&code=Continuously&arg1=0&arg2=0&arg3=0&arg4=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

- 3D positioning

Table 4-126

Syntax	http://<server>/cgi-bin/ptzBase.cgi?action=moveDirectly&channel=<ChannelNo>&startPoint[0]=<startX>&startPoint[1]=<startY>&endPoint[0]=<endX>&endPoint[1]=<endY>
Method	GET
Description	Three-dimensional orientation. Move to the rectangle with screen coordinate [startX, startY], [endX, endY] .
Example	http://192.168.1.108/cgi-bin/ptzBase.cgi?action=moveDirectly&channel=1&startPoint[0]=7253&startPoint[1]=2275&endPoint[0]=7893&endPoint[1]=3034
Success Return	OK
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1. <b>startX, startY, endX, endY</b> : relative coordinates, range is 0-8192. The two points [startX, startY] and [endX, endY] makes the destination rectangle.

- PTZ relative movement

Table 4-127

Syntax	http://<server>/cgi-bin/ptz.cgi?action=moveRelatively&channel=<ch>&arg1=<arg1>&arg2=<arg2>&arg3=<arg3>
Method	GET
Description	PTZ relative movement.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=moveRelatively&channel=1&arg1=1&arg2=1&arg3=1
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is relative horizontal coordinate <b>arg2</b> is relative vertical coordinate <b>arg3</b> is relative zoom coordinate The normalization of above three values is[-1,1]

- PTZ accurate positioning

Table 4-128

Syntax	http://<server>/cgi-bin/ptz.cgi?action=moveAbsolutely&channel=<ch>&arg1=<arg1>&arg2=<arg2>&arg3=<arg3>
Method	GET

Description	Accurate positioning.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=moveAbsolutely&channel=1&arg1=-0.8&arg2=0.3&arg3=0.5
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].</p> <p><b>arg1</b> is absolute horizontal coordinate, the normalization value is [-1, 1].  arg1&lt;0: angle = 180.0 * arg1 + 360.0, real range is [180.0, 360.0]  arg1&gt;=0: angle = 180.0 * arg1, real range is [0, 180.0]</p> <p><b>arg2</b> is absolute vertical coordinate, the normalization value is [-1, 1].  arg2: angle = -180.0 * arg2, real range is [-180.0, 180.0]</p> <p><b>arg3</b> is absolute zoom coordinate, the normalization value is [0, 1].</p> <p>Actual range subject to equipment, you can get the detail information by protocol 4.10.3(Get PTZ Capability of Current Protocol).</p>

## 4.10.6 Preset

- Get preset information

Table 4-129

Syntax	http://<server>/cgi-bin/ptz.cgi?action=getPresets[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Get preset of PTZ control.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=getPresets&channel=1
Success Return	<pre>presets[0].Index=1 presets[0].Name=Preset 1 presets[0].Type=0 presets[0].PresetFunction="VideoBlack" presets[0].Position=[900, -900, 5] ...</pre>
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.  The size of presets-array is the number of presets.</p>

- Go to preset

Table 4-130

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=GotoPreset&arg1=0&arg2=< <b>index</b> >&arg3=0
Method	GET
Description	Go to preset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=GotoPreset&arg1=0&arg2=1&arg3=0
Success Return	OK

Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg2</b> is number of preset
---------	--

- Set preset

Table 4-131

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=SetPreset&arg1=0&arg2=< <b>index</b> >&arg3=0
Method	GET
Description	Set preset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=SetPreset&arg1=0&arg2=2&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg2</b> is preset number

- Set the name for specified preset

Table 4-132

Syntax	http://<server>/cgi-bin/ptz.cgi?action=setPreset&channel=< <b>ch</b> >&arg1=< <b>index</b> >&arg2=< <b>name</b> >
Method	GET
Description	Set the name for specified preset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=setPreset&channel=1&arg1=2&arg2=2
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is preset number <b>arg2</b> is preset name, could be letter,number or Chinese

- Delete specified preset

Table 4-133

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=ClearPreset&arg1=0&arg2=< <b>index</b> >&arg3=0
Method	GET
Description	Delete the specified preset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=ClearPreset&arg1=0&arg2=1&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg2</b> is preset number

## 4.10.7 Tour

- Start the tour

Table 4-134

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=StartTour&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Start the tour.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=StartTour&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour route

- Stop the tour

Table 4-135

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=StopTour&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Stop the tour.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=StopTour&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour route

- Add tour group

Table 4-136

Syntax	http://<server>/cgi-bin/ptz.cgi?action=setTour&channel=<ch>&arg1=<index>&arg2=<name>
Method	GET
Description	Add tour group, including set the tour route and name. Frequently used with add tour preset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=setTour&channel=1&arg1=1&arg2=1
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour route

	<b>arg2</b> is tour name
--	--------------------------

- Delete tour group

Table 4-137

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=ClearTour&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Delete the tour group with specified number.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=ClearTour&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour group

- Add tour preset

Table 4-138

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=AddTour&arg1=<index1>&arg2=<index2>&arg3=0
Method	GET
Description	Add preset to tour group.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=AddTour&arg1=1&arg2=2&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour route <b>arg2</b> is the number of preset

- Delete tour preset

Table 4-139

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=DelTour&arg1=<index1>&arg2=<index2>&arg3=0
Method	GET
Description	Delete the specified preset from a particular tour group.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=DelTour&arg1=1&arg2=2&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the number of tour route <b>arg2</b> is the number of preset

## 4.10.8 Scan

- Set left boundary

Table 4-140

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=SetLeftLimit&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Set left boundary.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=SetLeftLimit&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the scan number

- Set right boundary

Table 4-141

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=SetRightLimit&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Set right boundary.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=SetRightLimit&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is the scan number

- Start scan

Table 4-142

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=AutoScanOn&arg1=<index>&arg2=0&arg3=0
Method	GET
Description	Start scanning.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=AutoScanOn&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1—n]. <b>arg1</b> is scan number

- Stop scan

Table 4-143

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=AutoScanOff&arg1=< <b>index</b> >&arg2=0&arg3=0
Method	GET
Description	Stop scanning.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=AutoScanOff&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is scan number

#### 4.10.9 Pattern

- Start pattern record

Table 4-144

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=SetPatternBegin&arg1=< <b>index</b> >&arg2=0&arg3=0
Method	GET
Description	Start pattern record.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=SetPatternBegin&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is pattern number

- Stop pattern record

Table 4-145

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=SetPatternEnd&arg1=< <b>index</b> >&arg2=0&arg3=0
Method	GET
Description	Stop pattern record.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=SetPatternEnd&arg1=1&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is pattern number

- Start pattern

Table 4-146

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=start&amp;channel=&lt;ch&gt;&amp;code=StartPattern&amp;arg1=&lt;index&gt;&amp;arg2=0&amp;arg3=0</code>
Method	GET
Description	Start pattern.
Example	<code>http://192.168.1.108/cgi-bin/ptz.cgi?action=start&amp;channel=1&amp;code=StartPattern&amp;arg1=1&amp;arg2=0&amp;arg3=0</code>
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is pattern number

- Stop pattern

Table 4-147

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=start&amp;channel=&lt;ch&gt;&amp;code=StopPattern&amp;arg1=&lt;index&gt;&amp;arg2=0&amp;arg3=0</code>
Method	GET
Description	Stop pattern.
Example	<code>http://192.168.1.108/cgi-bin/ptz.cgi?action=start&amp;channel=1&amp;code=StopPattern&amp;arg1=1&amp;arg2=0&amp;arg3=0</code>
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>arg1</b> is pattern number

#### 4.10.10 Pan

- Start pan

Table 4-148

Syntax	<code>http://&lt;server&gt;/cgi-bin/ptz.cgi?action=start&amp;channel=&lt;ch&gt;&amp;code=AutoPanOn&amp;arg1=0&amp;arg2=0&amp;arg3=0</code>
Method	GET
Description	Start pan.
Example	<code>http://192.168.1.108/cgi-bin/ptz.cgi?action=start&amp;channel=1&amp;code=AutoPanOn&amp;arg1=0&amp;arg2=0&amp;arg3=0</code>
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

- Stop pan

Table 4-149

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=AutoPanOff&arg1=0&arg2=0&arg3=0
Method	GET
Description	Stop pan.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=AutoPanOff&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

#### 4.10.11 PTZ Auto Movement

- Get PTZ auto movement configuration

Table 4-150

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=PtzAutoMovement
Method	GET
Description	Get PTZ auto movement configuration.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=PtzAutoMovement
Success Return	table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].Enable = true table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].TimeSection[week][section] = 0 00:00:00-23:59:59 ... table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].Function = None table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].ScanId = 0 table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].PresetId = 0 table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].PatternId = 0 table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].TourId = 0 table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].AutoHoming.Time = 30 table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].SnapshotEnable = false table.PtzAutoMovement[ <b>port</b> ][ <b>Task</b> ].SnapshotDelayTime = 30
Comment	Parameters in Response: <b>port</b> is PTZ port index, start from 0. <b>Task</b> is the number of task, start from 0. <b>week</b> : from 1 to 7. <b>section</b> : time section, from 0 to 5.

- Set PTZ auto movement

Table 4-151

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET

Description	Set PTZ auto movement.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&PtzAutoMovement[0][0].Function=Preset&PtzAutoMovement[0][0].PresetId=1
Success Return	OK
Comment	Parameters in URL: In table below, <b>head</b> =PtzAutoMovement[port][task] <b>port</b> is PTZ port index, start from 0. <b>task</b> is the number of task, start from 0. <b>week</b> : from 1 to 7. <b>section</b> : time section, from 0 to 5.

#### Appendix:

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	Enable/Disable PtzAutoMovement
<b>head</b> .TimeSection	timeSchedule	timeSchedule[ <b>week</b> ][ <b>section</b> ]=1 10:00:00-11:00:00 ...
<b>head</b> .Function	string	Range is {Scan, Preset, Pattern, Tour}.
<b>head</b> .ScanId	integer	Scan Id, start from 1
<b>head</b> .PresetId	integer	Preset Id, start from 1
<b>head</b> .PatternId	integer	Pattern Id, start from 1
<b>head</b> .TourId	integer	Tour Id, start from 1
<b>head</b> .AutoHoming.Time	integer	Recover time, unit is second.
<b>head</b> .SnapshotEnable	bool	Enable/Disable Snap, when "Fuction" is "Preset".
<b>head</b> .SnapshotDelayTime	integer	Delay time of snap, when "Fuction" is "Preset".

#### 4.10.12 PTZ Restart

Table 4-152

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=Restart&arg1=0&arg2=0&arg3=0
Method	GET
Description	PTZ restart.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=Restart&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

#### 4.10.13 PTZ Reset

Table 4-153

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=Reset&arg1=0&arg2=0&arg3=0
Method	GET
Description	PTZ reset.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=Reset&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

#### 4.10.14 OSD Menu

- Enter the menu

Table 4-154

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=Menu&arg1=0&arg2=0&arg3=0
Method	GET
Description	Enter the menu.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=Menu&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

- Exit the menu

Table 4-155

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=Exit&arg1=0&arg2=0&arg3=0
Method	GET
Description	Exit the menu.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=Exit&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

- Confirm

Table 4-156

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=<ch>&code=Enter&arg1=0&arg2=0&arg3=0
Method	GET

Description	Confirm the menu.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=Enter&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n].

- Start the basic operation of menu

Table 4-157

Syntax	http://<server>/cgi-bin/ptz.cgi?action=start&channel=< <b>ch</b> >&code=< <b>code</b> >&arg1=< <b>arg1</b> >&arg2=< <b>arg2</b> >&arg3=< <b>arg3</b> >
Method	GET
Description	Start the basic operation of menu.
Example	http://192.168.1.108/cgi-bin/ptz.cgi?action=start&channel=1&code=MenuUp&arg1=0&arg2=0&arg3=0
Success Return	OK
Comment	Parameters in URL: <b>ch</b> is PTZ channel index which starts from 1. Range is [1 — n]. <b>code</b> is PTZ operation, and arg1, arg2, arg3 are arguments of operation. <b>code</b> and argN values are listed in table below.

#### Appendix:

Code	Code description	arg1	arg2	arg3
MenuUp	MenuUp	0	0	0
MenuDown	MenuDown	0	0	0
MenuLeft	MenuLeft	0	0	0
MenuRight	MenuRight	0	0	0

## 4.11 Record

### 4.11.1 Get Capability of Recording

Table 4-158

Syntax	http://<server>/cgi-bin/recordManager.cgi?action=getCaps
Method	GET
Description	Get record manager capabilities.
Example	http://192.168.1.108/cgi-bin/recordManager.cgi?action=getCaps
Success Return	caps.MaxPreRecordTime=30 caps.PacketLengthRange[0]=1 caps.PacketLengthRange[1]=60 caps.PacketSizeRange[0]=131072 caps.PacketSizeRange[1]=2097152

	caps.SupportExtraRecordMode=true caps.SupportHoliday=true caps.SupportPacketType[0]=Time caps.SupportPacketType[1]=Size caps.SupportResumeTransmit=false
Comment	—

## 4.11.2 Record Config

- Get record config

Table 4-159

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Record
Method	GET
Description	Get record config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Record
Success Return	table.Record[ <b>channel</b> ].PreRecord=6 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][0]=65535 00:00:00-24:00:00 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][1]=0 02:00:00-24:00:00 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][2]=0 03:00:00-24:00:00 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][3]=0 04:00:00-24:00:00 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][4]=0 05:00:00-24:00:00 table.Record[ <b>channel</b> ].TimeSection[ <b>weekday</b> ][5]=0 06:00:00-24:00:00
Comment	Parameters in Response: <b>channel</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1). <b>weekday</b> : range is [0-6] (Sunday - Saturday). Record config contains pre record time and record time sections of every day.

- Set record config

Table 4-160

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set record config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&Record[0].TimeSection[0][0]=6 00:00:00-23:59:59 Set record time to every Sunday all day. Record type is motion detection and alarm. In this example, "6 00:00:00-23:59:59" means motion detection and alarm record all day (6 = 4 & 2, alarm is 4, motion detection is 2.).
Success Return	OK
Comment	Parameters in URL: In table below, <b>ch</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

	<b>wd:</b> week day index <b>ts:</b> time section index
--	--

**Appendix:**

ParamName	ParamValue type	Description
Record[ <i>ch</i> ].PreRecord	integer	Range is [0—300]. Prerecord seconds, 0 means no prerecord. <i>ch</i> (Channel number) starts form 0
Record[ <i>ch</i> ].TimeSection[ <i>wd</i> ][ <i>ts</i> ]	string	<i>wd</i> (week day) range is [0—7] ,[0-6]:(Sunday - Saturday),7:Holiday <i>ts</i> (time section) range is [0—23], time section table index.  Format: mask hh:mm:ss-hh:mm:ss Mask: [0—4294967295], hh: [0—24], mm: [0—59], ss: [0—59] Mask indicates record type by bits: Bit0: regular record Bit1: motion detection record Bit2: alarm record Bit3: card record

### 4.11.3 Record Mode

- Get record mode config

Table 4-161

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=RecordMode
Method	GET
Description	Get record mode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RecordMode
Success Return	table.RecordMode[ <b>Channel</b> ].Mode=0
Comment	Parameters in Response: <b>Channel:</b> integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set record mode config

Table 4-162

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >
Method	GET
Description	Set record mode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&RecordMode[0].Mode=0

Success Return	OK
Comment	Parameters in URL: In table below, <b>Channel</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

**Appendix:**

ParamName	ParamValue type	Description
RecordMode[ <b>Channel</b> ].Mode	integer	Range is {0, 1, 2 }. 0: automatically record 1: manually record 2: stop record.

### 4.11.4 Media Global

- Get media global config

Table 4-163

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=MediaGlobal
Method	GET
Description	Get media global config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=MediaGlobal
Success Return	table.MediaGlobal.OverWrite=true, table.MediaGlobal.PacketType=0, table.MediaGlobal.PacketLength=60, table.MediaGlobal.PacketSize=1024, table.MediaGlobal.LogRecord=false, table.MediaGlobal.LogEncode=false, table.MediaGlobal.SnapFormatAs=MainFormat
Comment	—

- Set media global config

Table 4-164

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>
Method	GET
Description	Set media global config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&MediaGlobal.SnapFormatAs=MainFormat
Success Return	OK
Comment	—

**Appendix:**

ParamName	ParamValue type	Description
MediaGlobal.SnapFormatAs	string	The range is {"MainFormat", "ExtraFormat"}.

## 4.11.5 Find Media Files

Step 1 Create a media files finder.

Table 4-165

Syntax	http://<server>/cgi-bin/mediaFileFind.cgi?action=factory.create
Method	GET
Description	Create a media file finder.
Example	http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=factory.create
Success Return	result=08137
Comment	The <b>result</b> is the finder's <b>objectId</b> , needed by the following API.

Step 2 Start to find media files satisfied the conditions with the finder.

Table 4-166

Syntax	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile&object=<objectId>&condition.Channel=<ChannelNo>&condition.StartTime=<start>&condition.EndTime=<end>[&condition.Dirs[0]=<dir>&condition.Types[0]=<type>&condition.Flags[0]=<flags>&condition.Events[0]=<event>&condition.VideoStream=<stream>]
Method	GET
Description	Check if there are files that satisfy all the conditions.
Example	Find a file in channel 1, in directory "/mnt/dvr/sda0", event type is "AlarmLocal" or "VideoMotion", file type is "dav", and time between 2014-1-1 12:00:00 and 2015-1-10 12:00:00, URL is: http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condition.Channel=1&condition.Dirs[0]=/mnt/dvr/sda0&condition.Types[0]=dav&condition.Events[0]=AlarmLocal&condition.Events[1]=VideoMotion&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.VideoStream=Main
Success Return	OK

Comment	<p>Start to find a file with the above condition. If files exist, return OK, else return Error.</p> <p>Parameters in URL:</p> <p><b>objectId</b>: The object Id is the finder created before. You must create a finder before finding files.</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p><b>start / end</b>: the start/end time when recording.</p> <p><b>dir</b> : in which directories you want to find the file. It is an array. The index starts from 0. The range of dir is {"/mnt/dvr/sda0", "/mnt/dvr/sda1"}. This condition can be omitted. If omitted, find files in all the directories.</p> <p><b>type</b> : which types of the file you want to find. It is an array. The index starts from 0. The range of type is {"dav", "jpg", "mp4"}. If omitted, find files with all the types.</p> <p><b>flags</b> : which flags of the file you want to find. It is an array. The index starts from 0. The range of flag is {"Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout"}. If omitted, find files with all the flags.</p> <p><b>event</b> : by which event the record file is triggered. It is an array. The index starts from 0. The range of event is {"AlarmLocal", "VideoMotion", "VideoLoss", "VideoBlind", "Traffic*"}. This condition can be omitted. If omitted, find files of all the events.</p> <p><b>stream</b> : which video stream type you want to find. The range of stream is {"Main", "Extra1", "Extra2", "Extra3"}. If omitted, find files with all the stream types.</p>
---------	--

Step 3 Get the media file information found by the finder.

Table 4-167

Syntax	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=< <b>objectId</b> >&count=< <b>fileCount</b> >
Method	GET
Description	Find the next files no more than <b>fileCount</b> .
Example	http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100
Success Return	<b>found</b> =1 items[0]. <b>Channel</b> =1 items[0]. <b>StartTime</b> =2011-1-1 12:00:00 items[0]. <b>EndTime</b> =2011-1-1 13:00:00 items[0]. <b>Type</b> =dav items[0]. <b>Events[0]</b> =AlarmLocal items[0]. <b>VideoStream</b> =Main items[0]. <b>FilePath</b> =/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0]. <b>Length</b> =790 items[0]. <b>Duration</b> =3600
Comment	The maximum value of <b>fileCount</b> is 100.

**Appendix:**

Field in Response	Description
found	Count of found file, found is 0 if no file is found.
Channel	Internal video channel index starts from 0, equals to API findFile input condition.Channel -1.

Field in Response	Description
StartTime	Start time.
EndTime	End time.
Type	File type.
Events	Event type.
VideoStream	Video stream type.
FilePath	File path.
CutLength	File length that cut between StartTime and EndTime.
Length	File length of the whole file.
Duration	Duration time.

**Step 4** Close the finder.

Table 4-168

Syntax	http://<server>/cgi-bin/mediaFileFind.cgi?action=close&object=<objectId>
Method	GET
Description	Stop finding.
Example	http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=close&object=08137
Success Return	OK
Comment	—

**Step 5** Destroy the finder.

Table 4-169

Syntax	http://<server>/cgi-bin/mediaFileFind.cgi?action=destroy&object=<objectId>
Method	GET
Description	Destroy the media file finder.
Example	http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=destroy&object=08137
Success Return	OK
Comment	—

## 4.11.6 Find media files with FaceDetection info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and FaceDetection condition with the finder.

Table 4-170

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile
Method	GET
Description	Start to find media files satisfied the common conditions and FaceDetection conditions with the finder.
[ Request Params ] ( key=value format at URL )	

Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Types	array<string>	O	Search file types, should be "jpg"
+Flags	array<string>	O	Search flags, can be : "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout" should include "Event"
+Events	array<string>	R	Search event list, must be one string : " <b>FaceDetection</b> "
+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB filter object
++FaceDetectionRecordFilter	object	R	DB filter for FaceDetection
+++ImageType	string	O	Picture type, if omit, means search all type. "GlobalSence" : means big picture of the full sence, "Small" : means small picture of the people face.
+++Sex	string	O	Sex, it can be "Man", "Woman", if omit, search all
+++Age	array<int>	O	Age range, ex: [25, 40]
+++Glasses	int	O	Glasses Status, 0: all, 1: not wear, 2: wear
+++Mask	int	O	Mask Status, 0: all, 1: not wearing mask, 2: wearing mask
+++Beard	int	O	Beard Status, 0: all, 1: no beard, 2: has beard
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condition.Channel=1&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.Types[0]=jpg&condition.Flags[0]=Event&condition.Events[0]=FaceDetection&condition.DB.FaceDetectionRecordFilter.ImageType=GlobalSence&condition.DB.FaceDetectionRecordFilter.Sex=Man&condition.DB.FaceDetectionRecordFilter.Age[0]=25&condition.DB.FaceDetectionRecordFilter.Age[1]=40&condition.DB.FaceDetectionRecordFilter.Glasses=1		
Response	OK		

**Step 3** Get the media file information found by the finder

Table 4-171

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id

count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"
+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+SummaryNew	array<object>	R	DB record object
++Key	string	R	DB record name, should be "FaceDetectionRecord"
++Value	object	R	DB record value
+++ImageType	string	O	Picture type. It can be: "GlobalSence", "Small"
+++TimeStamp	object	O	Picture timestamp
++++UTC	int	O	UTC seconds
++++UTCMS	int	O	UTC milliseconds
+++Sex	string	O	Sex, it can be "Man", "Woman", "Unknown"
+++Age	int	O	Age
+++Glasses	int	O	Glasses Status, 0: unknown, 1: not wear, 2: wear
+++Mask	int	O	Mask Status, 0: unknown, 1: not wearing mask, 2: wearing mask
+++Beard	int	O	Beard Status, 0: unknown, 1: no beard, 2: has beard
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100		

Response	<pre> found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=FaceDetection items[0].FilePath=/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].CutLength=79000 items[0].SummaryNew[0].Key=FaceDetectionRecord items[0].SummaryNew[0].Value.ImageType=GlobalSence items[0].SummaryNew[0].Value.TimeStamp.UTC=134652732 items[0].SummaryNew[0].Value.TimeStamp.UTCMS=134 items[0].SummaryNew[0].Value.Sex=Man items[0].SummaryNew[0].Value.Age=30 items[0].SummaryNew[0].Value.Glasses=1 items[0].SummaryNew[0].Value.Mask=2 items[0].SummaryNew[0].Value.Beard=1 ... </pre>
----------	---

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

## 4.11.7 Find media files with FaceRecognition info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and FaceRecognition conditions with the finder.

Table 4-172

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile		
Method	GET		
Description	Start to find media files satisfied the common conditions and FaceRecognition conditions with the finder.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	O	Start time to search, ex: "2010-05-25 12:05:00" Ignored, use StartTime in DB param
+EndTime	string	O	End time to search, ex: "2010-06-25 12:05:00" Ignored, use EndTime in DB param
+Types	array<string>	O	Search file types, should be "jpg"

+Flags	array<string>	O	Search flags. It can be: "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout" should include "Event".
+Events	array<string>	R	Search event list, must be one string: "FaceRecognition"
+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB filter object
++FaceRecognitionRecordFilter	object	R	DB filter for FaceRecognition
+++MachineAddress	string	O	Machine address
+++StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+++EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+++Person	object	O	Search condition of history person and the similar face group person candidates
++++Name	string	O	Person's name, max string length is 15
++++Sex	string	O	Sex, can be "Male", "Female", if omit, search all
++++Birthday	string	O	Birthday, max string length is 11, ex: 1990-5-1
++++Country	string	O	Country, ISO 3166, string length should be 2
++++Province	string	O	Province, max string length is 63
++++City	string	O	City, max string length is 63
++++CertificateType	string	O	Certificate Type. It can be: "IC", "Passport", "Military", "Unknown"
++++ID	string	O	Person ID of CertificateType, max string length is 31
++++GroupID	string	O	The identity of the Face Group that this Person in. max string length is 63
++++Age	array<int>	O	Age range, ex: [25, 40]
++++Glasses	int	O	Glasses Status, 0: all, 1: not wear, 2: wear
++++Mask	int	O	Mask Status, 0: all, 1: not wearing mask, 2: wearing mask
++++Beard	int	O	Beard Status, 0: all, 1: no beard, 2: has beard
+++GroupID	array<string>	O	GroupID list
+++SimilaryRange	array<int>	O	Similary Range, ex: [40, 100]

[ Response Params ] ( OK )

[ Example ]

Request	<pre>GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&amp;object=08137&amp;condition.Channel=1&amp;condition.StartTime=2014-1-1%2012:00:00&amp;condition.EndTime=2015-1-10%2012:00:00&amp;condition.Types[0]=jpg&amp;condition.Flags[0]=Event&amp;condition.Events[0]=FaceRecognition&amp;condition.DB.FaceRecognitionRecordFilter.RegType=RecSuccess&amp;condition.DB.FaceRecognitionRecordFilter.StartTime=2014-1-1%2012:00:00&amp;condition.DB.FaceRecognitionRecordFilter.EndTime=2015-1-10%2012:00:00&amp;condition.DB.FaceRecognitionRecordFilter.Person.Sex=Male&amp;condition.DB.FaceRecognitionRecordFilter.Person.Country=CN&amp;condition.DB.FaceRecognitionRecordFilter.Person.Age[0]=25&amp;condition.DB.FaceRecognitionRecordFilter.Person.Age[1]=40&amp;condition.DB.FaceRecognitionRecordFilter.Person.Glasses=1</pre>
---------	--

	&condition.DB.FaceRecognitionRecordFilter.GroupID[0]=10001&condition.DB.FaceRecognitionRecordFilter.GroupID[1]=10003&condition.DB.FaceRecognitionRecordFilter.GroupID[2]=10005&condition.DB.FaceRecognitionRecordFilter.SimilarityRange[0]=40&condition.DB.FaceRecognitionRecordFilter.SimilarityRange[1]=100
Response	OK

Step 3 Get the media file information found by the finder

Table 4-173

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"
+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+SummaryNew	array<object>	R	DB record object
++Key	string	R	DB record name, should be <b>"FaceRecognitionRecord"</b>
++Value	object	R	DB record value
+++RecResult	int	O	Recognition result, 0 means recognition failed, no candidates. 1 means recognition success, has candidates.
+++MachineAddress	string	O	Machine address, string max length is 259
+++IsGlobalScene	bool	O	Is the global scene picture or not
+++ImageInfo	object	O	Big picture info
++++Length	int	O	The length of the picture
++++FilePath	string	O	The file path of the picture, max string length is 259
+++Object	object	O	The target face info
++++Sex	string	O	Sex, it can be "Man", "Woman", "Unknown"
++++Age	int	O	Age
++++Glasses	int	O	Glasses Status, 0: unknown, 1: not wear, 2: wear
++++Eye	int	O	Eye status, 0: not detected, 1: close eye, 2: open eye

++++Mouth	int	O	Mouth status, 0: not detected, 1: close mouth, 2: open mouth
++++Mask	int	O	Mask status, 0: not detected, 1: not wearing mask, 2: wearing mask
++++Beard	int	O	Beard status, 0: not detected, 1: no beard, 2: has beard
+++Candidates	array<object>	O	The info of candidates face from face group.
++++Similarity	int	O	Similarity
++++Person	object	O	person info
+++++Name	string	O	Person Name, max string length is 63
+++++Sex	string	O	Sex, it can be "Male", "Female", "Unknown"
+++++Birthday	string	O	The person's birthday, ex: "1980-01-01"
+++++Country	string	O	Country name, length must be 2, value should be according to ISO3166
+++++Province	string	O	Province name, max string length is 63
+++++City	string	O	City name, max string length is 63
+++++CertificateType	string	O	Certificate Type. It can be: "IC", "Passport", "Military", "Unknown"
+++++ID	string	O	Person ID of CertificateType, max string length is 31
+++++Feature State	int	O	Feature State, 0:Unknown, 1:Failed, 2:OK

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100
Response	found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=FaceRecognition items[0].FilePath =/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].Length =790 items[0].SummaryNew[0].Key=FaceRecognitionRecord items[0].SummaryNew[0].Value.RecResult=1 items[0].SummaryNew[0].Value.MachineAddress=BinJiang items[0].SummaryNew[0].Value.IsGlobalScene=true items[0].SummaryNew[0].Value.ImageInfo.Length=123 items[0].SummaryNew[0].Value.ImageInfo.FilePath=/tmp/1.jpg items[0].SummaryNew[0].Value.Object.Sex=Man items[0].SummaryNew[0].Value.Object.Age=40 items[0].SummaryNew[0].Value.Object.Glasses=1 items[0].SummaryNew[0].Value.Object.Eye=2 items[0].SummaryNew[0].Value.Object.Mouth=1 items[0].SummaryNew[0].Value.Candidates[0].Similarity=50 items[0].SummaryNew[0].Value.Candidates[0].Person.Name=ZhangSan items[0].SummaryNew[0].Value.Candidates[0].Person.Birthday=1980-01-01 items[0].SummaryNew[0].Value.Candidates[0].Person.Sex=Male

<pre> items[0].SummaryNew[0].Value.Candidates[0].Person.Country=CN items[0].SummaryNew[0].Value.Candidates[0].Person.Province=ZheJiang items[0].SummaryNew[0].Value.Candidates[0].Person.City=HangZhou items[0].SummaryNew[0].Value.Candidates[0].Person.CertificateType=IC items[0].SummaryNew[0].Value.Candidates[0].Person.ID=1234567890 items[0].SummaryNew[0].Value.Candidates[0].Person.FeatureState=0 ... </pre>
---

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

## 4.11.8 Find media files with HumanTrait info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and HumanTrait conditions with the finder

Table 4-174

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile		
Method	GET		
Description	Start to find media files satisfied the common conditions and HumanTrait conditions with the finder.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	O	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	O	End time to search, ex: "2010-06-25 12:05:00"
+Types	array<string>	O	Search file types, should be "jpg"
+Flags	array<string>	O	Search flags. It can be: "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout" should include "Event"
+Events	array<string>	R	Search event list, must be one string: "HumanTrait"
+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB filter object
++HumanTrait RecordFilter	object	R	DB filter for HumanTrait
+++HumanAttributes	object	R	Human attributes
++++CoatColor	array<string>	O	Coat color, refer to CoatColor in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values, max array size is 4

++++CoatType	array<int>	O	Coat type, refer to CoatType in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values, max array size is 2
++++Trousers Color	array<string>	O	Trousers color, refer to TrousersColor in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values, max array size is 4
++++TrousersType	array<int>	O	Trousers type, refer to TrousersType in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values, max array size is 2
++++HasHat	int	O	Has hat or not, 0: all, 1: not has hat, 2: has hat
++++HasBag	int	O	Has bag or not, 0: all, 1: not has bag, 2: has bag
++++Sex	string	O	Sex, can be "Man", "Woman", if omit, search all
++++Age	array<int>	O	Age range, ex: [25, 40]
++++HairStyle	int	O	Hair style, 0: all, 1: long hair, 2: short hair, 3: ponytail, 4: updo, 5: hidden

[ Response Params ] ( OK )

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condition.Channel=1&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.Types[0]=jpg&condition.Flags[0]=Event&condition.Events[0]=HumanTrait&condition.DB.HumanTraitRecordFilter.HumanAttributes.CoatColor[0]=White&condition.DB.HumanTraitRecordFilter.HumanAttributes.CoatColor[1]=Yellow&condition.DB.HumanTraitRecordFilter.HumanAttributes.CoatType=1&condition.DB.HumanTraitRecordFilter.HumanAttributes.HasHat=2&condition.DB.HumanTraitRecordFilter.HumanAttributes.Sex=Man&condition.DB.HumanTraitRecordFilter.HumanAttributes.Age[0]=30&condition.DB.HumanTraitRecordFilter.HumanAttributes.Age[1]=50&condition.DB.HumanTraitRecordFilter.HumanAttributes.HairStyle=1
Response	OK

Step 3 Get the media file information found by the finder

Table 4-175

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"

+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+SummaryNew	object	R	Face info
++Key	string	R	DB record name, should be <b>"HumanTraitRecord"</b>
++Value	object	R	DB record value
+++HumanAttributes	object	O	Human attributes
++++CoatColor	string	O	Coat color, refer to CoatColor in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values,
++++CoatType	int	O	Coat type, refer to CoatType in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values,
++++TrousersColor	string	O	Trousers color, refer to TrousersColor in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values,
++++TrousersType	int	O	Trousers type, refer to TrousersType in <a href="#">16.1.3.5 [Event] HumanTrait</a> for available values,
++++HasHat	int	O	Has hat or not, 0: unknown, 1: not has hat, 2: has hat
++++HasBag	int	O	Has bag or not, 0: unknown, 1: not has bag, 2: has bag
++++Sex	string	O	Sex, can be "Man", "Woman", "Unknown"
++++Age	int	O	Age
++++HairStyle	int	O	Hair style, 0: unknown, 1: long hair, 2: short hair, 3: ponytail, 4: updo, 5: hidden
++++HasUmbrella	int	O	Has umbrella or not, 0: unknown, 1: not has umbrella, 2: has umbrella
++++Bag	int	O	Bag type, 0: unknown, 1: handbag, 2: shoulder bag, 3: knapsack, 4: draw-bar box
++++Cap	int	O	Cap style, 0: unknown, 1: normal cap, 2: helmet
+++FaceAttributes	object	O	Face attributes
++++Sex	string	O	Sex, can be "Man", "Woman", "Unknown"
++++Age	int	O	Age
++++Mask	int	O	Mask status, 0: not detected, 1: not wearing mask, 2: wearing mask
++++Beard	int	O	Beard status, 0: not detected, 1: no beard, 2: has beard
++++Glass	int	O	Glasses Status, 0: unknown, 1: not wearing, 2: normal Glasses, 3: sun glasses, 4: black frame glasses
++++Emotion	string	O	Emotion info. It can be: "Unknown", "Smile", "Anger", "Sadness", "Disgust", "Fear", "Surprise", "Neutral", "Laugh", "Happy", "Confused", "Scream", "Lookaside"
+++FacePath	string	O	Face picture path, max string length is 259
+++FaceScenePath	string	O	Face scene picture path, max string length is 259
+++HumanPath	string	O	Human picture path, max string length is 259

+++HumanScenePath	string	O	Human scene picture path, max string length is 259
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100		
Response	found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=HumanTrait items[0].FilePath=/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].Length=790 items[0].SummaryNew.Key=HumanTraitRecord items[0].SummaryNew.Value.HumanAttributes.CoatColor=White items[0].SummaryNew.Value.HumanAttributes.CoatType=1 items[0].SummaryNew.Value.HumanAttributes.TrousersColor=Black items[0].SummaryNew.Value.HumanAttributes.TrousersType=1 items[0].SummaryNew.Value.HumanAttributes.HasHat=2 items[0].SummaryNew.Value.HumanAttributes.HasBag=1 items[0].SummaryNew.Value.HumanAttributes.Sex=Man items[0].SummaryNew.Value.HumanAttributes.Age=30 items[0].SummaryNew.Value.HumanAttributes.HairStyle=2 items[0].SummaryNew.Value.HumanAttributes.HasUmbrella=1 items[0].SummaryNew.Value.HumanAttributes.Bag=0 items[0].SummaryNew.Value.HumanAttributes.Cap=2 items[0].SummaryNew.Value.FaceAttributes.Sex=Man items[0].SummaryNew.Value.FaceAttributes.Age=35 items[0].SummaryNew.Value.FaceAttributes.Mask=0 items[0].SummaryNew.Value.FaceAttributes.Beard=1 items[0].SummaryNew.Value.FaceAttributes.Glass=2 items[0].SummaryNew.Value.FaceAttributes.Emotion=Smile items[0].SummaryNew.Value.FacePath=/mnt/2010/8/11/dav/15:40:50.jpg items[0].SummaryNew.Value.FaceScenePath=/mnt/2010/8/11/dav/15:40:51.jpg ...		

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

## 4.11.9 Find media files with TrafficCar info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and TrafficCar conditions with the finder

Table 4-176

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile		
Method	GET		
Description	<p>Start to find media files satisfied the common conditions and TrafficCar conditions with the finder.</p> <p><b>Note:</b> some request params can use compare condition, it's value is an array, first item is a string of compare condition type, and the following items are compare values.</p> <p>Compare condition type can be:</p> <p>"==" : means equal, followed with one param, if param type is string, then param value can contain some "*" to match any string.</p> <p>"  " : means equal one of the params, followed with one or more params.</p> <p>"&lt;&gt;" : means inside range, followed by two integer param,</p> <p>"&gt;&lt;" : means outside range, followed by two integer param,</p>		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	O	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	O	End time to search, ex: "2010-06-25 12:05:00"
+Types	array<string>	O	Search file types
+Flags	array<string>	O	Search flags. It can be: "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout"
+Events	array<string>	O	Search event list Ignored, use Event under DB param.
+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB object
++TrafficCar	object	R	DB filter for TrafficCar
+++PlateNumber	<compare condition>	O	PlateNumber condition, use compare condition format, ex: [ "==", "*888" ]
+++Event	<compare condition>	O	Event condition, ex: [ "  ", "TrafficGate", "Alarm*" ]
+++Speed	<compare condition>	O	Speed condition, ex: [ "<>", 40, 80 ]
+++PlateType	<compare condition>	O	Plate type condition, use compare condition format, ex: [ "==", "Armed"],
+++PlateColor	<compare condition>	O	Plate color condition, use compare condition format, ex: [ "==", "Blue"],
+++VehicleColor	<compare condition>	O	Vehicle color condition, use compare condition format, ex: [ "==", "White"],
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condit		

	ion.Channel=1&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.Types[0]=jpg&condition.Flags[0]=Event&condition.DB.TrafficCar.PlateNumber[0]=%3d%3d&condition.DB.TrafficCar.PlateNumber[1]=%2a888&condition.DB.TrafficCar.Speed[0]=%3c%3e&condition.DB.TrafficCar.Speed[1]=40&condition.DB.TrafficCar.Speed[2]=80&condition.DB.TrafficCar.VehicleColor[0]=%3d%3d&condition.DB.TrafficCar.VehicleColor[1]=White
Response	OK

**Step 3** Get the media file information found by the finder

Table 4-177

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"
+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+Summary	object	R	DB record info
++TrafficCar	object	R	TrafficCar record info
+++PlateNumber	string	R	Car plate number
+++PlateType	string	O	Plate type,
+++PlateColor	string	O	Plate color, ex: "Yellow", "Blue", ... etc
+++VehicleColor	string	O	Vehicle color, ex: "Yellow", "Blue", ... etc
+++Country	string	O	Country info. max string length is 19
+++Speed	int	O	Vehicle speed, unit is km/hour
+++Event	string	O	The event info, ex: "TrafficJunction"
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100		

Response	<pre> found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=TrafficJunction items[0].FilePath=/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].Length=790 items[0].Summary.TrafficCar.PlateNumber=A08888 items[0].Summary.TrafficCar.PlateType=Unknown items[0].Summary.TrafficCar.PlateColor=Blue items[0].Summary.TrafficCar.VehicleColor=White items[0].Summary.TrafficCar.Country=China items[0].Summary.TrafficCar.Speed=70 items[0].Summary.TrafficCar.Event=TrafficJunction ... </pre>
----------	--

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

## 4.11.10 Find media files with IVS info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and IVS conditions with the finder

Table 4-178

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile		
Method	GET		
Description	Start to find media files satisfied the common conditions and IVS conditions with the finder.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	O	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	O	End time to search, ex: "2010-06-25 12:05:00"
+Types	array<string>	O	Search file types
+Flags	array<string>	O	Search flags. It can be: "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout"
+Events	array<string>	O	Search event list,

+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB object
++IVS	object	R	DB filter for IVS
+++Rule	string	O	IVS rule condition. It can be: "CrossLineDetection" , "CrossRegionDetection" , "LeftDetection" , "WanderDetection" , "MoveDetection" , "RioterDetection" , "CrossFenceDetection" , "TakenAwayDetection" , "PasteDetection" , "Preservation" , "StayDetection" , "TailDetection"
+++Action	string	O	IVS action. It can be: "Appear" , "Disappear" , "Inside" , "Cross"
+++ObjectType	array<string>	O	IVS object type, item in array can be: "Unknown" , "Human" , "Vehicle" , "NonMotor"
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condition.Channel=1&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.Types[0]=jpg&condition.DB.IVS.Rule=CrossLineDetection&condition.DB.IVS.Action=Cross&condition.DB.IVS.ObjectType[0]=Human&condition.DB.IVS.ObjectType[1]=NonMotor		
Response	OK		

Step 3 Get the media file information found by the finder

Table 4-179

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"
+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+Summary	object	R	DB record info
++IVS	object	R	IVS record info

+++Rule	string	R	IVS rule, see above for valid value
+++Action	string	O	IVS action, see above for valid value
+++ObjectType	string	O	IVS object type, see above for valid value
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100		
Response	found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=CrossLineDetection items[0].FilePath=/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].Length=790 items[0].Summary.IVS.Rule=CrossLineDetection items[0].Summary.IVS.Action=Cross items[0].Summary.IVS.ObjectType=Human ...		

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

### 4.11.11 Find media files with NonMotor info

**Step 1** Create a media files finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 2** Start to find media files satisfied the common conditions and NonMotor conditions with the finder

Table 4-180

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findFile		
Method	GET		
Description	Start to find media files satisfied the common conditions and NonMotor conditions with the finder.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
condition	object	R	The search conditions
+Channel	int	R	The video channel to search, video channel index start from 1, use -1 to search all video channel.
+StartTime	string	O	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	O	End time to search, ex: "2010-06-25 12:05:00"
+Types	array<string>	O	Search file types
+Flags	array<string>	O	Search flags. It can be: "Timing", "Manual", "Marker", "Event", "Mosaic", "Cutout"

+Events	array<string>	O	Search event list,
+Dirs	array<string>	O	Search directory list, if omit, search all. Each dir path max string length is 259.
+DB	object	R	DB object
++NonMotorRecordFilter	object	R	DB filter for NonMotor
+++NumOfCycling	int	O	Number of people that cycling, value can be 1 ~ 3
+++Color	string	O	NonMotor color, can be "White" "Orange" "Pink" "Black" "Red" "Yellow" "Gray" "Blue" "Green" "Purple" "Brown" "Sliver" "Darkviolet" "Maroon" "Dimgray" "Whitesmoke" "Darkorange" "Mistyrose" "Tomato" "Olive" "Gold" "Darkolivegreen" "Chartreuse" "Greenyellow" "Forestgreen" "Seagreen" "Deepskyblue" "Cyan" "Other"
+++Category	string	O	NonMotor type, can be : "Non-Motor" "Bicycle" "Tricycle" "Motorcycle" "DualTriWheelMotorcycle" "LightMotorcycle" "EmbassyMotorcycle" "MarginalMotorcycle" "AreaoutMotorcycle" "ForeignMotorcycle" "TrialMotorcycle" "CoachMotorcycle"
+++Helmet	int	O	Helmet status, 0 : unknown, 1 : without helmet, 2 : with helmet
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findFile&object=08137&condition.Channel=1&condition.StartTime=2014-1-1%2012:00:00&condition.EndTime=2015-1-10%2012:00:00&condition.Types[0]=jpg&condition.DB.NonMotorRecordFilter.NumOfCycling=2&condition.DB.NonMotorRecordFilter.Color=White&condition.DB.NonMotorRecordFilter.Category=Bicycle&condition.DB.NonMotorRecordFilter.Helmet=1		
Response	OK		

Step 3 Get the media file information found by the finder

Table 4-181

URL	http://<server>/cgi-bin/mediaFileFind.cgi?action=findNextFile		
Method	GET		
Description	Get the media file information found by the finder		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
object	int	R	The finder object id
count	int	R	Record num to get, should less than 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
found	int	R	Record num that found.
items	array<object>	R	Result record items.
+Channel	int	O	The video channel index start from 0,
+StartTime	string	R	Start time to search, ex: "2010-05-25 12:05:00"
+EndTime	string	R	End time to search, ex: "2010-06-25 12:05:00"
+Type	string	O	File type, should be: "jpg"

+Events	array<string>	O	Events with this record
+FilePath	string	O	File path, string max length is 259
+CutLength	int	O	File length that cut between start time and end time
+Length	int	O	File length of the whole file
+SummaryNew	object	R	NonMotor info
++Key	string	R	DB record name, should be "NonMotorRecordFilter"
++Value	object	R	DB record value
+++NumOfCycling	int	O	Number of people that cycling, value can be 1 ~ 3
+++Color	string	O	NonMotor color, can be "White" "Orange" "Pink" "Black" "Red" "Yellow" "Gray" "Blue" "Green" "Purple" "Brown" "Sliver" "Darkviolet" "Maroon" "Dimgray" "Whitesmoke" "Darkorange" "Mistyrose" "Tomato" "Olive" "Gold" "Darkolivegreen" "Chartreuse" "Greenyellow" "Forestgreen" "Seagreen" "Deepskyblue" "Cyan" "Other"
+++Category	string	O	NonMotor type, can be : "Non-Motor" "Bicycle" "Tricycle" "Motorcycle" "DualTriWheelMotorcycle" "LightMotorcycle" "EmbassyMotorcycle" "MarginalMotorcycle" "AreaoutMotorcycle" "ForeignMotorcycle" "TrialMotorcycle" "CoachMotorcycle"
+++Helmet	int	O	Helmet status, 0 : unknown, 1 : without helmet, 2 : with helmet
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/mediaFileFind.cgi?action=findNextFile&object=08137&count=100		
Response	found=100 items[0].Channel=1 items[0].StartTime=2011-1-1 12:00:00 items[0].EndTime=2011-1-1 13:00:00 items[0].Type=jpg items[0].Events[0]=NonMotorDetect items[0].FilePath=/mnt/dvr/sda0/2010/8/11/dav/15:40:50.jpg items[0].Length=790 items[0].SummaryNew.Key=NonMotorRecordFilter items[0].SummaryNew.Value.NumOfCycling=2 items[0].SummaryNew.Value.Color=White items[0].SummaryNew.Value.Category=Bicycle items[0].SummaryNew.Value.Helmet=1 ...		

**Step 4** Close the finder.

This API is the same as the API in "4.11.5 Find Media Files".

**Step 5** Destroy the finder.

This API is the same as the API in "4.11.5 Find Media Files".

## 4.11.12 Download Media File with the File Name

Syntax	http://<server>/cgi-bin/RPC_Loadfile/< <b>Filename</b> >
Method	GET
Description	Download a file by filename. To get filename by chapter file finding.
Example	http://192.168.1.108/cgi-bin/RPC_Loadfile/mnt/sd/2015-01-08/001/dav/19/19.57.12-19.58.25[M][0@0][0].dav
Success Return	HTTP Code: 200 OK Content-Type: Application/octet-stream Content-Length: <fileLength> Body: <data> <data>
Comment	Parameters in URL: <b>Filename</b> : name of media files which would be downloaded.

### 4.11.13 Download Media File between Times

Table 4-183

Syntax	http://<server>/cgi-bin/loadfile.cgi?action=startLoad&channel=< <b>ChannelNo</b> >&startTime=< <b>starttime</b> >&endTime=< <b>endtime</b> >[&subtype=< <b>typeNo</b> >][&Types=< <b>types</b> >]
Method	GET
Description	Download the media data between start time and end time.
Example	http://192.168.1.108/cgi-bin/loadfile.cgi?action=startLoad&channel=1&startTime=2012-10-8%2013:00:01&endTime=2012-10-8%2014:00:01&subtype=0&Types=dav
Success Return	HTTP Code: 200 OK Content-Type: Application/octet-stream Content-Length:<fileLength> Body: <data> <data>
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>typeNo</b> : the stream type, default 0 if not specified. 0-Main Stream 1-Extra Stream 1 2-Extra Stream 2 <b>starttime</b> & <b>endtime</b> : video start time and end time. Time format: yyyy-mm-dd hh:mm:ss <b>types</b> : file type

### 4.11.14 Encrypted Download Media File with the File Name

Table 4-184

Syntax	http://<server>/cgi-bin/RecordStreamInterleaved.cgi?action=attachStream&path=< <b>File name</b> >[&password=< <b>password</b> >]
Method	GET

Description	Encrypted Download a file by filename. To get filename by chapter file finding.
Example	http://172.29.2.241/cgi-bin/RecordStreamInterleaved.cgi?action=attachStream&path=/mnt/sd/2019-07-01/001/dav/12/12.36.16-12.36.26[F][0@0][0].dav
Success Return	HTTP Code: 200 OK Content-Type: Application/octet-stream Content-Length: <fileLength> Body: <data> <data>
Comment	Parameters in URL: <b>Filename</b> : name of media files which would be downloaded. <b>password</b> : password for encrypting media file data. if not set password, then use the preset password

## 4.12 User management

### 4.12.1 Get Information of a Particular User

Table 4-185

Syntax	http://<server>/cgi-bin/userManager.cgi?action=getUserInfo&name=<userName>
Method	GET
Description	Get user information with name <b>userName</b> .
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=getUserInfo&name=admin
Success Return	user.Name=admin user.Memo=admin 's account user.Group=admin user.Reserved=true user.Sharable=true user.AuthList=<authList>
Comment	—

### 4.12.2 Get Information of All Users

Table 4-186

Syntax	http://<server>/cgi-bin/userManager.cgi?action=getUserInfoAll
Method	GET
Description	Get information of all users.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=getUserInfoAll
Success Return	users[0].Group=admin users[0].Id=1 users[0].Memo=admin 's account users[0].Name=admin users[0].Reserved=true users[0].Sharable=true

	users[0]. AuthList=<authList> users[1].Group=admin ...
Comment	—

### 4.12.3 Get Information of All Active Users

Table 4-187

Syntax	http://<server>/cgi-bin/userManager.cgi?action=getActiveUserInfoAll
Method	GET
Description	Get active users.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=getActiveUserInfoAll
Success Return	users[0].name=admin users[0].ip=10.43.2.16 users[0].group=admin users[0].clienttype=web3.0 users[0].logintime=2011-11-08 09:51:03
Comment	—

### 4.12.4 Get Information of a Particular Group

Table 4-188

Syntax	http://<server>/cgi-bin/userManager.cgi?action=getGroupInfo&name=< <b>groupName</b> >
Method	GET
Description	Get group setting with name <b>groupName</b> .
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=getGroupInfo&name=admin
Success Return	group.Name=admin group.Memo=administrator group group.AuthorityList=<authList>
Comment	Parameters in URL: The device has one or two default user groups: "admin" or "admin" and "user". The "admin" group has all the authorities of operating the device. The "user" group only has monitoring and replaying authorities. <b>groupName</b> : name of the group. If the group named groupName does not exist, the device returns Error.

### 4.12.5 Get Information of All Groups

Table 4-189

Syntax	http://<server>/cgi-bin/userManager.cgi?action=getGroupInfoAll
Method	GET
Description	Get information of all groups.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=getGroupInfoAll

Success Return	<pre> group[0].Id=1 group[0].Name=admin group[0].Memo=administrator group group[0].AuthorityList=&lt;authList&gt; group[0].AuthorityDetail.Bypass[0]=1 group[0].AuthorityDetail.Bypass[1]=2 group[0].AuthorityDetail.Bypass[2]=3 ... group[0].AuthorityDetail.AccessControl[0]=1 group[0].AuthorityDetail.AccessControl[1]=3 group[0].AuthorityDetail.AccessControl[2]=5 ... group[1].Id=1 group[1].Name=user group[1].Memo=user group group[1].AuthorityList=&lt;authList&gt; group[1]..... ... </pre>
Comment	—

#### 4.12.6 Add a New User

Access Control products are not supported.

Table 4-190

Syntax	<pre> http://&lt;server&gt;/cgi-bin/userManager.cgi?action=addUser&amp;user.Name=&lt;userName&gt;&amp;user.Password=&lt;userPassword&gt;&amp;user.Group=&lt;userGroup&gt;&amp;user.Sharable=&lt;userSharable&gt;&amp;user.Memo=&lt;userMemo&gt;&amp;user.Reserved=&lt;userReserved&gt;] </pre>
Method	GET
Description	Add a user.
Example	<pre> http://192.168.1.108/cgi-bin/userManager.cgi?action=addUser&amp;user.Name=George&amp;user.Password=123456&amp;user.Group=user&amp;user.Sharable=true&amp;user.Reserved=false </pre>
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>userGroup</b>: string, the range is "admin" and "user". In different group, the user has different authorities.</p> <p><b>userSharable</b>: bool, true means allow multi-point login.</p> <p><b>userReserved</b>: bool, true means this user can't be deleted.</p>

#### 4.12.7 Delete a User

Access Control products are not supported.

Table 4-191

Syntax	http://<server>/cgi-bin/userManager.cgi?action=deleteUser&name=<userName>
Method	GET
Description	Delete user with name <b>username</b> .
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=deleteUser&name=George
Success Return	OK
Comment	—

## 4.12.8 Modify User Information

Access Control products are not supported.

Table 4-192

Syntax	http://<server>/cgi-bin/userManager.cgi?action=modifyUser&name=<UserName>&user.Memo=<userMemo>&user.Group=<userGroup>&user.Reserved=<userReserved>&user.Sharable=<userSharable>
Method	GET
Description	Modify user info.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=modifyUser&name=George&user.Group=admin
Success Return	OK
Comment	User is identified by <UserName>, other params are the same with AddUser.

## 4.12.9 Modify User's Password

Access Control products are not supported.

Table 4-193

Syntax	http://<server>/cgi-bin/userManager.cgi?action=modifyPassword&name=<username>&pwd=<newPwd>&pwdOld=<oldPwd>
Method	GET
Description	Modify user password.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=modifyPassword&name=George&pwd=abcdef&pwdOld=123456
Success Return	OK
Comment	Old password <b>oldPwd</b> should be supplied, new password is <b>newPwd</b> .

## 4.12.10 Manager Modify Common User's Password

Access Control products are not supported.

Table 4-194

Syntax	http://<server>/cgi-bin/userManager.cgi?action=modifyPasswordByManager&username=<username>&pwd=<newPwd>&managerName=<managerName>&managerPwd=<managerPwd>&accountType=<accountType>
Method	GET
Description	Manager Modify common user's password.
Example	http://192.168.1.108/cgi-bin/userManager.cgi?action=modifyPasswordByManager&username=Geoge&pwd=123456&managerName=admin&managerPwd=abc123&accountType=0
Success Return	OK
Comment	the name of common user to be modified is username new password is newPwd manager name is only admin manager password is managerPwd, that is password of admin accountType support 0 is Dahua account, 1 is Onvif account

## 4.13 Log

### 4.13.1 Find Logs

- Whether or not found logs satisfied the conditions

Table 4-195

Syntax	http://<server>/cgi-bin/log.cgi?action=startFind&condition.StartTime=<start>&condition.EndTime=<end>[&condition.Type=<type>]
Method	GET
Description	Start to find log.
Example	Find log between 2011-1-1 12:00:00 and 2011-1-10 12:00:00, URL is: http://192.168.1.108/cgi-bin/log.cgi?action=startFind&condition.StartTime=2011-1-1%2012:00:00&condition.EndTime=2011-1-10%2012:00:00
Success Return	token=1 count=100
Comment	Parameters in URL: <b>start/end</b> : the start/end time of log. Format is: yyyy-mm-dd hh:mm:ss. In response, there is a token for further log finding process. If token is greater than 0, logs are found; otherwise no logs are found. <b>Type</b> : log type. The range is { "System", "Config", "Event", "Storage", "Account", "Data",

	"File", "CourseRecord" }. <b>count</b> : The logcount for the search condition.
--	--

2. Get the particular number of logs

Table 4-196

Syntax	http://<server>/cgi-bin/log.cgi?action=doFind&token=< <b>TokenValue</b> >&count=< <b>logCount</b> >
Method	GET
Description	Find log with token TokenValue and count logCount.
Example	http://192.168.1.108/cgi-bin/log.cgi?action=doFind&token=1&count=100
Success Return	<b>found=2</b> items[0]. <b>RecNo</b> =789 items[0]. <b>Time</b> =2011-05-20 11:59:10 items[0]. <b>Type</b> =ClearLog items[0]. <b>User</b> =admin items[1]. <b>Detail</b> .Compression=H.264->MJPEG items[1]. <b>Detail</b> .Data=Encode items[1]. <b>RecNo</b> =790 items[1]. <b>Time</b> =2011-05-20 11:59:21 items[1]. <b>Type</b> =SaveConfig items[1]. <b>User</b> =System ...
Comment	Parameters in URL: The <b>TokenValue</b> is got by startFind in the above section, and <b>logCount</b> is the count of logs for this query. The maximum value of <b>logCount</b> is 100.

**Appendix:**

Field in Response	Description
found	Count of found log, found is 0 if no log is found.
User	User name.
Type	Log type.
Time	Time of this log.
RecNo	Log number.
Detail	Log details.

3. Stop query logs

Table 4-197

Syntax	http://<server>/cgi-bin/log.cgi?action=stopFind&token=< <b>TokenValue</b> >
Method	GET
Description	Stop query log by token <b>TokenValue</b> .
Example	http://192.168.1.108/cgi-bin/log.cgi?action=stopFind&token=1
Success Return	OK
Comment	Parameters in URL:

The **TokenValue** is got by startFind in above section

## 4.13.2 Clear All the Logs

Table 4-198

Syntax	http://<server>/cgi-bin/log.cgi?action=clear
Method	GET
Description	Clear all the logs.
Example	http://192.168.1.108/cgi-bin/log.cgi?action=clear
Success Return	OK
Comment	—

## 4.13.3 Backup Logs

Table 4-199

Syntax	http://<server>/cgi-bin/Log.backup?action=All&condition.StartTime=< <b>startTime</b> >&condition.EndTime=< <b>endTime</b> >
Method	GET
Description	Download the log information between the start time and the end time as a file named Log.Backup default.
Example	http://192.168.1.108/cgi-bin/Log.backup?action=All&condition.StartTime=2014-8-25%2000:02:32&condition.EndTime=2020-8-25%2001:02:32
Success Return	HTTP/1.1 200 OK CONTENT-LENGTH: 743087 CONNECTION: close Content-type: application/binarytet-stream; charset=utf-8  &w_User: default &Time: 2014-09-01 15:20:45 &Type: VideoLoss &Content: EventType: VideoLoss channel: <8> StartTime: 2014-09-01 15:20:45  ...
Comment	Parameters in URL: <b>startTime/endTime</b> : the start/end time when log info built. 24 hour Format, as: yyyy-mm-dd hh:mm:ss. For example: 2014-8-25 00:02:32 2020-8-25 01:02:32

## 4.13.4 Seek Find Logs

Syntax	http://<server>/cgi-bin/log.cgi?action=doSeekFind&token=< <b>TokenValue</b> >&offset=< <b>offsetValue</b> >&count=< <b>logCount</b> >
Method	GET
Description	Find log with token TokenValue, offset offsetValue and count logCount.
Example	http://192.168.1.108/cgi-bin/log.cgi?action=doSeekFind&token=46878&offset=200&count=200
Success Return	<pre> <b>found</b>=2 items[0].<b>RecNo</b>=789 items[0].<b>Time</b>=2011-05-20 11:59:10 items[0].<b>Type</b>=ClearLog items[0].<b>User</b>=admin items[0].<b>Detail</b>.Compression=H.264-&gt;MJPG items[0].<b>Detail</b>.Data=Encode items[1].<b>RecNo</b>=790 items[1].<b>Time</b>=2011-05-20 11:59:21 items[1].<b>Type</b>=SaveConfig items[1].<b>User</b>=System items[1].<b>Detail</b>.Compression=H.264-&gt;MJPG items[1].<b>Detail</b>.Data=Encode ... </pre>
Comment	<p>Parameters in URL:</p> <p>The <b>TokenValue</b> is got by startFind in the above section, and <b>logCount</b> is the count of logs for this query.<b>offsetValue</b> is the number which count from the first matched.</p> <p>The maximum value of <b>logCount</b> is 100.</p>

## 4.13.5 Export Encrypted Log

Syntax	http://<server>/cgi-bin/Log.exportEncryptedLog?action=All&condition.StartTime=< <b>startTime</b> >&condition.EndTime=< <b>endTime</b> >&condition.Password=< <b>password</b> >
Method	GET
Description	Get encrypted log in zip format
Example	http://192.168.1.108/cgi-bin/Log.exportEncryptedLog?action=All&condition.StartTime=2014-8-25%2000:02:32&condition.EndTime=2020-8-25%2001:02:32&condition.Password=12345
Success Return	<pre> HTTP/1.1 200 OK Transfer-Encoding: chunked CONNECTION: keep_alive Content-type: application/binarytet-stream; charset=utf-8 The binary data of encrypted log in zip format </pre>
Comment	StartTime/EndTime: the start/end time when log info built. 24 hour Format, as: yyyy-mm-dd hh:mm:ss.

	<p>For example:  2014-8-25 00:02:32  2020-8-25 01:02:32  The Password is the input parameter for encrypted log in zip format.  The binary data should be saved as .zip forma</p>
--	--

## 4.14 Upgrader

### 4.14.1 Strat to Upgrade

Table 4-200

Syntax	<code>http://&lt;server&gt;/cgi-bin/upgrader.cgi?action=uploadFirmware</code>
Method	POST
Description	Use this message to upload the firmware, and when the device receiving all the data successfully, it will start to upgrade the device, and then use the getState method to get the state.
Example	<pre>POST /cgi-bin/upgrader.cgi?action=uploadFirmware HTTP/1.1 Host: 192.168.1.108 Connection: keep-alive Content-Type: multipart/form-data; boundary=-----8655433224198 Content-Length: xxxxxxxx  -----8655433224198 Content-Disposition:form-data;name="upgrade"; filename= "xxxxxx.bin" Content-Type: application/octet-stream  Firmware data...  -----8655433224198--</pre>
Success Return	OK
Comment	—

### 4.14.2 Get Upgrade State

Table 4-201

Syntax	<code>http://&lt;server&gt;/cgi-bin/upgrader.cgi?action=getState</code>
Method	GET
Description	Use this message to upload the firmware, and when the device receiving all the data successfully, it will start to upgrade the device, and then use the getState method to get the state.
Example	<code>http://192.168.1.108/cgi-bin/upgrader.cgi?action=getState</code>
Success	state. <b>State</b> =Upgrading

Return	state. <b>Progress</b> =45
Comment	Parameters in Response: <b>State</b> : the state of the upgrade, it can be Preparing, Downloading, DownloadFailed, Upgrading, Invalid, Failed, Succeeded, Cancelled, and NotEnoughMemory. <b>Progress</b> : the progress of the upgrade.

### 4.14.3 Set upgrader url

url with Upgrade packet address

- Request

<b>Syntax</b>	http://<server>/cgi-bin/upgrader.cgi?action=updateFirmwareByUrl[&Url=<Url>][&checkType=<checkType>][&checksum=<checksum>]			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
Url	string	Y	Upgrade package address, also support ftp address	https://aaa/bbb/ccc/license.bin
checkType	int	N	check method enumint8{ 0: MD5 }	0
checksum	string	N	Upgrade package checksum	f38ad920
<b>Example</b>				
http://<server>/cgi-bin/upgrader.cgi?action=updateFirmwareByUrl=http://yfssp.dahuotech.com/#/approval/dhReview/browse/210617165614&checkType=0&checksum=f38ad920				

- Response

<b>Parameter Format</b>	OK in response			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>	OK			

### 4.14.4 Cancel Upgrade

- Request

<b>Syntax</b>	http://<server>/cgi-bin/upgrader.cgi?action=cancel			
<b>Method</b>	GET			
<b>参数格式</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>				
http://<server>/cgi-bin/upgrader.cgi?action=cancel				

- Response

<b>Parameter Format</b>	OK in response			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>				
OK				

## 4.14.5 Checking Cloud Update Version

Check cloud update version

<b>Request URL</b>	http://<server>/cgi-bin/api/CloudUpgrader/check			
<b>Method</b>	POST			
<b>Request Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
way	int32	R	Checking methods. enumint8{ 0: Updating by updating server through direct connection. 1: Checking through proxy server. 2: Detecting by acquiring cached test results. }	0
proxy	object	O	Proxy server address, valid when way==1 (optional).	
+IP	char[40]	R	Network address	"10.1.2.3"
+Port	int32	O	Port	8080
<b>Request Example</b>				
<pre>{   "way" : 0,   "proxy" : {     "IP" : "10.1.2.3",     "Port" : 8080   } }</pre>				

<b>Response Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
info	object	O	Test results	
Status	enumchar[12 ]	R	Update status. enumchar[12]{ "None": No update is detected "Regular": Regular update (It requires user	"None"

			confirmation, and can only be updated to a later version). "Emergency": Mandatory update (The device automatically performs detection and update, and it can also degrade to an earlier version). "Automatic": Automatic update (The device automatically updates when a new version is available. It is currently for custom use and needs to be enabled)}	
+PackageType	enumchar[8]	O	Update package type of new version. When State is not None, return enumchar[8]{ "all": All package. "ptz": PTZ main control package. "web" "logo" "custom" "gui" "pd" "data" "ptz_power": PTZ power. "ptz_light": PTZ light. "ptz_heater": PTZ heater. }	"all"
+OldVersion	char[64]	O	Old version, which needs to be returned when State is not None.	"0000"
+NewVersion	char[64]	O	New version, which needs to be returned when State is not None.	"0004"
+Attention	char[2048]	O	Updated content of the new update package.	"What is new"
+PackageUrl	char[256]	O	Download address of update package (required for agent upgrade).	"https://lechange.com/1.zip"
+PackageId	char[64]	O	Update package ID	"1d2ee7"
+Checksum	char[64]	O	SHA-256 checksum of the update package	"F3D288AB"
+BuildTime	char[24]	O	Build time of update package	"08-10-2018 01:01:02"

#### Response Example

```
{
  "info" : {
    "State" : "None",
    "PackageType" : "all",
    "OldVersion" : "0000",
```

```

    "NewVersion" : "0004",
    "Attention" : "What is new",
    "PackageUrl" : "https://lechange.com/1.zip",
    "Packageld" : "1d2ee7",
    "Checksum" : "F3D288AB"
    "BuildTime" : "08-10-2018 01:01:02"
  }
}

```

## 4.14.6 Performing Online Update

Perform online update

<b>Request URL</b>	http://<server>/cgi-bin/api/CloudUpgrader/execute			
<b>Method</b>	POST			
<b>Request Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
NewVersion	char[64]	O	If the cloud version is newer than the new version obtained in the last check, the update fails.	"2.42.00.001"
way	enumint8	R	Update method enumint8{ 0: Updating by updating server through direct connection. 1: Updating by proxy server}.	0
proxy	object	O	Proxy server address, which is valid when way=1 (optional).	
+IP	char[40]	O	Network address	"10.1.2.3"
+Port	uint16	O	Port	8080
info	object	O	Update package information, which is needed when way==1	

			(optional).	
+PackageUrl	char[256]	O	Download address of the update package (required for agent upgrade).	"https://lechange.com/1.zip"
+PackageId	char[64]	O	Update package ID	"1d2ee7"
+Checksum	char[64]	O	SHA-256 checksum of the update package	"F3D288AB"

#### Request Example

```
{
  "NewVersion" : "2.42.00.001",
  "way" : 0,
  "proxy" : {
    "IP" : "10.1.2.3",
    "Port" : 8080
  },
  "info" : {
    "PackageUrl" : "https://lechange.com/1.zip",
    "PackageId" : "1d2ee7",
    "Checksum" : "F3D288AB"
  }
}
```

#### Response Params ( JSON format in body )

Name	Type	R/O	Description	Example
<b>Response Example</b>				
{}				

### 4.14.7 Canceling Online Update

Cancel the online update during the download process. If you have already started writing Flash, you cannot cancel the update.

<b>Request URL</b>	http://<server>/cgi-bin/api/CloudUpgrader/cancel			
<b>Method</b>	GET			
<b>Request Params (JSON format in body)</b>				
Name	Type	R/O	Description	Example
<b>Request Example</b>				
{}				

Response Params ( JSON format in body )				
Name	Type	R/O	Description	Example
Response Example				
{ }				

## 4.15 Wiper

### 4.15.1 Move Continuously

Table 4-202

Syntax	http://<server>/cgi-bin/rainBrush.cgi?action=moveContinuously&interval=< <b>Second</b> >[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Control the wiper to move continuously.
Example	http://192.168.1.108/cgi-bin/rainBrush.cgi?action=moveContinuously&interval=5
Success Return	OK
Comment	<b>Second</b> : integer, rain brush movement time interval which start from 1. <b>ChannelNo</b> : integer, video channel index which starts from 1, default 1 if not specified.

### 4.15.2 Stop Move

Table 4-203

Syntax	http://<server>/cgi-bin/rainBrush.cgi?action=stopMove[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Control the wiper to stop moving.
Example	http://192.168.1.108/cgi-bin/rainBrush.cgi?action=stopMove
Success Return	OK
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1, default 1 if not specified.

### 4.15.3 Move Once

Table 4-204

Syntax	http://<server>/cgi-bin/rainBrush.cgi?action=moveOnce[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Control the wiper to move once.
Example	http://192.168.1.108/cgi-bin/rainBrush.cgi?action=moveOnce
Success Return	OK
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1, default 1 if not specified.

## 4.16 Security

### 4.16.1 Set User Login Authentication Policy

Table 4-205

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set User Login Authenticate Policy
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&LoginAuthCtrl.PriSvrPolicy=2
Success Return	OK
Comment	<b>LoginAuthCtrl.PriSvrPolicy</b> : DVRIP login policy setting for Dahua Private Protocol. 1 - safe mode, only support digest authentication. (Recommended.); 2 - compatible mode for traditional devices;

## 5.1 Image

### 5.1.1 Brightness, Contrast and Saturation

- Get brightness, contrast and saturation

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoColor
Method	GET
Description	Get brightness, contrast and saturation
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoColor
Success Return	<b>head.Name=Day,</b> <b>head.Brightness=50,</b> <b>head.Contrast=50,</b> <b>head.Saturation=50,</b> <b>head.Hue=50,</b> <b>head.Gamma=50,</b> <b>head.ChromaSuppress=50,</b> <b>head.Style=Standard,</b> <b>head.TimeSection=1 00:00:00-24:00:00</b>
Comment	Parameters in URL: paramName and paramValue are as table below. In table below, <b>head</b> = table.VideoInSharpness [ <b>ChannelNo</b> ] [ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

- Set brightness, contrast and saturation

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue> ...]
Method	SET
Description	Set brightness, contrast and saturation
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoColor[0][0].Brightness=50
Success Return	OK
Comment	Parameters in URL: paramName and paramValue are as table below. In table below,

	<p><b>head</b> = table.VideoInSharpness [<b>ChannelNo</b>] [<b>ConfigNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>
--	---

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> .Brightness	integer	Brightness, range is [0—100]
<b>head</b> .Contrast	integer	Contrast, range is [0—100]
<b>head</b> .Saturation	integer	Saturation, range is [0—100]

### 5.1.2 Sharpness

- Get sharpness

Table 5-1

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInSharpness
Method	GET
Description	Get sharpness
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInSharpness
Success Return	head.Level=4 head.Sharpness=8
Comment	<p>Parameters in URL: paramName and paramValue are as table below. In table below, <b>head</b> = table.VideoInSharpness [<b>ChannelNo</b>] [<b>ConfigNo</b>] <b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>

- Set sharpness

Table 5-2

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set sharpness
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInSharpness[0][0].Level=10&VideoInSharpness[0][0].Mode=1&VideoInSharpness[0][0].Sharpness=0
Success Return	OK
Comment	<p>Parameters in URL: paramName and paramValue are as table below.</p>

	<p>In table below,</p> <p><b>head</b> = VideoInSharpness [<b>ChannelNo</b>] [<b>ConfigNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>
--	--

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> .Sharpness	integer	Range is 0—100
<b>head</b> .Level	integer	Range is 0—100

### 5.1.3 Flip, Mirror and Rotate90

- Get flip, mirror and Rotate90

Table 5-3

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoImageControl
Method	GET
Description	Get flip, mirror and Rotate90
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoImageControl
Success Return	head.Flip=true head.Mirror=false head.Rotate90=0
Comment	<p>Parameters in Response:</p> <p><b>head</b> = table.VideoImageControl [<b>ChannelNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>

- Set flip, mirror and Rotate90

Table 5-4

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set flip, mirror and Rotate90
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoImageControl[0].Flip=true
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>head</b> = VideoImageControl [<b>ChannelNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to</p>

	video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.
--	---

**Appendix:**

ParamName	ParamValue type	Description
<b>head.Flip</b>	bool	true: enable video flip function false: disable video flip function
<b>head.Mirror</b>	bool	true: enable video mirror function false: disable video mirror function
<b>head.Rotate90</b>	integer	Range is {0,1,2} Video rotation: 0: No rotate 1: clockwise rotate 90° 2: anticlockwise rotate 90°

## 5.2 Exposure

### 5.2.1 Exposure Config

- Get exposure

Table 5-5

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInExposure
Method	GET
Description	Get exposure
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInExposure
Success Return	head.AntiFlicker=0 head.Gain=50 head.GainMax=50 head.GainMin=0 head.Iris=50 head.IrisAuto=false head.Mode=0 head.Value1=40 head.Value2=40
Comment	Parameters in Response: <b>head</b> = table.VideoInExposure[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

- Set exposure

Table 5-6

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <i>paramName</i> >=< <i>paramValue</i> >[&< <i>paramName</i> >=< <i>paramValue</i> >...]
Method	SET
Description	Set exposure
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInExposure[0][0].Iris=50
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>head</b> = VideoInExposure[<b>ChannelNo</b>][<b>ConfigNo</b>]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>

## Appendix:

ParamName	ParamValue type	Description
<b>head</b> .AntiFlicker	integer	Range is {0,1,2} AntiFlicker mode: 0: Outdoor 1: 50 Hz AntiFlicker 2: 60 Hz AntiFlicker
<b>head</b> .Gain	integer	Range is [0—100] If GainAuto is true, it's upper limit of auto gain, else it's the fixed gain adjust value.
<b>head</b> .GainMax	integer	Range is 0–100; the value must be greater than GainMin.
<b>head</b> .GainMin	integer	Range is 0–100; the value must be smaller than GainMax.
<b>head</b> .Iris	integer	Manual Iris setting. Range is 0–100.
<b>head</b> .IrisAuto	bool	Enable Iris automatically. true: IrisAuto <b>false: No IrisAuto</b>
<b>head</b> .Mode	integer	0: "Auto" by default 1: Low noise 2: Anti-smear 4: Manual ( <b>range</b> ) 5: <b>Aperture priority</b> 6: Manual (fixed) 7: <b>Gain priority</b> 8: <b>Shutter priority</b> 9: Flash light matching mode
<b>head</b> .Value1	float	Range is [0-1000], unit is millisecond If ExposureSpeed is 0(AutoExposure enable), it's

ParamName	ParamValue type	Description
		lower limit of AutoExposure time, otherwise it's time of manualExposure
<b>head.Value2</b>	float	Range is [0-1000], unit is millisecond Upper limit of AutoExposure time, should be bigger than ExposureValue1

## 5.3 Backlight

### 5.3.1 Backlight Config

- Get backlight

Table 5-7

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInBacklight
Method	GET
Description	Get backlight
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInBacklight
Success Return	head.GlareInhibition=50 head.Mode=Off head.WideDynamicRange=50
Comment	Parameters in Response: <b>head</b> = table.VideoInBacklight[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

- Set backlight

Table 5-8

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set backlight
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInBacklight[0][0].GlareInhibition=50
Success Return	OK
Comment	Parameters in URL: <b>head</b> = VideoInBacklight[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

## Appendix:

ParamName	ParamValue type	Description
<b>head</b> .GlareInhibition	integer	Range is 1–100.
<b>head</b> .Mode	char[32]	Off: Switched off Backlight: Backlight compensation GlareInhibition: HLC WideDynamic: WDR SSA: Scene adaptation
<b>head</b> .WideDynamicRange	integer	Range is 1–100.

## 5.4 White Balance

### 5.4.1 White Balance Config

- Get white balance

Table 5-9

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInWhiteBalance
Method	GET
Description	Get white balance
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInWhiteBalance
Success Return	head.GainBlue=50 head.GainRed=50 head.Mode=Auto
Comment	Parameters in Response: <b>head</b> = table.VideoInWhiteBalance[ <b>ChannelNo</b> ][ <b>ColorConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

- Set white balance

Table 5-10

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set white balance
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInWhiteBalance[0][0].GainBlue=50
Success Return	OK

Comment	Parameters in URL: <b>head</b> = VideoInWhiteBalance[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.
---------	---

**Appendix:**

ParamName	ParamValue type	Description
<b>head</b> .GainBlue	integer	Range is [0—100] Gain for blue value, Value is effective when WhiteBalance is "Custom."
<b>head</b> .GainRed	integer	Range is [0—100] Gain for red value, Value is effective when WhiteBalance is "Custom."
<b>head</b> .Mode	char[[32]	Auto; Indoor; Outdoor; ATW; Manual; Sodium; Natural; StreetLamp; ManualDatum

## 5.5 Day-Night

### 5.5.1 Day-Night Config

- **Get day-night**

Table 5-11

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInDayNight
Method	GET
Description	Get day-night
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInDayNight
Success Return	head.Delay=10 head.Mode=Brightness head.Sensitivity=2 head.Type=Mechanism
Comment	<b>head</b> = table.VideoInDayNight[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ]: <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.

- **Set Day-Night**

Table 5-12

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
--------	--

Method	SET
Description	Set day-night
Example	<a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInDayNight[0][0].Mode=BlackWhite">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInDayNight[0][0].Mode=BlackWhite</a>
Success Return	OK
Comment	Parameters in URL: ParamName and paramValue are as table below. In table below, <b>head</b> = VideoInDayNight[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ]

#### Appendix:

ParamName	ParamValue type	Description
<b>head.Type</b>	string	The range is {"Electron", "Mechanism", "NightICR", "Auto"}, the way of ICR switching.
<b>head.Mode</b>	string	The range is {"Color", "Brightness", "BlackWhite", "Photoresistor", "Gain"}. "Color": Always "color" "Brightness": Day/Night Auto "BlackWhite": Always black-and-white "Photoresistor": Switch according to photoresistor, "Gain": Switch according to gain
<b>head.Sensitivity</b>	integer	Range is [1-3]. Sensitivity of switching mode
<b>head.Delay</b>	integer	Range is [2-10]. Delay seconds when switching mode.

## 5.6 Zoom and Focus

To get the capability set of video input, refer to 4.5.12. For instance, you can use the following URL:

<http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getCaps&channel=1>

If ElectricFocus or SyncFocus of the return value is true, use 5.6.1 ~ 5.6.4; Otherwise, use 5.6.5 ~ 5.6.6 .

### 5.6.1 Adjust Focus

Table 5-13

Syntax	<a href="http://&lt;server&gt;/cgi-bin/devVideoInput.cgi?action=adjustFocus&amp;focus=&lt;zoomNo&gt;[&amp;channel=&lt;ChannelNo&gt;]">http://&lt;server&gt;/cgi-bin/devVideoInput.cgi?action=adjustFocus&amp;focus=&lt;zoomNo&gt;[&amp;channel=&lt;ChannelNo&gt;]</a>
Method	GET
Description	Ajust magnification and the focus.
Example	<a href="http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=adjustFocus&amp;focus=0.5&amp;zoom=-0.5">http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=adjustFocus&amp;focus=0.5&amp;zoom=-0.5</a>
Success Return	OK
Comment	Parameters in URL: <b>focusNo</b> : float, the range is between 0 and 1; -1 means reset to position 0. <b>zoomNo</b> : float, the range is between 0 and 1; -1 means reset to position 0. <b>ChannelNo</b> : integer, video channel index which starts from 1.

## 5.6.2 Adjust Focus Continuously

Table 5-14

Syntax	<code>http://&lt;server&gt;/cgi-bin/devVideoInput.cgi?action=adjustFocusContinuously&amp;focus=&lt;focusNo&gt;&amp;zoom=&lt;zoomNo&gt;[&amp;channel=&lt;ChannelNo&gt;]</code>
Method	GET
Description	Adjust magnification and the focus continuously.
Example	<p>If we want to adjust focus, the API like this:  <code>http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=adjustFocusContinuously&amp;focus=0.02&amp;zoom=-1</code>            and when the motor is moving, we send below command to let it stop:  <code>http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=adjustFocusContinuously&amp;focus=0&amp;zoom=-1</code></p>
Success Return	OK
Comment	<p>Parameters in URL:  <b>focusNo</b>: float, the range is <math>-1 &lt; \text{focus} &lt; 1</math>; 0 means stop.  <b>zoomNo</b>: float, the range is <math>-1 &lt; \text{zoom} &lt; 1</math>; 0 means stop.  <b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p>The value means the moving speed of motor lens, positive value means move forwards, negative value means move backwards.            This command is used to drive the lens move continuously, until it reaches end.            When the motor is moving, you can send this command again with "focus" or "zoom" parameter as 0 to stop it immediately.            In this command, when you adjust the focus parameter, the zoom parameter should be -1, and the focus parameter should be -1 when adjust the zoom parameter.</p>

## 5.6.3 Auto Focus

Table 5-15

Syntax	<code>http://&lt;server&gt;/cgi-bin/devVideoInput.cgi?action=autoFocus[&amp;channel=&lt;ChannelNo&gt;]</code>
Method	GET
Description	Auto focus.
Example	<code>http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=autoFocus</code>
Success Return	OK
Comment	<p>Parameters in URL:  <b>ChannelNo</b>: integer, video channel index which starts from 1.</p>

## 5.6.4 Get Focus Status

Table 5-16

Syntax	http://<server>/cgi-bin/devVideoInput.cgi?action=getFocusStatus[&channel=< <b>ChannelNo</b> >]
Method	GET
Description	Get device focus status.
Example	http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getFocusStatus
Success Return	status.Focus=0.5 status.Zoom=0.5 status.Status=Normal
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. Parameters in Response: The range of status. <b>Status</b> is "Normal" and "Autofocus". This command must be continual executed until status.Status is "Normal".

## 5.6.5 Zoom Config

- Get zoom

Table 5-17

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInZoom
Method	GET
Description	Get zoom
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInZoom
Success Return	head.DigitalZoom=true head.Speed=7
Comment	Parameters in URL: <b>head</b> = table.VideoInZoom[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b> : array index, can be 0, 1 or 2. 0 means config for day, 1 means

- Set zoom

Table 5-18

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set zoom
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInZoom[0][0].DigitalZoom=false&VideoInZoom[0][0].Speed=8
Success Return	OK
Comment	Parameters in URL: <b>head</b> = VideoInZoom[ <b>ChannelNo</b> ][ <b>ConfigNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to

<p>video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo:</b> array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>
---

**Appendix:**

ParamName	ParamValue type	Description
<i>head</i> .DigitalZoom	integer	true: Enable Digital Zoom false: Disable Digital Zoom
<i>head</i> .Speed	integer	Range is 0—100

### 5.6.6 Focus Config

- Get focus

Table 5-19

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInFocus
Method	GET
Description	Get focus
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInFocus
Success Return	head.Name=Day, head.Mode=0, head.Sensitivity=1, head.FocusLimitSelectMode=Manual, head.FocusLimit=2000, head.FocusFarLimit=5000, head.AutoFocusTrace=0, head.IRCorrection=0
Comment	—

- Set focus

Table 5-20

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	SET
Description	Set focus
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoInFocus[0][0].FocusLimit=1000
Success Return	OK

Comment	<p>Parameters in URL: paramName and paramValue are as table below.</p> <p>In table below, <b>head</b> = VideoInFocus [<b>ChannelNo</b>] [<b>ConfigNo</b>] <b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>
---------	--

## Appendix:

ParamName	ParamValue type	Description
<b>head</b> .Mode	integer	2—Auto focus, 3—Half auto focus, 4—Manual focus
<b>head</b> .Name	string	Scene name
<b>head</b> .FocusLimit	integer	100, 1000, 2000, 3000, 5000. If xxxx1 is included in the queue, replace 1 with 0, and add + sign at the end; for instance, 200001 will be displayed as 200000mm+. Get the "FocusLimit" capacity to determine the value range.
<b>head</b> .FocusFarLimit	integer	Customized requirement. Remote focus limit recommendation, unit: mm
<b>head</b> .AutoFocusTrace	integer	focus trace 0 for off , 1 for on
<b>head</b> .IRCorrection	integer	IR Correction 0: don't correct 1: correct manually 2: auto correct
<b>head</b> .Sensitivity	integer	Range is 0,1,2 0—high, 1—default, 2—low
<b>head</b> .FocusLimitSelectMode	string	Manual or Auto.

## 5.7 Lighting

### 5.7.1 Lighting Config

- Get lighting

Table 5-21

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Lighting
Method	GET
Description	Get lighting
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=Lighting
Success	head.Correction=50
Return	head.FarLight[0].Angle=0

	<pre>head.FarLight[0].Light=0 head.Mode=ZoomPrio head.NearLight[0].Angle=0 head.NearLight[0].Light=0 head.MiddleLight[0].Angle=50 head.MiddleLight[0].Light=50</pre>
Comment	<p><b>head</b> = table.Lighting[<b>ChannelNo</b>][<b>ConfigNo</b>]:</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>ConfigNo</b>: array index, can be 0, 1 or 2. 0 means config for day, 1 means config for night, and 2 means config for normal scene.</p>

- Set lighting

Table 5-22

Syntax	<pre>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;paramName&gt;=&lt;paramValue&gt;[&amp;&lt;paramName&gt;=&lt;paramValue&gt;...]</pre>
Method	GET
Description	Set lighting
Example	<p>Turn on light:  <pre>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;Lighting[0][0].FarLight[0].Light=10&amp;Lighting[0][0].NearLight[0].Light=90&amp;Lighting[0][0].Mode=Manual</pre></p> <p>Shift the light to ZoomPrio mode:  <pre>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;Lighting[0][0].Correction=50&amp;Lighting[0][0].Mode=ZoomPrio</pre></p>
Success Return	OK
Comment	<p>Parameters in URL:  paramName and paramValue are as table below.  In table below,  <b>head</b> = Lighting[<b>ChannelNo</b>][<b>ConfigNo</b>]</p> <p>Some lighting equipment has the following restriction:  Lighting[0][0].FarLight[0].Light + Lighting[0][0].NearLight[0].Light+ Lighting[0][0].MiddleLight [0]. Light &lt;=100.</p>

**Appendix:**

ParamName	ParamValue type	Description
<b>head.Mode</b>	string	<p>Light mode.  The range is {"Manual", "Auto", "Off", "ZoomPrio"}.</p> <p>The following are special for composite lamps.  The range is { "Timing" , "SmartLight" , "ExclusiveManual" }</p>

ParamName	ParamValue type	Description
<b>head</b> .Correction	integer	Light compensation. The range is [0—100], effective in ZoomPrio mode.
<b>head</b> .FarLight[Index].Light	integer	Range is [0—100]. The luminance of far light.
<b>head</b> .MiddleLight[Index].Light	integer	Range is [0—100]. The luminance of middle light.
<b>head</b> .NearLight[Index].Light	integer	Range is [0—100]. The luminance of near light.

## 5.8 Video in Options

### 5.8.1 Video in Options Config

It's not recommended to use the CGI command from " video in options "; It's now recommended to use the commands in 5.1 – 5.7.

- Get video in options

Table 5-23

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoInOptions
Method	GET
Description	Get video in options config, such as Backlight, ExposureSpeed, DayNightColor, DayOptions, NightOptions, and NormalOptions.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoInOptions
Success Return	<pre> <b>head</b>.Backlight=0 <b>head</b>.DayNightColor=false <b>head</b>.ExposureSpeed=0 <b>head</b>.ExposureValue1=0.100000 <b>head</b>.ExposureValue2=80.000000 <b>head</b>.ExternalSync=0 <b>head</b>.ExternalSyncPhase=0 <b>head</b>.FlashControl.Mode=0 <b>head</b>.FlashControl.Pole=0 <b>head</b>.FlashControl.Value=0 <b>head</b>.FlashControl.PreValue=0 <b>head</b>.Flip=false <b>head</b>.Gain=50 <b>head</b>.GainAuto=true <b>head</b>.IrisAuto=false <b>head</b>.Mirror=false <b>head</b>.NightOptions.AntiFlicker=0 <b>head</b>.NightOptions.Backlight=0 <b>head</b>.NightOptions.BacklightRegion[0]=3096 <b>head</b>.NightOptions.BacklightRegion[1]=3096 </pre>

**head.NightOptions.BacklightRegion[2]=5096**  
**head.NightOptions.BacklightRegion[3]=5096**  
**head.NightOptions.BrightnessThreshold=50**  
**head.NightOptions.DayNightColor=2**  
**head.NightOptions.ExposureMode=0**  
**head.NightOptions.ExposureSpeed=0**  
**head.NightOptions.ExposureValue1=0**  
**head.NightOptions.ExposureValue2=40**  
**head.NightOptions.ExternalSyncPhase=125**  
**head.NightOptions.Flip=false**  
**head.NightOptions.Gain=50**  
**head.NightOptions.GainAuto=true**  
**head.NightOptions.GainBlue=50**  
**head.NightOptions.GainGreen=50**  
**head.NightOptions.GainMax=50**  
**head.NightOptions.GainMin=0**  
**head.NightOptions.GainRed=50**  
**head.NightOptions.GlareInhibition=0**  
**head.NightOptions.IrisAuto=true**  
**head.NightOptions.Mirror=false**  
**head.NightOptions.Profile=3**  
**head.NightOptions.ReferenceLevel=50**  
**head.NightOptions.Rotate90=0**  
**head.NightOptions.SunriseHour=0**  
**head.NightOptions.SunriseMinute=0**  
**head.NightOptions.SunriseSecond=0**  
**head.NightOptions.SunsetHour=23**  
**head.NightOptions.SunsetMinute=59**  
**head.NightOptions.SunsetSecond=59**  
**head.NightOptions.SwitchMode=4**  
**head.NightOptions.WhiteBalance=Auto**  
**head.NightOptions.WideDynamicRange=0**  
**head.NightOptions.WideDynamicRangeMode=0**  
**head.NormalOptions.AntiFlicker=0**  
**head.NormalOptions.Backlight=0**  
**head.NormalOptions.BacklightRegion[0]=3096**  
**head.NormalOptions.BacklightRegion[1]=3096**  
**head.NormalOptions.BacklightRegion[2]=5096**  
**head.NormalOptions.BacklightRegion[3]=5096**  
**head.NormalOptions.BrightnessThreshold=50**  
**head.NormalOptions.DayNightColor=1**  
**head.NormalOptions.ExposureMode=0**  
**head.NormalOptions.ExposureSpeed=0**  
**head.NormalOptions.ExposureValue1=0**  
**head.NormalOptions.ExposureValue2=40**  
**head.NormalOptions.ExternalSyncPhase=125**  
**head.NormalOptions.Flip=false**

	<pre> <b>head</b>.NormalOptions.Gain=50 <b>head</b>.NormalOptions.GainAuto=true <b>head</b>.NormalOptions.GainBlue=50 <b>head</b>.NormalOptions.GainGreen=50 <b>head</b>.NormalOptions.GainMax=50 <b>head</b>.NormalOptions.GainMin=0 <b>head</b>.NormalOptions.GainRed=50 <b>head</b>.NormalOptions.GlareInhibition=0 <b>head</b>.NormalOptions.IrisAuto=true <b>head</b>.NormalOptions.Mirror=false <b>head</b>.NormalOptions.Profile=0 <b>head</b>.NormalOptions.ReferenceLevel=50 <b>head</b>.NormalOptions.Rotate90=0 <b>head</b>.NormalOptions.SunriseHour=0 <b>head</b>.NormalOptions.SunriseMinute=0 <b>head</b>.NormalOptions.SunriseSecond=0 <b>head</b>.NormalOptions.SunsetHour=23 <b>head</b>.NormalOptions.SunsetMinute=59 <b>head</b>.NormalOptions.SunsetSecond=59 <b>head</b>.NormalOptions.SwitchMode=0 <b>head</b>.ReferenceLevel=50 <b>head</b>.ReferenceLevelEnable=false <b>head</b>.Rotate90=0 <b>head</b>.SignalFormat=BT656 <b>head</b>.WhiteBalance=Disable </pre>
Comment	<p>Parameters in Response:</p> <p><b>head</b> = table.VideoInOptions[<b>ChannelNo</b>]  <b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p>

- Set video in options

Table 5-24

Syntax	<pre> http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;[&amp;&lt;<b>paramName</b>&gt;=&lt;<b>paramValue</b>&gt;...] </pre>
Method	GET
Description	Set video in options config, such as Backlight, ExposureSpeed, DayNightColor, DayOptions, NightOptions, and NormalOptions.
Example	<p>Set Auto Exposure:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=0&amp;VideoInOptions[0].ExposureSpeed=0">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=0&amp;VideoInOptions[0].ExposureSpeed=0</a></p> <p>Set Low Noise:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=1&amp;VideoInOptions[0].ExposureSpeed=0&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=60">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=1&amp;VideoInOptions[0].ExposureSpeed=0&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=60</a></p>

	<p>Set Low Motion Blur:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=2&amp;VideoInOptions[0].ExposureSpeed=0&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=50&amp;VideoInOptions[0].ExposureValue1=0&amp;VideoInOptions[0].ExposureValue2=20">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=2&amp;VideoInOptions[0].ExposureSpeed=0&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=50&amp;VideoInOptions[0].ExposureValue1=0&amp;VideoInOptions[0].ExposureValue2=20</a></p> <p>Set Manual:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=4&amp;VideoInOptions[0].ExposureSpeed=32&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=50&amp;VideoInOptions[0].ExposureValue1=40&amp;VideoInOptions[0].ExposureValue2=40">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].ExposureMode=4&amp;VideoInOptions[0].ExposureSpeed=32&amp;VideoInOptions[0].GainMin=0&amp;VideoInOptions[0].GainMax=50&amp;VideoInOptions[0].ExposureValue1=40&amp;VideoInOptions[0].ExposureValue2=40</a></p> <p>Set SmartIRExposure:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].SmartIRExposure=true">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].SmartIRExposure=true</a></p> <p>Set Video Rotate:          Flip:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Flip=true">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Flip=true</a>          Mirror:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Mirror=true">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Mirror=true</a>          Or turn 90°:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Rotate90=1">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].Rotate90=1</a></p> <p>Set White Balance:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Night">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Night</a>          Or  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Custom&amp;VideoInOptions[0].GainRed=50&amp;VideoInOptions[0].GainBlue=50&amp;VideoInOptions[0].GainGreen=50">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Custom&amp;VideoInOptions[0].GainRed=50&amp;VideoInOptions[0].GainBlue=50&amp;VideoInOptions[0].GainGreen=50</a>          (Sometimes you should set mode first before set GainRed or GainBlue:  <a href="http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Custom">http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;VideoInOptions[0].WhiteBalance=Custom</a> )</p>
Success Return	OK
Comment	Parameters in URL: In table below, <b>head</b> = VideoInOptions[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

**Appendix:**

ParamName	ParamValue type	Description
-----------	-----------------	-------------

ParamName	ParamValue type	Description
<b>head</b> .Backlight	integer	Range is [0—n] n depends on capability in <b>GetVideoInputCaps</b> 0: backlight closed. 1: backlight grade 1 ... n - backlight grade n
<b>head</b> .DayNightColor	integer	Range is {0,1,2} 0: always multicolor 1: autoswitch along with brightness, 2: always monochrome
<b>head</b> .ExposureMode	integer	Range is {0,1,2, 4} 0: AutoExposure 1: Gain first 2: Exposure first 4: Manual.
<b>head</b> .ExposureSpeed	integer	Range is [0 — n+1] n depends on capability in <b>GetVideoInputCaps</b> 0: AutoExposure 1-n-1: manual Exposure grade n: AutoExposure with time limit. n+1: manualExposure with user-defined time (n is supported maximum exposure grade )
<b>head</b> .ExposureValue1	float	Range is [0.1—80], unit is millisecond If ExposureSpeed is 0(AutoExposure enable), it's lower limit of AutoExposure time, otherwise it's time of manualExposure
<b>head</b> .ExposureValue2	float	Range is [0.1-80], unit is millisecond Upper limit of AutoExposure time, should be bigger than ExposureValue1
<b>head</b> .ExternalSync	integer	Range is {0,1} External Synchronous 0: Internal Synchronization 1: External Synchronous
<b>head</b> .ExternalSyncPhase	foat	Range is [0°—360°] External Synchronous Signal Phase
<b>head</b> .SmartIRExposure	bool	true: enable, false: disable
<b>head</b> .FlashControl.Mode	integer	Range is {0,1,2} 0: forbid flash 1: always flash 2: auto flash
<b>head</b> .FlashControl.Pole	integer	Range is {0,1, 2, 3} Trigger mode: 0: low level 1: high level 2: rising-edge 3: falling-edge

ParamName	ParamValue type	Description
<b>head.FlashControl.Value</b>	integer	Range is [0—15] Flashlight time-unit: 0: 0us, 1: 64us, 2: 128us, 3: 192us ... 15 - 960us
<b>head.FlashControl.PreValue</b>	integer	Range is [0—100] It is threshold of brightness value: if brightness is less than this value, flash light will begin to work.
<b>head.Flip</b>	bool	true: enable video flip function false: disable video flip function
<b>head.Gain</b>	integer	Range is [0—100] If GainAuto is true, it's upper limit of auto gain, else it's the fixed gain adjust value.
<b>head.GainBlue</b>	integer	Range is [0—100] Gain for blue value, Value is effective when WhiteBalance is "Custom."
<b>head.GainRed</b>	integer	Range is [0—100] Gain for red value, Value is effective when WhiteBalance is "Custom."
<b>head.GainGreen</b>	integer	Range is [0—100] Gain for green value, Value is effective when WhiteBalance is "Custom."
<b>head.GainAuto</b>	bool	true: GainAuto false: No GainAuto
<b>head.IrisAuto</b>	bool	true: IrisAuto false: No IrisAuto
<b>head.Mirror</b>	bool	true: enable video mirror function false: disable video mirror function
<b>head.WhiteBalance</b>	String	White balance Mode. Range is {Disable, Auto, Custom, Sunny, Cloudy, Home, Office, Night} Some IPC supports common modes: "Disable", "Auto", "Sunny", "Night", "Outdoor", "Custom" Sometimes the device support other advanced modes: "CustomColorTemperature", "Indoor", "ATW", "Manual", "AutoOutdoor", "ManualDatum" and so on.
<b>head.ReferenceLevel</b>	integer	Range is [0—100] The expected average brightness level of video frames.

ParamName	ParamValue type	Description
<b>head.Rotate90</b>	integer	Range is {0,1,2} Video rotation: 0: No rotate 1: clockwise rotate 90° 2: anticlockwise rotate 90°
<b>head.SignalFormat</b>	String	Range is {Inside, BT656, 720p, 1080p, 1080i, 1080sF} Input Signal Mode
<b>head.AntiFlicker</b>	integer	Range is {0,1,2} AntiFlicker mode: 0: Outdoor 1: 50 Hz AntiFlicker 2: 60 Hz AntiFlicker
<b>head.GlareInhibition</b>	integer	Range is [0—100] GlareInhibition: 0: Close GlareInhibition.
<b>head.NightOptions.BrightnessThreshold</b>	integer	NightOptions contain a set of parameters used when brightness is not enough. Range is [0—100] when brightness is less than the BrightnessThreshold, parameters change to Nightoptions.
<b>head.NightOptions.IrisAuto</b>	bool	true: IrisAuto false: No IrisAuto
<b>head.NightOptions.SunriseHour</b>	integer	Range is [00—23] Sunrise hour.
<b>head.NightOptions.SunriseMinute</b>	integer	Range is [00—59] Sunrise minute
<b>head.NightOptions.SunriseSecond</b>	integer	Range is [00—59] Sunrise second
<b>head.NightOptions.SunsetHour</b>	integer	Sunset time. Its range is same with sunrise time, and it should be after sunrise time. NightOptions are used if time is after sunset time and before sunrise time.
<b>head.NightOptions.SunsetMinute</b>	integer	
<b>head.NightOptions.SunsetSecond</b>	integer	
<b>head.NightOptions.SwitchMode</b>	integer	Range is {0,1,2} 0: NoSwitch, always use day options. 1: Switch depends on brightness. 2: Switch depends on time, switch to NightOptions when time is after sunset time and before sunrise. 3: NoSwitch, always use NightOptions. 4: No switch, always use NormalOptions.

ParamName	ParamValue type	Description
<b>head.NightOptions.Profile</b>	integer	Range is {0,1,2,3} 0: use temporary day options. 1: use temporary NightOptions. 2: use temporary NormalOptions. 3: depends on <i>head.NightOptions.SwitchMode</i>
<b>head.NightOptions.ExposureSpeed</b>	integer	<p>Range is the same as relevant items of day options in this table. Example: Value range of <i>head.NightOptions.ExposureSpeed</i> is the same with <i>head.ExposureSpeed</i>.</p>
<b>head.NightOptions.ExposureValue1</b>	float	
<b>head.NightOptions.ExposureValue2</b>	float	
<b>head.NightOptions.Gain</b>	integer	
<b>head.NightOptions.GainAuto</b>	bool	
<b>head.NightOptions.GainBlue</b>	integer	
<b>head.NightOptions.GainGreen</b>	integer	
<b>head.NightOptions.GainRed</b>	integer	
<b>head.NightOptions.WhiteBalance</b>	String	
<b>head.NightOptions.ReferenceLevel</b>	integer	
<b>head.NightOptions.ExternalSyncPhase</b>	float	
<b>head.NightOptions.AntiFlicker</b>	integer	
<b>head.NightOptions.Backlight</b>	integer	
<b>head.NightOptions.DayNightColor</b>	integer	
<b>head.NightOptions.ExposureMode</b>	integer	
<b>head.NightOptions.GlareInhibition</b>	integer	
<b>head.NightOptions.Mirror</b>	integer	
<b>head.NightOptions.Flip</b>	integer	
<b>head.NightOptions.Rotate90</b>	integer	
<b>head.NormalOptions.BrightnessThreshold</b>	integer	<p>NomalOptions contain a set of parameters similar with NightOptions. Range is the same as relevant items of NightOptions in this table.</p>
<b>head.NormalOptions.IrisAuto</b>	bool	
<b>head.NormalOptions.SunriseHour</b>	integer	
<b>head.NormalOptions.SunriseMinute</b>	integer	
<b>head.NormalOptions.SunriseSecond</b>	integer	

ParamName	ParamValue type	Description
<b>head</b> .NormalOptions.SunsetHour	integer	
<b>head</b> .NormalOptions.SunsetMinute	integer	
<b>head</b> .NormalOptions.SunsetSecond	integer	
<b>head</b> .NormalOptions.ExposureSpeed	integer	
<b>head</b> .NormalOptions.ExposureValue1	float	
<b>head</b> .NormalOptions.ExposureValue2	float	
<b>head</b> .NormalOptions.Gain	integer	
<b>head</b> .NormalOptions.GainAuto	bool	
<b>head</b> .NormalOptions.GainBlue	integer	
<b>head</b> .NormalOptions.GainGreen	integer	
<b>head</b> .NormalOptions.GainRed	integer	
<b>head</b> .NormalOptions.WhiteBalance	String	
<b>head</b> .NormalOptions.ReferenceLevel	integer	
<b>head</b> .NormalOptions.ExternalSyncPhase	float	
<b>head</b> .NormalOptions.AntiFlicker	integer	
<b>head</b> .NormalOptions.Backlight	integer	
<b>head</b> .NormalOptions.DayNightColor	integer	
<b>head</b> .NormalOptions.ExposureMode	integer	
<b>head</b> .NormalOptions.GlareInhibition	integer	
<b>head</b> .NormalOptions.Mirror	integer	
<b>head</b> .NormalOptions.Flip	integer	
<b>head</b> .NormalOptions.Rotate90	integer	

## 6.1 Storage Devices

### 6.1.1 Get Hard Disk Information

Table 6-1

Syntax	http://<server>/cgi-bin/storageDevice.cgi?action=factory.getPortInfo
Method	GET
Description	Get the storage device port info.
Example	http://192.168.1.108/cgi-bin/storageDevice.cgi?action=factory.getPortInfo
Success Return	info.Total=2 info.Plug=1 info.Mask=1 info.Bad=0 info.IDE=1 info.Esata=4
Comment	—

### 6.1.2 Get All the Names of Storage Devices

Table 6-2

Syntax	http://<server>/cgi-bin/storageDevice.cgi?action=factory.getCollect
Method	GET
Description	Get all the names of storage devices.
Example	http://192.168.1.108/cgi-bin/storageDevice.cgi?action=factory.getCollect
Success Return	list[0]="/dev/sda0" list[1]="/dev/sda1" list[2]="/dev/sg1"
Comment	—

### 6.1.3 Get Storage Device Information

Table 6-3

Syntax	http://<server>/cgi-bin/storageDevice.cgi?action=getDeviceAllInfo
Method	GET
Description	Get all the storage device information.
Example	http://192.168.1.108/cgi-bin/storageDevice.cgi?action=getDeviceAllInfo
Success Return	list.info[0].Detail[0].IsError=false list.info[0].Detail[0].Pointer=27023434

	list.info[0].Detail[0].TotalBytes=0 list.info[0].Detail[0].Type=ReadWrite list.info[0].Detail[0].Path="/mnt/dvr/sda0" list.info[0].Detail[0].UsedBytes=0 list.info[0].Name="/dev/sda" list.info[0].State=Success list.info[0].TotalDays=64 list.info[0].FreeDays=32
Comment	—

## 6.1.4 Get Storage Capability

Table 6-4

Syntax	http://<server>/cgi-bin/storage.cgi?action=getCaps
Method	GET
Description	Get storage capabilities.
Example	http://192.168.1.108/cgi-bin/storage.cgi?action=getCaps
Success	caps.RedundantDisk.Support=false
Return	caps.SupportRemoteLimit=true
Comment	—

## 6.1.5 Format Camera SD-Card

Table 6-5

Syntax	http://<server>/cgi-bin/storageDevice.cgi?action=setDevice&type=FormatPatition&path=<path> [&<paramName>=<paramValue>...]		
Method	GET		
Description	Format camera SD card.		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
type	string	R	operation
path	string	O	disk part path
formattype	Array<object>	O	disk part information
+fs	string	R	File system type
Example	http://192.168.1.108/cgi-bin/storageDevice.cgi?action=setDevice&type=FormatPatition&path= /dev/sda&formattype[0].fs=fat32		
Success Return	OK		
Comment	If set successfully, return true, else return false. <b>path</b> : The <b>stringValue</b> is got from cgi API "Get storage device information" (/cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). <b>stringValue</b> is list[0].Name.		

## 6.2 NAS

### 6.2.1 NAS Information

- Get NAS config

Table 6-6

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=NAS
Method	GET
Description	Get all the directories on the NAS server.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=NAS
Success Return	table.NAS[0].Name="FTP1" table.NAS[0].Enable=true table.NAS[0].Protocol="FTP" table.NAS[0].Address="www.ttt.com" table.NAS[0].Port=21 table.NAS[0].UserName="anonymity" table.NAS[0].Password="none" table.NAS[0].Directory="share"
Comment	—

- Set NAS config

Table 6-7

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> < <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> <...>]
Method	GET
Description	Set NAS config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&NAS[0].Name=nas01&NAS[0].Enable=true
Success Return	OK
Comment	Parameters in URL: In table below, <b>Head</b> =NAS[ <b>index</b> ] <b>index</b> : The index of the NAS Server

#### Appendix:

ParamName	ParamValue type	Description
<b>Head</b> .Name	string	NAS name.
<b>Head</b> .Enable	bool	Enable/Disable the NAS.
<b>Head</b> .Protocol	string	The range is {"FTP", "SFTP", "SMB", "NFS", "Cloud", "ISCSI" }
<b>Head</b> .Address	string	The IP address or host name.
<b>Head</b> .Port	integer	NAS port.

ParamName	ParamValue type	Description
<b>Head</b> .UserName	string	NAS username.
<b>Head</b> .Password	string	NAS password.
<b>Head</b> .Directory	string	Directory name.

## 6.3 Storage Point

### 6.3.1 Record Storage Point

- Get record storage point config

Table 6-8

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=RecordStoragePoint
Method	GET
Description	Get record storage point config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RecordStoragePoint
Success Return	table.RecordStoragePoint [0].TimingRecord.Local=1 table.RecordStoragePoint [0].TimingRecord.Redundant=Redundant table.RecordStoragePoint [0].TimingRecord.Remote=FTP table.RecordStoragePoint [0].TimingRecord.AutoSync=false table.RecordStoragePoint [0].TimingRecord.AutoSyncRange=0 table.RecordStoragePoint [0].TimingRecord.LocalForEmergency=false table.RecordStoragePoint [0].TimingRecord.CompressBefore=15
Comment	—

- Set record storage point config

Table 6-9

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set record storage point config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&RecordStoragePoint[0].TimingRecord.Local=local
Success Return	OK
Comment	Parameters in URL: In table below, <b>ch</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1). <b>recType</b> : The range is {"TimingRecord", "VideoDetectRecord", "AlarmRecord", "EventRecord", "TimingSnapShot", "VideoDetectSnapShot", "AlarmSnapShot", "EventSnapShot"}

#### Appendix:

ParamName	ParamValue type	Description
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Local	integer/boolean	Local directory number, count from 1. For some reason, The front-end devices use Boolean type. True for SD card, false for no SD card.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Redundant	string	Redundant directory name.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Remote	string	Remote directory name.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Auto Sync	bool	When remote directory recovers, auto synchronize local directory to remote directory or not.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Auto SyncRange	integer	From the remote directory recovering time, how long the data needs to be synchronized. The unit is hour. If it is 0, all the data needs to be synchronized.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].Local ForEmergency	bool	When the remote directory is unusable, save the data the local directory or not.
RecordStoragePoint[ <i>ch</i> ].[ <i>recType</i> ].CompressBefore	integer	The days' data which will be compressed.

### 6.3.2 Storage Group

- Get storage group config

Table 6-10

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageGroup
Method	GET
Description	Get storage group config.
Example	http://192.168.1.168/cgi-bin/configManager.cgi?action=getConfig&name=StorageGroup
Success Return	<pre>table.StorageGroup[0].Channels[0].MaxPictures=0 table.StorageGroup[0].FileHoldTime=0 table.StorageGroup[0].Memo=For Reading &amp; Writing Files table.StorageGroup[0].Name=ReadWrite table.StorageGroup[0].OverWrite=true table.StorageGroup[0].PicturePathRule=%y-%M-%d/%c/jpg/%h/%m/%s[%E][%O@%S][%R].jpg table.StorageGroup[0].RecordPathRule=%y-%M-%d/%c/dav/%h/%h.%m.%s-%h.%m.%s[%E][%O@%S][%R].dav table.StorageGroup[1].Channels[0].MaxPictures=0 table.StorageGroup[1].FileHoldTime=0 table.StorageGroup[1].Memo=For FTP Files</pre>

	<pre>table.StorageGroup[1].Name=Remote table.StorageGroup[1].OverWrite=true table.StorageGroup[1].PicturePathRule=%y-%M-%d/%c/jpg/%h/%m/%s[%E][%O@%S][%R].jpg table.StorageGroup[1].RecordPathRule=%y-%M-%d/%c/dav/%h/%h.%m.%s-%h.%m.%s[%E][%O@%S][%R].dav</pre>
Comment	—

- Set storage group config

Table 6-11

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> < <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> <...]
Method	GET
Description	Set storage group config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&StorageGroup[0].Name=main
Success Return	OK
Comment	<p>Parameters in URL:</p> <p>In table below,</p> <p><b>Index</b> = Storage Group index</p> <p><b>ch</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p>

#### Appendix:

ParamName	ParamValue type	Description
StorageGroup[ <b>Index</b> ].Name	string	Storage group name.
StorageGroup[ <b>Index</b> ].Memo	string	Storage group memo.
StorageGroup[ <b>Index</b> ].FileHoldTime	integer	How many days the file will hold.
StorageGroup[ <b>Index</b> ].OverWrite	bool	Over write or not when there is not enough storage.
StorageGroup[ <b>Index</b> ].Channels[ <b>ch</b> ].MaxPictures	Integer	The max pictures beyond which the old pictures will be over written. If it is 0, the old pictures will be not over written.
StorageGroup[ <b>Index</b> ].Channels[ <b>ch</b> ].Path	string	The channel path.

## 6.4 SDEncrypt

### 6.4.1 Encrypt SD Card

Table 6-12

Syntax	http://<server>/cgi-bin/SDEncrypt.cgi?action=encrypt&deviceName=< <b>&lt;deviceName&gt;</b> &password=< <b>&lt;password&gt;</b>
--------	---

Method	GET
Description	SD encryption operation.
Example	http://192.168.1.108/cgi-bin/SEncrypt.cgi?action=encrypt&deviceName=/dev/mmc0&password=123456
Success Return	OK
Comment	If set successfully, return true, else return false. deviceName: The stringValue is got from cgi API Get storage device information (cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). password: The stringValue Card is Encrypted

## 6.4.2 Decrypt SD Card

Table 6-13

Syntax	http://<server>/cgi-bin/SEncrypt.cgi?action=decrypt&deviceName=<deviceName>&password=<password>
Method	GET
Description	SD decrypt operation.
Example	http://192.168.1.108/cgi-bin/SEncrypt.cgi?action=decrypt&deviceName=/dev/mmc0&password=123456
Success Return	OK
Comment	If set successfully, return true, else return false. <b>deviceName:</b> The stringValue is got from cgi API Get storage device information (cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). <b>password:</b> The stringValue ard is Encrypted

## 6.4.3 Clear SD Card Password

Table 6-14

Syntax	http://<server>/cgi-bin/SEncrypt.cgi?action=clearPassword&deviceName=<deviceName>&password=<password>
Method	GET
Description	SD clear password operation.
Example	http://192.168.1.108/cgi-bin/SEncrypt.cgi?action=clearPassword&deviceName=/dev/mmc0&password=123456
Success Return	OK
Comment	If set successfully, return true, else return false. <b>deviceName:</b> The stringValue is got from cgi API Get storage device information (cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). password: The stringValue ard is Encrypted

## 6.4.4 Modify SD Card Password

Table 6-15

Syntax	http://<server>/cgi-bin/SDEncrypt.cgi?action=modifyPassword&deviceName=<deviceName>&password=<password>&oldPassword=<oldPassword>
Method	GET
Description	SD change password operation.
Example	http://192.168.1.108/cgi-bin/SDEncrypt.cgi?action=clearPassword&deviceName=/dev/mmc0&password=123456&oldPassword=admin123
Success Return	OK
Comment	If set successfully, return true, else return false. deviceName: The stringValue is got from cgi API Get storage device information (cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). password: The new stringValue Card is Encrypted oldPassword: The old stringValue Card is Encrypted

## 6.4.5 Get SD Card Operate Error Policy

Table 6-16

Syntax	http://<server>/cgi-bin/SDEncrypt.cgi?action=getOperateErrorPolicy&deviceName=<deviceName>&operate=<operate>
Method	GET
Description	When decrypt, clearPassword, modifyPassword failed, get the error info, leftTimes and left time.
Example	http://192.168.1.108/cgi-bin/SDEncrypt.cgi?action=getOperateErrorPolicy&deviceName=/dev/mmc0&operate=decrypt
Success Return	policy.leftTimes=5 policy.lockSeconds=30
Comment	If set successfully, return true, else return false. deviceName: the stringValue is got from cgi API Get storage device information (cgi-bin/storageDevice.cgi?action=getDeviceAllInfo). Operate: operate type, can be: decrypt, modifyPassword and clearPassword leftTimes : remain operate times, max is 5 lockSeconds : lock operate time, unit is seconds, max is 30

## 6.4.6 Storage Health Alarm Settings

- Get StorageHealthAlarm config

Table 6-17

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageHealthAlarm
Method	GET

Description	Get SD health info alarm config.
Example	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=StorageHealthAlarm
Success Return	table.StorageHealthAlarm.Enable=true table.StorageHealthAlarm.LowerLimit=10 table.StorageHealthAlarm.EventHandler= (output of EventHandler is described in GetEventHandler)
Comment	—

- Set StorageHealthAlarm config

Table 6-18

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> >[&< <b>&lt;paramName&gt;</b> < <b>&lt;paramValue&gt;</b> >...]
Method	GET
Description	Set SD health info alarm config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&StorageHealthAlarm.Enable=true
Success Return	OK
Comment	—

## 7.1 GUI

### 7.1.1 GUISet

- Get GUISet config

Table 7-1

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=GUISet
Method	GET
Description	Get the GUI settings. Every video out screen has a group setting.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=GUISet
Success Return	<pre>table.GUISet[<i>index</i>].WindowAlpha=128 table.GUISet[<i>index</i>].TimeTitleEnable=true table.GUISet[<i>index</i>].TimeTitlePos[0]=0 table.GUISet[<i>index</i>].TimeTitlePos[1]=0 table.GUISet[<i>index</i>].TimeTitlePos[2]=8191 table.GUISet[<i>index</i>].TimeTitlePos[3]=8191 table.GUISet[<i>index</i>].MenuShowOption=0 table.GUISet[<i>index</i>].MenuAutoHideTime=10 table.GUISet[<i>index</i>].AutoLogout=10 table.GUISet[<i>index</i>].ChannelTitleShowEnable=true table.GUISet[<i>index</i>].ChannelTitlePos[0]=0 table.GUISet[<i>index</i>].ChannelTitlePos[1]=0 table.GUISet[<i>index</i>].ChannelTitlePos[2]=8191 table.GUISet[<i>index</i>].ChannelTitlePos[3]=8191 table.GUISet[<i>index</i>].AutoGuideEnable=true ...</pre>
Comment	Parameters in Response : <b>index</b> : the array index which starts from 0.

- Set GUISet config

Table 7-2

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set the GUI settings. Every video out screen has a group setting.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&GUISet[0].WindowAlpha=192&GUISet[0].TimeTitleEnable=false&GUISet[0].MenuShowOption=1
Success Return	OK

Comment	Parameters in URL: The paramName and paramValue are in the table below. in table below, <b>index</b> : the array index which starts from 0.
---------	--

## Appendix:

ParamName	ParamValue type	Description
GUISet[ <i>index</i> ].WindowAlpha	integer	Diaphaneity of the window background.
GUISet[ <i>index</i> ].TimeTitleEnable	bool	Show the time title or not.
GUISet[ <i>index</i> ].TimeTitlePos[0]	integer	The position of the time title.
GUISet[ <i>index</i> ].TimeTitlePos[1]	integer	
GUISet[ <i>index</i> ].TimeTitlePos[2]	integer	
GUISet[ <i>index</i> ].TimeTitlePos[3]	integer	
GUISet[ <i>index</i> ].MenuShowOption	integer	0: Show the directory. 1: Hide the directory. 2: Timing-hide the directory.
GUISet[ <i>index</i> ].MenuAutoHideTime	integer	How many seconds to hide the directory.
GUISet[ <i>index</i> ].AutoLogout	integer	How many minutes to auto logout. The range is [0-120]. 0 expresses not logout.
GUISet[ <i>index</i> ].ChannelTitleShowEnable	bool	Show the channel title or not.
GUISet[ <i>index</i> ].ChannelTitlePos[0]	integer	The position of the channel title.
GUISet[ <i>index</i> ].ChannelTitlePos[1]	integer	
GUISet[ <i>index</i> ].ChannelTitlePos[2]	integer	
GUISet[ <i>index</i> ].ChannelTitlePos[3]	integer	
GUISet[ <i>index</i> ].AutoGuideEnable	bool	Auto guide or not when startup.

## 7.2 Split Screen

### 7.2.1 Split Screen Mode

- Get split screen mode

Table 7-3

Syntax	http://<server>/cgi-bin/split.cgi?action=getMode&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get the split screen mode.
Example	http://192.168.1.108/cgi-bin/split.cgi?action=getMode&channel=1
Success	mode=split1
Return	group=4
Comment	Parameters in URL: <b>ChannelNo</b> : the display screen No. Start from 1 and <= 2.

- Set split screen mode

Table 7-4

Syntax	http://<server>/cgi-bin/split.cgi?action=setMode&channel=< <b>ChannelNo</b> >&mode=< <b>mode</b> >&group=< <b>group</b> >
Method	GET
Description	Set the split screen mode.
Example	http://192.168.1.108/cgi-bin/split.cgi?action=setMode&channel=1&mode=split4&group=1
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: the display screen No. Start from 1.</p> <p><b>mode</b>:enum{split1,split2,split4,split6,split8,split9,split12,split16,split20,split25,split36,split64,split144,pip1,pip3, "Free", "CompositeSplit1" / "FitDisplayUnit1", "CompositeSplit1" / "FitDisplayUnit4"};</p> <p><b>group</b>: the No. of a group which contains certain number channels. For example, if 16 video channels display in split4 Mode which contains 4 video channels on Screen, then there are 4 groups and each group contains 4 video channels.</p>

## 7.3 Monitor Tour

### 7.3.1 Monitor Tour

- Get monitor tour config

Table 7-5

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=MonitorTour
Method	GET
Description	Get monitor tour config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=MonitorTour
Success Return	table.MonitorTour[ch].Enable=128 table.MonitorTour[ch].Interval=true table.MonitorTour[ch].Mask.Split1=0,1,5 table.MonitorTour[ch].Mask.Split8=0,1,5 table.MonitorTour[ch].Collections=Favortite1, Favortite2...
Comment	—

- Set monitor tour config

Table 7-6

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set monitor tour config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&MonitorTour[0].Enable=true

Success Return	OK.
Comment	Parameters in URL: The paramName and paramValue are in the table below.

#### Appendix:

ParamName	ParamValue type	Description
MonitorTour[ch].Enable	bool	MonitorTour or not.
MonitorTour[ch].Interval	integer	MonitorTour interval.
MonitorTour[ch].Mask.Split1	—	Channel array for split1
MonitorTour[ch].Mask.Split8	—	Channel array for split8
MonitorTour[ch].Collections	—	Split collections

### 7.3.2 Enable Tour

Table 7-7

Syntax	http://<server>/cgi-bin/split.cgi?action=enableTour&channel=< <b>ChannelNo</b> >&enable=< <b>flag</b> >
Method	GET
Description	Enable tour in every video channel on a screen or not.
Example	http://192.168.1.108/cgi-bin/split.cgi?action=enableTour&channel=1&enable=true
Success Return	OK
Comment	<b>ChannelNo</b> : the display screen No. Start from 1 and <= 2. <b>flag</b> : true or false

### 7.3.3 Monitor Collection

- Get monitor collection config

Table 7-8

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=MonitorCollection
Method	GET
Description	Get monitor collection config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=MonitorCollection
Success Return	table.MonitorCollection.collectionname. Mode=Split1 table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].Enable= true table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].Device=device1 table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].VideoChannel=5 table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].VideoStream=Main table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].AudioChannel=5 table.MonitorCollection.collectionname.Windows[ <b>winno</b> ].AudioStream=Main ...
Comment	Parameters in Response :

	<b>winno</b> : integer, the array index which equals to the window index in a screen and starts from 0.
--	---

- Set monitor collection config

Table 7-9

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set monitor collection config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&MonitorCollection.Favorite1.Mode=split4&MonitorCollection.Favorite1.Windows[1].Enable=true&MonitorCollection.Favorite1.Windows[1].VideoChannel=2
Success Return	OK
Comment	<p>Parameters in URL:  The paramName and paramValue are in the table below.  In table below:  <b>Collect</b>=MonitorCollection.<b>collectionname</b>.  <b>collectionname</b>: can be any name.  <b>winno</b>: integer, the array index which equals to the window index in a screen and starts from 0.</p>

**Appendix:**

ParamName	ParamValue type	Description
<b>Collect</b> .Mode	string	The range is the same as <b>SetSplitMode</b> .
<b>Collect</b> .Windows[ <b>winno</b> ].Enable	bool	Enable the window or not.
<b>Collect</b> .Windows[ <b>winno</b> ].Device	string	The device Id.
<b>Collect</b> .Windows[ <b>winno</b> ].VideoChannel	integer	The video channel.
<b>Collect</b> .Windows[ <b>winno</b> ].VideoStream	string	The range is {"Main", "Extra1", "Extra2", "Extra3", "Auto"}.
<b>Collect</b> .Windows[ <b>winno</b> ].AudioChannel	integer	The audio channel.
<b>Collect</b> .Windows[ <b>winno</b> ].AudioStream	string	The range is {"Main", "Extra1", "Extra2", "Extra3", "Auto"}.

## 8.1 Video Analyse

### 8.1.1 Get Video Analyse Capability

Table 8-1

Syntax	http://<server>/cgi-bin/devVideoAnalyse.cgi?action=getcaps&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get video analyse capabilities.
Example	http://192.168.1.108/cgi-bin/devVideoAnalyse.cgi?action=getcaps&channel=1
Success Return	<pre> caps.CalibrateBoxes[0]=2 caps.CalibrateBoxes[1]=3 caps.ComplexSizeFilter=false caps.MaxCelebateAreas=10 caps.MaxExcludeRegions=0 caps.MaxInternalOptions=512 caps.MaxModules=1 caps.SupportGlobalDeviceParam=1 caps.MaxPointOfLine=20 caps.MaxPointOfRegion=20 caps.MaxRules=10 caps.MaxStaffs=4 caps.SpecifiedObjectFilter=true caps.SupportedRules[0]=CrossLineDetection caps.SupportedRules[1]=CrossRegionDetection caps.SupportedRules[2]=LeftDetection caps.SupportedRules[3]=TakenAwayDetection caps.SupportedScene[0]=Normal caps.SupportedScene[1]=FaceDetection caps.SupportedScene[2]=VideoDiagnosis caps.SupportedScenes.FaceDetection.SupportedCalibrateParams.Groud.HorizontalStaffs[0]=0 caps.SupportedScenes.FaceDetection.SupportedCalibrateParams.Groud.HorizontalStaffs[1]=0 caps.SupportedScenes.FaceDetection.SupportedCalibrateParams.Groud.VerticalStaffs[0]=0 caps.SupportedScenes.FaceDetection.SupportedCalibrateParams.Groud.VerticalStaffs[1]=0 caps.SupportedScenes.StereoNumber.SupportedRules.ManNumDetection.SupportLocalDataStore=false caps.SupportedScenes.NumberStat.CameraType=1 </pre>

	caps.SupportedScenes.NumberStat.SupportedRules.NumberStat.MaxRules=8 caps.SupportedScenes.StereoBehavior.OnlyFindLastEvtList[0]="ManStandDetection"
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

#### Appendix

ParamName	ParamValue type	Description
caps.SupportedScenes.StereoBehavior.OnlyFindLastEvtList	array<string>	event type list in which last ones are supported to be inquired.

## 8.1.2 Video Analyse Global

- Get video analyse global config

Table 8-2

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoAnalyseGlobal
Method	GET
Description	Get video analyse global config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoAnalyseGlobal
Success Return	<b>head</b> .Scene.Type=Normal <b>head</b> .Scene.PtzPresetId=1 <b>head</b> .Scene.Depth=Far <b>head</b> .Scene.Detail.CameraAngle=30 <b>head</b> .Scene.Detail.CameraDistance=10.000000 <b>head</b> .Scene.Detail.CameraHeight=6.200000 <b>head</b> .TimePeriod.Day[0]=8:00:00 <b>head</b> .TimePeriod.Day[1]=20:00:00 <b>head</b> .TimePeriod.Night[0]=20:00:00 <b>head</b> .TimePeriod.Night[1]=8:00:00 ...
Comment	Parameters in Response : <b>head</b> =table.VideoAnalyseGlobal[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).

- Set video analyse global config

Table 8-3

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set video analyse global config.

Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoAnalyseGlobal[0].Scene.Type=Normal&VideoAnalyseGlobal[0].Scene.PtzPresetId=1
Success Return	OK
Comment	<p>Parameters in URL: paramName and paramValue are as table below.</p> <p>In table below, head = <b>VideoAnalyseGlobal[ChannelNo]</b> <b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). ParamName start with head.Scene.Detail depends on head.Scene.Type.</p>

#### Appendix

ParamName	ParamValue type	Description
<b>head.Scene.Type</b>	string	Scene class, the range is { "", "Normal", "Indoor", "ATM", "Traffic", "FaceRecognition", "FaceDetection", "Prison", "NumberStat", "HeatMap", "VideoDiagnosis", "VehicleAnalyse", "TrafficPatrol", "CourseRecord", "Vehicle", "ObjectDetect", "VehicleCompare" }
<b>head.Scene.PtzPresetId</b>	integer	Range is 0—255, 0 means that the scene is unassociated with PTZ.
<b>head.Scene.Depth</b>	string	Picture distance feature, the range is { "Normal", "Far", "Middle", "Near" }
<b>head.Scene.Detail.Value</b>	string	Detail config of a scene. For example, when Scene.Type is "Normal", it's detail includes CameraAngle, CameraDistance, CameraHeight, etc.
<b>head.TimePeriod.Day[0]</b>	string	The start time of Day, it's format is hh:mm:ss
<b>head.TimePeriod.Day[1]</b>	string	The end time of Day
<b>head.TimePeriod.Night[0]</b>	string	The start time of Night, it's format is hh:mm:ss
<b>head.TimePeriod.Night[1]</b>	string	The end time of Night

### 8.1.3 Video Analyse Rule

- Get video analyse rule

Table 8-4

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoAnalyseRule
Method	GET
Description	Get video analyse rules config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoAnalyseRule
Success Return	<p><b>head.Name</b>= line1</p> <p><b>head.Type</b>=CrossLineDetection</p> <p><b>head.VideoAnalyseRule[0][0].Enable</b> =true</p> <p><b>head.VideoAnalyseRule[0][0].EventHandler</b>= (output of EventHandler is described in</p>

	GetEventHandler) ...
Comment	Parameters in Response : <b>head</b> =table.VideoAnalyseRule[ <b>ChannelNo</b> ] [ <b>RuleNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>RuleNo</b> =rule index.

- Set video analyse rule

Table 8-5

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set video analyse rules config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoAnalyseRule[0][0].Name=myAnalyseRule1&VideoAnalyseRule[0][0].Type=CrossLineDetection
Success Return	OK
Comment	Parameters in URL: paramName and paramValue are as table below. In table below, <b>head</b> =VideoAnalyseRule[ <b>ChannelNo</b> ] [ <b>RuleNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>RuleNo</b> =rule index. ParamName start with head.Config is only effective with {"CrossLineDetection", "CrossRegionDetection", "LeftDetection", "TakenAwayDetection"}.

#### Appendix

ParamName	ParamValue type	Description
<b>head</b> .Name	string	Rule name, it must be unique.
<b>head</b> .Type	string	The range is {"CrossLineDetection", "CrossRegionDetection", "LeftDetection", "TakenAwayDetection", "VideoAbnormalDetection", "FaceDetection", "AudioMutation", "AudioAnomaly", "VideoUnFocus", "WanderDetection", "RioterDetection", "ParkingDetection", "MoveDetection", "NumberStat", "HeatMap", "FaceAttribute", "VehicleDetect", "NonMotorDetect", "HumanTrait", "ObjectPlacement", "ObjectRemoval" }.
<b>head</b> .Enable	bool	Enable/Disable this rule.
<b>head</b> .EventHandler	—	Setting of EventHandler is described in <b>SetEventHandler</b> .
<b>head</b> .Config.DetectLine[0][0]	integer	The start point of DetectLine 0;
<b>head</b> .Config.DetectLine[0][1]	integer	The end point of DetectLine 0;

ParamName	ParamValue type	Description
<b>head</b> .Config.DetectLine[1][0]	integer	The start point of DetectLine 1;
<b>head</b> .Config.DetectLine[1][1]	integer	The end point of DetectLine 1;
<b>head</b> .Config.Direction	string	The range is {"LeftToRight", "RightToLeft", "Both"}.
<b>head</b> .Config.SizeFilter.MaxSize[0]	integer	Maximum width. The width of the object must not be beyond maximum width. Adapt to {"CrossLineDetection", "CrossRegionDetection", "LeftDetection", "TakenAwayDetection", "FaceDetection", "WanderDetection", "RioterDetection", "ParkingDetection", "MoveDetection"}.
<b>head</b> .Config.SizeFilter.MaxSize[1]	integer	Maximum height. The height of the object must not be beyond maximum height.
<b>head</b> .Config.SizeFilter.MinSize[0]	integer	Minimum width. The width of the object must not be less than minimum width.
<b>head</b> .Config.SizeFilter.MinSize[1]	integer	Minimum height. The height of the object must not be beyond minimum height.
<b>head</b> .Config.DetectRegion[0][0]	integer	The start point of DetectRegion 0; Adapt to {"CrossRegionDetection", "LeftDetection", "TakenAwayDetection", "WanderDetection", "RioterDetection", "ParkingDetection", "MoveDetection"}.
<b>head</b> .Config.DetectRegion[0][1]	integer	The end point of DetectRegion 0;
<b>head</b> .Config.DetectRegion[1][0]	integer	The start point of DetectRegion 1;
<b>head</b> .Config.DetectRegion[1][1]	integer	The end point of DetectRegion 1;
<b>head</b> .Config.DetectRegion[2][0]	integer	The start point of DetectRegion 2;
<b>head</b> .Config.DetectRegion[2][1]	integer	The start point of DetectRegion 2;
<b>head</b> .Config.MinDuration	integer	Range is 1—600, adapt to {"LeftDetection", "TakenAwayDetection", "WanderDetection"}. Range is 10-300, adapt to {"RioterDetection"}. Range is 6-300, adapt to {"ParkingDetection"}.
<b>head</b> .Config.Sensitivity	integer	Range is 1—10, adapt to {"RioterDetection", "MoveDetection"}.
<b>head</b> .Config.EnterThreshold	integer	Range is 0 — 100000000, adapt to {"NumberStat"}.
<b>head</b> .Config.ExitThreshold	integer	Range is 0 — 100000000, adapt to {"NumberStat"}.
<b>head</b> .Config.OutsideThreshold	integer	Range is 0 — 100000000, adapt to {"NumberStat"}.

ParamName	ParamValue type	Description
<b>head</b> .Config.DirectionStats.Enable	bool	Enable/Disable objectdetect Direction. Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.DirectionStats.DetectLine[0][0]	integer	The start point of DetectLine 0, Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.DirectionStats.DetectLine[0][1]	integer	The start point of DetectLine 0, Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.DirectionStats.DetectLine[1][0]	integer	The start point of DetectLine 1, Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.DirectionStats.DetectLine[1][1]	integer	The start point of DetectLine 1, Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.DirectionStats.Direction	string	The range is {"LeftToRight", "RightToLeft", "Both"}. Rule range is {"VehicleDetect", "NonMotorDetect", "HumanTrait"}.
<b>head</b> .Config.FilterUnAliveEnable	bool	Enable the non-living filtering function. It is false by default.
<b>head</b> .Config.EyesDistThreshold	uint32	Set filtering threshold of pupillary distance. If the actual pupillary distance is less than the threshold, it will be filtered. Refer to GB/T 35678-2017: Value range [50–no upper limit] required for face recognition application image technology. The default value 0 indicates no filtering.
<b>head</b> .Config.HelmetEnable	bool	Enable safety helmet detection. With the function enabled, the face recognition result carries information related to safety helmet detection. It is false by default.
<b>head</b> .Config.TempSwitch	uint8	Select temperature monitoring status: 0: Disable temperature monitoring 1: Normal temperature monitoring 2: Debug temperature monitoring
<b>head</b> .Config.TempModel	uint8	Temperature monitoring mode (valid when TempSwitch is not 0): 0: Automatical mode

ParamName	ParamValue type	Description
		1: Thermal image detection mode 2: Calibration mode
<b>head</b> .Config.TempStrategy.TempValueMax	double	Maximum value of normal face temperature range
<b>head</b> .Config.TempStrategy.TempValueMin	double	Minimum value of normal face temperature range
<b>head</b> .Config.TempStrategy.TempType	uint8	Strategy type for calculating face temperature: 0: Use the highest temperature 1: Use the average temperature
<b>head</b> .Config.TempStrategy.SlideNum	uint8	The number of sliding cache frames. 0 means no sliding, and the maximum cache is 32 frames.
<b>head</b> .Config.TempStrategy.HightTempStrategy.TempDetectRegion	uint8	Parameter for maximum temperature strategy (valid when tempType is 0). Calculate the highest temperature area: 0: Forehead 1: Whole face
<b>head</b> .Config.TempStrategy.AverageTempStrategy.StrategyType	uint8	Parameter for maximum temperature strategy (valid when tempType is 1). Average temperature strategy type: 0: Center point 1: high temperature point
<b>head</b> .Config.TempStrategy.AverageTempStrategy.PointNum	uint8	Parameter for average temperature strategy (valid when tempType is 1). The number of points used to find the average value, and it is the square of the integer, such as 4, 9, and 16.
<b>head</b> .Config.TempStrategy.AverageTempStrategy.EnableFilter	bool	Parameter for average temperature strategy (valid when tempType is 1). Whether to enable the removal of the highest temperature and the lowest temperature to take the average value: False: Disable True: Enable
<b>head</b> .Config.FilterMaskUnAliveEnable	bool	Select whether to enable liveness detection for person wearing mask. This value is related to FilterUnAliveEnable. When FilterUnAliveEnable is true (liveness detection is enabled), this field is valid.

## 8.1.4 Get Last Event Info

- Get Last Event Info

Table 8-6

Syntax	http://<server>/cgi-bin/devVideoAnalyse.cgi?action=getLastEventInfo&channel=< <b>ChannelNo</b> >&ClassName=< <b>ClassNameStr</b> >&EventName=< <b>EventNameStr</b> >
Method	GET
Description	Get last event info
Example	http://192.168.1.108/cgi-bin/devVideoAnalyse.cgi?action=getLastEventInfo&channel=1&ClassName=StereoBehavior&EventName=ManStandDetection
Success Return	State=1 EventInfo.Code=ManStandDetection EventInfo.Action=Start
Comment	Parameters in Response : <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>ClassNameStr</b> : str, means Class name. <b>EventNameStr</b> : str, means Event name.

## 8.1.5 [Config] GlobalDeviceParam

- Get Global Device Param

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=GlobalDeviceParam		
Method	GET		
Description	Get Global Device Param		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
table	object	R	
+LocateHeight	Float	R	Device locate Height
[ Example ]			
Request	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=GlobalDeviceParam		
Response	table.GlobalDeviceParam.LocateHeight=10.5		

- Set Global Device Param

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]		
Method	GET		
Description	Set Global Device Param		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description

GlobalDeviceParam	object	R	Global Device Param
+LocateHeight	float	R	Device locate Height
[ Response Params ] (OK at Body )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://<server>/cgi-bin/configManager.cgi?action=setConfig&GlobalDeviceParam.LocateHeight = 10.5		
Response	Ok		

## 8.1.6 getTemplateRule

- Gets all intelligent rule configuration templates and default values under the specified category

<b>Request URL</b>	http://<server>/cgi-bin/VideoInAnalyse.cgi?action=getTemplateRule			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Class	Char[32]	R	the same meaning with Type in config VideoAnalyseGlobal "Normal" "Traffic" "TrafficPatrol" "FaceDetection"(shared by face detect and face recognition) in added, Web can use "ALL" for all rules.	"Normal"
Channel	integer	R	video channel, which starts from 1.	1
Example				
http://<server>/cgi-bin/VideoInAnalyse.cgi?action=getTemplateRule&Channel=1&Class=Normal				

<b>Response Params ( key=value format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Rule	object	R	rule info	
+Normal	object	R	business class	
++CrossRegionDetection	object	R	cross region templet	
++CrossLineDetection	object	R	default config of crossline format as the same as each object in the second dimension of	

VideoAnalyseRule				
+++Config	object	R	detail of rule config	
++++SizeFilter	object	R		
+++++MinSize	int[2]	R		[0, 0]
+++++MaxSiz	int[2]	R		[8191, 8191]
e				
++++DetectLin	object[]	R		
e				
++++Direction	char[]	R		"Both"
+++PtzPresetI	integer	R	Preset number	0
d				
+++ObjectTyp	char[][]	R	type of detect object	["Unknown"]
es				
+++Type	char[64]	R	rule type	"CrossLineDetection"
+++Class	char[16]	R	business plan class	"Normal"
+++Enable	bool	R	enable the rule	false
+++Id	integer	R	rule id	0
Example				
Rule.Normal.CrossLineDetection.Id=0				
Rule.Normal.CrossLineDetection.Enable=false				
Rule.Normal.CrossLineDetection.Class=Normal				
Rule.Normal.CrossLineDetection.Type=CrossLineDetection				
Rule.Normal.CrossLineDetection.ObjectTypes[0]=Unknown				
Rule.Normal.CrossLineDetection.PtzPresetId=0				
Rule.Normal.Config.Direction=Both				
Rule.Normal.Config.SizeFilter.MaxSize[0]=[8191, 8191]				
Rule.Normal.Config.SizeFilter.MinSize[0]=[0, 0]				
...				

### 8.1.7 [Config] IntelliSchemeTourEnableSetting

- Request

<b>Syntax</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig&IntelliSchemeTour.Enable=<enableValue>			
<b>Method</b>	SET			
<b>Request Params</b>	key=value format at URL			
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
IntelliSchemeTour	object	O	IntelliSchemeTour Enable Setting	
+Enable	bool	R	IntelliSchemeTour Enable Setting	True
Example				
http://172.27.2.72/cgi-bin/configManager.cgi?action=setConfig&IntelliSchemeTour.Enable=true				

- Response

<b>Request Params</b>	key=value format at URL			
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>

Example
OK

## 8.1.8 Getting Intelligent Tour Plan

Get intelligent tour plan

- Request

<b>Syntax</b>	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=IntelliSchemeTour			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
IntelliSchemeTour	object	Yes	Intelligent package tour plan	
<b>Example</b>				
http://172.27.2.72/cgi-bin/configManager.cgi?action=getConfig&name=IntelliSchemeTour				

- Response

<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
IntelliSchemeTour	object	No	Intelligent package tour plan. No channel sensitive.	
+Enable	bool	No	Whether the package tour plan is enabled.	true
+TourPriMode	enumchar[32]	No	Tour priority mode  enumchar[32]{  "Switch": Scene priority, tour in strict accordance with the defined tour time.  "Detect": Detection priority, with the complete set of data as the condition for scene switching upon completion of scene detection (For illegal parking detection cameras. Multiple images may be needed in one	"Switch"

			<p>scene. In this case, the camera can rotate to the next scene after all images are captured).</p> <p>"Tour": Tour priority, similar to detection priority, with a set of data as the condition for scene switching upon completion of scene detection (both close shot and long shot available. Apply close shot first and then long shot according to the tour path).</p> <p>}  For details, refer to the capability set IntelliScheme.getCapabilities.</p>	
+IdleWaitingTime	uint32	No	Idle waiting time, that is, the idle delay time until the tour plan continues to take effect after a user stops operating the device, in second(s).	10
+SceneSwitching Time	uint32	No	Scene switching time. When the tour plan is enabled and the priority mode is either detection priority or tour priority, it will automatically rotate to the next scene after an object cannot be detected	20

			for a certain period of time.	
+TourPlan	object[7][16]	No	Package tour plan  2D array. The first dimension indicates the day of a week, totally 7 weekdays, and the second dimension indicates the tour groups supported that day.	
++TimeSection	TimeSection	No	Period of time. 00:00:00 indicates the start time and 23:59:59 indicates the end time.	"1 00:00:00-23:59:59"
++TourPath	object[]	No	Tour path  1D array, to adjust the order of packages. Up to 10 packages are supported. It is null by default and added by a user. Field values are specified by a user.	null
+++SchemeID	uint32	No	Intelligent package ID, value range: 1-300.	1
+++Duration	uint32	No	Retention time, value range: 30-36,000 s or 0.  It is 0 by default if this field is unavailable. Indicates that the switching has not been made during the period of time.	600
+++Speed	uint32	No	The speed of the camera rotating to the preset when a package includes a preset channel of a	7

			PTZ camera, value range: 1-10 speed levels.	
--	--	--	---	--

**Example**

```

table.IntelliSchemeTour.Enable=true
table.IntelliSchemeTour.IdleWaitingTime=10
table.IntelliSchemeTour.SceneSwitchingTime=20
table.IntelliSchemeTour.TourPlan[0][0].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][1].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][2].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][3].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][4].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][5].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][6].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][7].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][8].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][9].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][10].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][11].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][12].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][13].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][14].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[0][15].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][0].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][1].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][2].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][3].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][4].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][5].TimeSection=1 00:00:00-23:00:00
table.IntelliSchemeTour.TourPlan[1][6].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][7].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][8].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][9].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][10].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][11].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][12].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][13].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][14].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[1][15].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][0].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][1].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][2].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][3].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][4].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][5].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][6].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[2][7].TimeSection=1 00:00:00-00:00:00

```



```

table.IntelliSchemeTour.TourPlan[5][7].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][8].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][9].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][10].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][11].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][12].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][13].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][14].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[5][15].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][0].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][1].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][2].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][3].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][4].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][5].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][6].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][7].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][8].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][9].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][10].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][11].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][12].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][13].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][14].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPlan[6][15].TimeSection=1 00:00:00-00:00:00
table.IntelliSchemeTour.TourPriMode=Switch

```

### 8.1.9 Exporting Intelligent Diagnosis, Allowlist, and Blocklist Information

Export information, such as intelligent diagnosis, allowlist, and blocklist information.

<b>Request URL</b>	http://<server>/cgi-bin/api/ImExport/exportData			
<b>Method</b>	POST			
<b>Request Params (JSON format in body)</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
type	enumint	O	Exported data type. enumint{ 0 : License plate allowlist 1 : License plate blocklist 2 : License plate blocklist and allowlist 3: Voice talk contacts 4: Custom password 5: Card recordset	0

			6: Face information for access control 7: Fingerprints for access control 8: IPC information of voice talk 9: Calling person information 10: Public password recordset 11: Operation and maintenance information }	
<b>Request Example</b>				
<pre>{   "type": 1 }</pre>				

<b>Response Params (binary in body)</b>				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
HTTP/1.1 200 OK Server: Device/1.0 Content-Type: application/octet-stream Content-Length: <length>  < binary data >				

## 8.1.10 Importing Intelligent O&M, Allowlist, and Blocklist Information

Import intelligent O&M, allowlist, and blocklist information.

<b>Request URL</b>	http://<server>/cgi-bin/api/ImExport/importData			
<b>Method</b>	POST			
<b>Request Params (multipart in body)</b>				
Name	Type	R/O	Description	Example
type	uint32	R	Imported data type enumint{ 0 : License plate allowlist 1 : License plate blocklist 2 : License plate blocklist and allowlist 3: Voice talk contacts 4: Custom password 5: Card recordset	0

			6: Face information for access control 7: Fingerprint for access control 8: IPC information of voice talk 9: Calling person information 10: Public password recordset 11: Operation and maintenance information }	
--	--	--	---	--

#### Request Example

```

POST http://<server>/cgi-bin/api/ImExport/importData HTTP/1.1
User-Agent: client/1.0
Content-Type: multipart/x-mixed-replace; boundary=<boundary>

--<boundary>
Content-Type: application/json
Content-Length: 12

{
  "type": 0
}
--<boundary>
Content-Type: application/octet-stream
Content-Length: 800

< binary data >
--<boundary>

```

#### Response Params ( JSON format in body )

Name	Type	R/O	Description	Example
<b>Response Example</b>				
{ }				

## 8.2 Number of People

### 8.2.1 Video Widget Number Status

- Get video widget number status

Table 8-7

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=VideoWidgetNumberStat
Method	GET
Description	Get OSD config when display number status information of people.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=VideoWidgetNumberStat
Success Return	<b>head</b> .EncodeBlend=true <b>head</b> .ShowEnterNum=true <b>head</b> .ShowExitNum=true <b>head</b> .TextAlign=0 ...
Comment	Parameters in Response : <b>head</b> =table.VideoWidgetNumberStat[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set video widget number status

Table 8-8

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set OSD config when display number status information of people.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&VideoWidgetNumberStat[0].EncodeBlend=true&VideoWidgetNumberStat[0].ShowEnterNum=true
Success Return	OK
Comment	Parameters in URL: paramName and paramValue are as table below. In table below, <b>head</b> =VideoWidgetNumberStat[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1).

#### Appendix

ParamName	ParamValue type	Description
<b>head</b> .EncodeBlend	bool	Enable/Disable
<b>head</b> .ShowEnterNum	bool	Enable/Disable
<b>head</b> .ShowExitNum	bool	Enable/Disable
<b>head</b> .TextAlign	integer	0 for left, 2 for right

## 8.2.2 Get Heat Map Information

Table 8-9

Syntax	http://<server>/cgi-bin/heatMap.cgi?action=getPicByTime&channel=<ChannelNo>&St
--------	--

	artTime=< <b>start</b> >&EndTime=< <b>end</b> >[&PtzPresetId=< <b>PtzPresetId</b> >]
Method	GET
Description	Get binary data of heat map.
Example	http://192.168.1.108/cgi-bin/heatMap.cgi?action=getPicByTime&channel=1&StartTime=2015-08-20%2000:00:00&EndTime=2015-08-21%2023:59:59
Success Return	Content-Type: application/binarytet-stream Content-Length:< <b>heatMap size</b> > < <b>HeatMap data</b> >
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>start/end</b> : the start/end time of Heat Map info. 24 hour Format, as: yyyy-mm-dd hh:mm:ss. <b>PtzPresetId</b> : the ptz preset index which starts from 1.  Parameters in Response: <b>heatMap size</b> : width*height + 16. <b>HeatMap data</b> : format as table below.

#### Appendix: HeatMap Data Format

0	1	2	3	4	...	15	16	17	18	...
Width		Height		Reserved			Data: every byte symbolize a pixel			

### 8.2.3 Get People Heat Map Information

Table 8-10

URL	http://<server>/cgi-bin/videoStatServer.cgi?action=getHeatMap		
Method	GET		
Description	Get People Heat Map Information		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
channel	int	O	Video channel index which starts from 1, default is 1.
PlanID	int	O	The plan id, only valid for dome camera.
StartTime	string	R	The start time, ex: "2010-05-12 20:00:00"
EndTime	string	R	The end time, ex: "2010-05-12 22:00:00"
HeatMapType	string	R	The heat map type, it can be: "AverageStayTime" : average stay time heat map "HumanStatistics" : human statistics heat map "HumanTrack" : human track heat map
[ Response Params ] ( binary data )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=getHeatMap&channel=1&PlanID=2&StartTime=2010-05-12%2020:00:00&EndTime=2010-05-12%2022:00:00&HeatMapType=AverageStayTime		
Response	HTTP/1.1 200 OK Server: Device/1.0		

Content-Type: application/octet-stream Content-Length: <length>  <heat map binary data>
--

**Appendix A:** The heat map binary data format when HeatMapType is AverageStayTime.

octet	0	1	2	3	4~7	8~11	12~15
value	version	channel	preset	reserved	total data length	total lines	lines in this packet
octet	16~31						
value	reserved						
octet	32~35			36~39		40~43	44~44+4* (N-1)
value	line number			data num in this line		data 1	data N
octet	44+4* (N-1) +1~ 44+4* (N-1) +4			.....		.....	.....
value	next line number			data num in next line		data 1	data N

**Note:** If “lines in this packet” is 0, then all data has been sent, and connection will be closed.

**Appendix B:** The heat map binary data format when HeatMapType is HumanStatistics.

octet	0	1	2	3	4~7	8~11	12~15
value	version	channel	preset	reserved	total data length	total lines	lines in this packet
octet	16~31						
value	reserved						
octet	32~35			36~39		40~43	44~44+4* (N-1)
value	line number			data num in this line		data 1	data N
octet	44+4* (N-1) +1~ 44+4* (N-1) +4			.....		.....	.....
value	next line number			data num in next line		data 1	data N

**Note:** If “lines in this packet” is 0, then all data has been sent, and connection will be closed.

**Appendix C:** The heat map binary data format when HeatMapType is HumanTrack.

octet	0	1	2	3	4~7	8~11	12~15
value	version	channel	preset	reserved	total data length	total tracks	tracks in this packet
octet	16~31						
value	reserved						
octet	32~35			36~39		40~43	44~44+4* (N-1)
value	Object ID			track point num in this object		track point 1	track point N

octet	44+4* (N-1) +1~ 44+4* (N-1) +4	.....	.....	.....
value	next object ID	track point num in next object	track point 1	track point N

**Note:** If “tracks in this packet” is 0, then all data has been sent, and connection will be closed.

## 8.2.4 Subscribe People Realtime Trace Information

Table 8-11

URL	http://<server>/cgi-bin/videoStatServer.cgi?action=attachRealTraceProc		
Method	GET		
Description	Subscribe the people realtime trace information.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
channel	int	O	Video channel index which starts from 1, default is 1.
heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat meage are "Heartbeat". If this parameter is not present, its default value is 60.
[ Response Params ] (multipart binary data)			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=attachRealTraceProc&channel=1&heartbeat=5		
Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;  &lt;realtime trace binary data&gt; --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;  &lt;realtime trace binary data&gt; --&lt;boundary&gt;  ... </pre>		

**Appendix A:** The realtime trace binary data format.

octet	0	1	2	3	4~7	8~11	12~15
value	version	chan nel	pres et	reser ved	total data length	reserved	object num in this packet
octet	16~31						
value	reserved						
octet	32~35			36~37	38~39	40~43	44~47
value	object ID 1			object coord x,	object coord y	current time in utc	reserved
octet	48~51			52~53	54~55	56~57	58~61
value	object ID 2			object coord x	object coord y	current time in utc	reserved

## 8.2.5 Get People Histroy Trace Information

Table 8-12

URL	http://<server>/cgi-bin/videoStatServer.cgi?action=getHistoryTrace		
Method	GET		
Description	Get people history trace information.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
channel	int	O	Video channel index which starts from 1, default is 1.
PlanID	int	O	The plan id, only valid for dome camera.
StartTime	string	R	The start time, ex: "2010-05-12 20:00:00"
EndTime	string	R	The end time, ex: "2010-05-12 22:00:00"
[ Response Params ] ( multipart binary data )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=getHistoryTrace&channel=1&PlanID=2&StartTime=2010-05-12%2020:00:00&EndTime=2010-05-12%2022:00:00		
Response	HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=<boundary> Connection: closed  --<boundary> Content-Type: application/octet-stream Content-Length: <length>  <history trace binary data> --<boundary> Content-Type: application/octet-stream Content-Length: <length>		

	<pre>&lt;history trace binary data&gt; --&lt;boundary&gt; ...</pre>
--	---

**Appendix A:** The history trace binary data format.

octet	0	1	2	3	4~7	8~11	12~15
value	version	channel	preset	reserved	total data length	total object num	object num in this packet
octet	16~31						
value	reserved						
octet	32~35		36~37		38~39	40~43	44~47
value	object ID 1		object coord x	object coord y	current time in utc	reserved	
octet	48~51		52~53		54~55	56~57	58~61
value	object ID 2		object coord x	object coord y	current time in utc	reserved	

**Note:** If “object num in this packet” is 0, then all data has been sent, and connection will be closed.

## 8.2.6 Subscribe Heat Map Raw Data

Table 8-13

URL	http://<server>/cgi-bin/HeatMapManager.cgi?action=attachRaw		
Method	GET		
Description	Subscribe the heap map raw data.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
channel	int	O	Video channel index which starts from 1, default is 1.
heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat meage are "Heartbeat". If this parameter is not present, its default value is 60.
[ Response Params ] (multipart binary data)			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/HeatMapManager.cgi?action=attachRaw&channel=1&heartbeat=5		
Response	<pre>HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;</pre>		

	<pre> &lt;heap map raw data&gt; --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;  &lt;heap map raw data&gt; --&lt;boundary&gt; ... </pre>
--	--

**Appendix A:** The heap map raw data format.

octet	0	1	2	3	4 ~ 23
value	width		height		StartTime, ex: "2012-01-04 00:00:00"
octet	24~43				44 ~ 47
value	EndTime, ex: "2012-01-04 01:00:00"				reserved
octet	48	49	...		
value	Data: every byte symbolize a pixel				

## 8.2.7 Clear statistics in time section

Table 8-14

URL	http://<server>/cgi-bin/videoStatServer.cgi?action=clearSectionStat[&AreaID=<AreaID>][&PtzPresetId=<PtzPresetId>]		
Method	GET		
Description	Clear statistics in time section		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
channel	int	O	Video channel index which starts from 1, default is 1.
AreaID	int	O	The area index which starts from 1, valid for multiple area device, if omit means clear all area statistics.
PtzPresetId	int	O	The ptz preset index which starts from 1.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=clearSectionStat&AreaID=2		
Response	OK		

## 8.3 FishEye

### 8.3.1 Get FishEye Capability

The method described in the Section “4.5.12 Get video input capability” should be used first. In the response, it will contain these message "caps.FishEye=false", if the value of the FishEye is true, Then you can use the method described below to get the detail capability.

Table 8-15

Syntax	http://<server>/cgi-bin/devVideoInput.cgi?action=getCapsEx&channel=< <b>ChannelNo</b> >&name=VideoInFishEye
Method	GET
Description	Get fisheye capability.
Example	http://192.168.1.108/cgi-bin/devVideoInput.cgi?action=getCapsEx&channel=1&name=VideoInFishEye
Success Return	<p>caps.<b>Type</b>=Chip  caps.<b>MountMode</b>[0]=WallMode  caps.<b>MountMode</b>[1]=CeilMode  .....  caps.<b>CalibrateMode</b>[0]=Original  caps.<b>CalibrateMode</b>[1]=Config  caps.<b>CalibrateMode</b>[2]=Panorama  .....  caps.<b>EPtzCmd</b>[0]=Up  caps.<b>EPtzCmd</b>[1]=Down  .....</p>
Comment	<p>Parameters in Response:  <b>Type</b>: string, it can be Chip, Plugin, and ChipAndPlugin. Chip means only support calibrate by device.  Plugin means only support calibrate by plugin. ChipAndPlugin means support both.  <b>MountMode</b>: string and array.and <b>MountMode</b> means the install mode, it can be WallMode,CeilMode, FloorMode,180CeilMode.  <b>CalibrateMode</b>: string and array. It can be Original, Config, Panorama, DoublePanorama, OriginalPlusThreeEPtzRegion, Single, FourEPtzRegion, TwoEPtzRegion, and Normal.  <b>EPtzCmd</b>: string and array. It can be ZoomIn, ZoomOut, Up, Down, Left, Right, RotateClock, RotateAntiClock, Stop, TapView, and ShowRegion.</p>

### 8.3.2 FishEye Setting

- Get FishEye config

Table 8-16

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=FishEye
--------	---

Method	GET
Description	Get FishEye config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=FishEye
Success Return	<b>head.PlaceHolder=1</b> <b>head.CalibrateMode=Original</b> .....
Comment	Parameters in Response: <b>head</b> = table.FishEye[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>PlaceHolder</b> : integer, it can be 1(Ceilmode), 2(WallMode), 3(FloorMode). <b>CalibrateMode</b> : string. It can be Original, Config, Panorama, DoublePanorama, OriginalPlusThreeEPtzRegion, Single, FourEPtzRegion, TwoEPtzRegion, and Normal.

- Set FishEye config

Table 8-17

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set FishEye config.
Example	Change the placeholder setting of channel 1: http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&FishEye[0].Placeholder=1  Change the calibratemode setting of channel 1: http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&FishEye[0].CalibrateMode=Panorama  Also can use http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&FishEye[0].CalibrateMode=Panorama&FishEye[0].Placeholder=1 to change the placeholder and calibratemode in one message.
Success Return	OK
Comment	—

## 8.4 CrowdDistriMap

### 8.4.1 Get Channel Caps

Table 8-18

Syntax	http://<server>/cgi-bin/crowdDistriMap.cgi?action=getCaps
Method	GET
Description	Get Channel CrowdDistriMap Caps.
Example	http://192.168.1.108/cgi-bin/crowdDistriMap.cgi?action=getCaps

Success Return	CrowdCapsList[0].channel=< <b>ChannelNo</b> > CrowdCapsList[0].Support=< <b>Support</b> > CrowdCapsList[1].channel=< <b>ChannelNo</b> > CrowdCapsList[1].Support=< <b>Support</b> > ...
Comment	Parameters in URL and Response: <b>ChannelNo</b> : video channel index <b>Support</b> : true or false, support or not

## 8.4.2 Subscribe to Realtime Crowd Stat

Table 8-19

Syntax	http://<server>/cgi-bin/crowdDistriMap.cgi?action=attach&channel=< <b>ChannelNo</b> >[&heartbeat=< <b>Heartbeat</b> >]
Method	GET
Description	Subscribe the crowd distribute map information, return info at regular time.
Example	http://<server>/cgi-bin/crowdDistriMap.cgi?action=attach&channel=1&heartbeat=5
Success Return	HTTP/1.1 200 OK Content-Type: multipart/x-mixed-replace; boundary=< <i>boundary</i> > Connection: close  --< <i>boundary</i> > Content-Type: text/plain Content-Length: <data length>  CrowdStatData[0].Channel=1 CrowdStatData[0].GlobalPeopleNum =10 CrowdStatData[0].RegionNum =1 CrowdStatData[0].RegionPeopleList[0].RegionID=0 CrowdStatData[0].RegionPeopleList[0].Region[0][0]=10 CrowdStatData[0].RegionPeopleList[0].Region[0][1]=10 CrowdStatData[0].RegionPeopleList[0].Region[1][0]=10 CrowdStatData[0].RegionPeopleList[0].Region[1][1]=100 ... CrowdStatData[0].RegionPeopleList[0].RegionPeopleNum=100 CrowdStatData[0].CrowdEventNum =2 CrowdStatData[0].CrowdList[0].Center=2 CrowdStatData[0].CrowdList[0].Radius=2 CrowdStatData[0].RegionEventNum =2 CrowdStatData[0].RegionList[0].Region[0][0]=10 CrowdStatData[0].RegionList[0].Region[0][1]=10 CrowdStatData[0].RegionList[0].Region[1][0]=10 CrowdStatData[0].RegionList[0].Region[1][0]=100 ... CrowdStatData[0].RegionList[0].RegionID=0 CrowdStatData[0].RegionList[0].PeopleNum=100

	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;data length&gt;  CrowdStatData[0].Channel=1 ...</pre>
Comment	<p>Parameters in URL and Response</p> <p><b>ChannelNo</b>: integer, video channel index to subscribe</p> <p>Region is <b>Polygon</b>, has a list of <b>points</b>, every point has <b>x</b> and <b>y</b>, so Region[0][0] is first point's x, Region[0][1] is first point's y, Region[1][0] is second point's x, Region[1][1] is second point's y, and so on</p> <p><b>Heartbeat</b>: integer, range is [1,60],unit is second.If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client,the heartbeat meage are "Heartbeat".</p>

### 8.4.3 Get Current Crowd Stat

Table 8-20

Syntax	http://<server>/cgi-bin/crowdDistriMap.cgi?action=getSummary&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get crowd distribute map information, return info only once.
Example	http://<server>/cgi-bin/crowdDistriMap.cgi?action=getSummary&channel=1
Success Return	<pre>CrowdStatData[0].Channel=1 CrowdStatData[0].GloabalPeopleNum =10 CrowdStatData[0].RegionNum =1 CrowdStatData[0].RegionPeopleList[0].RegionID=0 CrowdStatData[0].RegionPeopleList[0].Region[0][0]=10 CrowdStatData[0].RegionPeopleList[0].Region[0][1]=10 CrowdStatData[0].RegionPeopleList[0].Region[1][0]=10 CrowdStatData[0].RegionPeopleList[0].Region[1][1]=100 ... CrowdStatData[0].RegionPeopleList[0].RegionPeopleNum=100 CrowdStatData[0].CrowdEventNum =2 CrowdStatData[0].CrowdList[0].Center=2 CrowdStatData[0].CrowdList[0].Radius=2 CrowdStatData[0].RegionEventNum =2 CrowdStatData[0].RegionList[0].Region[0][0]=10 CrowdStatData[0].RegionList[0].Region[0][1]=10</pre>

	CrowdStatData[0].RegionList[0].Region[1][0]=10 CrowdStatData[0].RegionList[0].Region[1][0]=100 ... CrowdStatData[0].RegionList[0].RegionID=0 CrowdStatData[0].RegionList[0].PeopleNum=100
Comment	Similar to above "Subscribe to Realtime Crowd Stat", but reply stat only once, not at regular time. <b>ChannelNo</b> : integer, video channel index which starts from 1.

## 8.5 Intelligent

### 8.5.1 Get Intelligent Caps

URL	http://<server>/cgi-bin/intelli.cgi?action=getCaps		
Method	GET		
Description	Get video analyse intelligent capabilities.		
[ Request Params ] ( key=value )			
capsName	string	O	Name of capability, it can be: AnalyseMode: get analyse mode capacity MultiChannelMode: get multi channel separate intelligent capacity and combined intelligent capacity MultiVideoAnalyse: get all intelligent capacity from the multichannel device in each channel if omit, means get all capacity
[ Response Params ] ( key=value )			
caps	object	R	The video service capabilities.
+AnalyseMode	string	O	Analyse mode, can be : RealStream, RecordFileStream, PicFileStream
+MultiChannelMode	object	O	Multi channel description
++IndepMode	Array<Array<object>>	O	Multi channel open at one time separately
+++Channel	int	R	Video channel number, start from 0
+++Type	string	R	Business class
++CompMode	Array<Array<object>>	O	Multi channel combined mode
+++Channel	int	R	Video channel number, start from 0
+++Type	string	R	Business class
+MultiVideoAnalyse	Array<object>	O	List of channel capabilities
++caps	object	R	Video analyse capabilities
+Algorithm	Array<object>	O	Algorithm version information
++Class	string	R	Intelligent analyse class, ex : ObjectDetect
++AlgorithmVersion	string	R	Algorithm version

++AlgorithmVendor	string	R	Algorithm Vendor name
+TotalCapacity	Array<object>	O	Intelligent analyse total capacity
++Class	string	R	Intelligent analyse class, ex : ObjectDetect
++Type	Array<string>	R	Intelligent analyse rules, ex : FaceDetection
++Number	int	R	The maximum number of video channels that can be analysed at same time

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/intelli.cgi?action=getCaps
Response	<pre> caps.AnalyseMode=RealStream caps.MultiChannelMode.IndepMode[0][0].Channel=0 caps.MultiChannelMode.IndepMode[0][0].Type=Normal caps.MultiChannelMode.IndepMode[0][1].Channel=1 caps.MultiChannelMode.IndepMode[0][1].Type=ObjectDetect ... caps.MultiChannelMode.CompMode[0][0].Channel=0 caps.MultiChannelMode.CompMode[0][0].Type=Normal caps.MultiChannelMode.CompMode[0][1].Channel=1 caps.MultiChannelMode.CompMode[0][1].Type=ObjectDetect ... caps.Algorithm[0].Class=Normal caps.Algorithm[0].AlgorithmVersion=V2.8 caps.Algorithm[0].AlgorithmVendor=Dahua caps.Algorithm[1].Class=ObjectDetect caps.Algorithm[1].AlgorithmVersion=V2.8 caps.Algorithm[1].AlgorithmVendor=Dahua ... caps.TotalCapacity[0].Class=Normal caps.TotalCapacity[0].Type[0]=FaceDetection caps.TotalCapacity[0].Number=3 caps.TotalCapacity[1].Class=ObjectDetect caps.TotalCapacity[1].Type[0]=FaceDetection caps.TotalCapacity[1].Number=3 ... </pre>

### 8.5.2 Subscribe Resource Usage Info

URL	http://<server>/cgi-bin/intelli.cgi?action=attachResource
Method	GET
Description	Subscribe the intelligent analyse resource
[ Request Params ] ( key=value format in URL )	

heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat meage are "Heartbeat". If this parameter is not present, its default value is 60.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
RemainCapacity	Array<object>	R	Remain capacity of intelligent analyse
+Class	string	R	Intelligent analyse class, ex : ObjectDetect
+Number	int	O	Remain number of video channels that can be analysed
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/intelli.cgi?action=attachResource&heartbeat=5		
Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  RemainCapacity[0].Class=Normal RemainCapacity[0].Number=1 RemainCapacity[1].Class=ObjectDetect RemainCapacity[1].Number=2  --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  RemainCapacity[0].Class=Normal RemainCapacity[0].Number=1 RemainCapacity[1].Class=ObjectDetect RemainCapacity[1].Number=1 ... </pre>		

## 8.6 WorkSuitCompareServer

### 8.6.1 Creating Compliance Library

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=createGroup		
Method	GET		
Description	Create compliance library		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
GroupName	string	R	Compliance library group name, up to 127 characters.
CutoutPolicy	uint32	R	Preferred solution 0: Full body 1: Upper body
GroupDetail	string	O	Remarks of compliance group, up to 255 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
GroupID	string	R	Compliance library group ID, up to 63 bytes.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=createGroup&GroupName=Test1&CutoutPolicy=0&GroupDetail=ForTest1		
Response	groupID=12		

### 8.6.2 Deleting Compliance Library

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=deleteGroup		
Method	GET		
Description	Delete compliance library		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
GroupID	string	R	Compliance library group ID, up to 63 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=deleteGroup&groupID=10000		
Response	OK		

### 8.6.3 Searching for Compliance Library

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=findGroup		
Method	GET		
Description	Searching for compliance library		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description

GroupID	string	O	Compliance library group ID, up to 63 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
GroupList	Array<object>	O	List of compliance group information found.
+GroupID	string	O	Compliance group ID, up to 63 characters.
+GroupName	string	O	Compliance group name, up to 127 characters.
+CutoutPolicy	int	O	Preferred solution 0: Full body 1: Upper body
+GroupDetail	string	O	Remarks of compliance group, up to 255 characters.
+GroupSize	integer	O	Number of workwears in the current group.
+Channels	Array<integer>	O	Video channel No., starting from 0.
+Similarity	Array<integer>	O	Similarity threshold; value range: 0–100. Binding video channel No.
+GroupType	string	O	Group type.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=findGroup		
Response	GroupList[0].GroupID=00001 GroupList[0].GroupName=Test1 GroupList[0].CutoutPolicy=0 GroupList[0].GroupDetail=ForTest1 GroupList[0].GroupSize=30 GroupList[0].Channels[0]=1 GroupList[0].Channels[1]=2 GroupList[0].Similarity[0]=80 GroupList[0].Similarity[1]=80 GroupList[0].GroupType=BlockListDB		

## 8.6.4 Getting Compliance Library Arming Information of Channels

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=getGroup		
Method	GET		
Description	Get compliance library arming information of channels Note: If the video channel has not been armed with any compliance library, the response will be returned successfully while body is null.		
[ Request Params ] (key=value format at URL)			
Name	Type	R/O	Param Description
Channel	int	R	Video channel number. Starting from 1.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
GroupList	Array<object>	O	List of armed group information.

+GroupID	string	O	Compliance group ID, up to 63 characters.
+Similarity	int	O	Similarity threshold; value range: 0–100.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=getGroup		
Response	GroupList[0].GroupID=00001 GroupList[0].Similarity=80		

## 8.6.5 Modifying Compliance Group Information

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=modifyGroup		
Method	GET		
Description	Modify compliance group information		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
GroupID	string	R	Compliance group ID, up to 63 characters.
GroupName	string	O	Compliance library group name, up to 127 characters.
GroupDetail	string	O	Remarks of compliance group, up to 255 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=modifyGroup&GroupName=Test1&GroupDetail=ForTest1		
Response	OK		

## 8.6.6 Arming Compliance Library by Video Channel

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=setGroup		
Method	GET		
Description	Arm some compliance libraries to the specified video channels. If the compliance libraries have been armed, update the similarity. Note: This method is the overall overwriting of arming information.		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
Channel	int	R	Video channel number
GroupList	Array<object>	O	Arming information list: If the list does not exist, the arming of all compliance libraries for the channels are deleted.
+GroupID	string	O	Compliance group ID, up to 63 characters.
+Similarity	int	O	Similarity threshold; value range: 0–100.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=setGroup&Channel=0&GroupList[0].GroupID=10000&GroupList[0].Similarity=80		
Response	OK		

## 8.6.7 Searching for Workwear Information in Compliance

### Library

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=startFind		
Method	GET		
Description	Start to search for workwear information in the compliance library. Note: If doFind is not executed within 60 s, the returned token will expire.		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
Condition	Object	R	Search condition.
+GroupID	Array<string>	R	Array list of compliance library IDs. Each compliance library ID is up to 63 characters.
WorkSuit	object	O	Search condition of workwear information
+FeatureState	int	O	Feature value status: 0: Unknown; 1: Extraction failed; 2: Extracted successfully; 3: Modeled but algorithm upgrade causes data unavailability and remodeling is required.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
Token	uint	R	Token for search task, which is used to get search results and stop the search.
TotalCount	int	R	Total number of results returned.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=startFind&Condition.GroupID[0]=10000&Condition.GroupID[1]=10003&WorkSuit.FeatureState=1		
Response	Token=123456789 TotalCount=24		

## 8.6.8 Getting Search Result

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=doFind		
Method	GET		
Description	Get a piece of workwear information from the search results. Note: If doFind is not executed within 60 s, the returned token will expire.		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Token for search task, which is used to get search results and stop the search.
BeginNumber	int	R	The starting position of the search result set should be between 0 and totalCount -1.
Count	int	R	The number of the search result.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
Found	int	R	The actual number returned by this search.
WorkSuitList	array<object>	R	List of workwear information found.

+UID	string	R	Workwear identifier, up to 31 characters.
+GroupID	string	R	Compliance library ID, up to 63 characters.
+FeatureState	int	O	Feature value status: 0: Unknown; 1: Extraction failed; 2: Extracted successfully; 3: Modeled but algorithm upgrade causes data unavailability and remodeling is required.
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=doFind&Token=123456789&BeginNumber=0&Count=1		
Response	Found=1 WorkSuitList[0].UID=0005 WorkSuitList[0].GroupID=10000 WorkSuitList[0].FeatureState=0		

## 8.6.9 Stopping Search

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=stopFind		
Method	GET		
Description	Stop search		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
Token	uint	R	Token for search task, which is used to get search results and stop the search.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=stopFind&Token=123456789		
Response	OK		
Response	OK		

## 8.6.10 Deleting Workwear Information

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=deleteByUID		
Method	GET		
Description	Delete workwear information		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
UID	Array<string>	R	Workwear identifier list, up to 31 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
ErrCode	Array<uint>	R	Error code list, indicating error messages when deleting information. 0: Success. 1: The workwear does not exist. 2: Database operation failed.

[ Example ]	
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=deletebyUID&UID[0]=10&UID[1]=11
Response	ErrCode[0]=0 ErrCode[1]=0

## 8.6.11 Re-extracting Features by Workwear

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=reAbstract		
Method	GET		
Description	Extract workwear features again. The re-extraction progress is reported through the "WorkSuitFeatureAbstract" event.		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
UID	Array<string>	O	Workwear identifier list; each UID is up to 31 characters.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=reAbstract&uid[0]=001&uid[1]=002		

## 8.6.12 Stopping Re-extracting Workwear Features

Syntax	http://<server>/cgi-bin/WorkSuitCompareServer.cgi?action=stopReAbstract		
Method	GET		
Description	Stop re-extracting workwear features		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/WorkSuitCompareServer.cgi?action=stopReAbstract		

## 8.7 SMD Query

### 8.7.1 Start SMD Query

Request URL	http://<server>/cgi-bin/api/SmdDataFinder.cgi?action=startFind			
Method	POST			
Request Params (JSON format in body)				
Name	Type	R/O	Description	Example

Condition	object	R	Search conditions	
+Channel	int32	O	Channel number. -1 means searching for all channels	0
+Channels	int[]	O	Channel number array (starting from 0). There should be at least one field, Channel or Channels. Generally, If both Channel and Channels fields exist, Channels should prevail.	
+SmdType	char[8][32]	O	Video type. Array "smdTypeHuman": Human. "smdTypeVehicle": Vehicle. "smdTypeHumanAndVehicle": Human and vehicle.	["smdTypeHuman", "smdTypeVehicle", "smdTypeHumanAndVehicle"]
+StartTime	char[32]	O	Start time	"2017-08-01 00:00:00"
+EndTime	char[32]	O	End time	"2017-08-02 00:00:00"
+Order	char[16]	O	Search for ordering method. "ascOrder": Ascending order (default value). "descOrder": Descending order.	"ascOrder"

#### Request Example

```
{
  "Condition": {
    "Channel": 0,
    "Channels": [0,1,2,5],
    "SmdType": ["smdTypeHuman", "smdTypeVehicle", "smdTypeHumanAndVehicle" ],
    "StartTime": "2017-08-01 00:00:00",
    "EndTime": "2017-08-02 00:00:00",
    "Order": "ascOrder"
  }
}
```

Response Params ( JSON format in body )				
Name	Type	R/O	Description	Example
Token	int	O	Return searched ID	1
Count	int	O	Total number of	100

			searched entries	
<b>Response Example</b>				
<pre>{   "Token": 1,   "Count": 100 }</pre>				

## 8.7.2 Searching for SMD Data

<b>Request URL</b>	http://<server>/cgi-bin/api/SmdDataFinder/doFind			
<b>Method</b>	POST			
<b>Request Params</b> (JSON format in body)				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Token	int	R	Search ID	1
Offset	int32	R	Location offset	0
Count	int	R	Number of searched items	100
<b>Request Example</b>				
<pre>{   "Token": 1,   "Offset": 0,   "Count": 100 }</pre>				

<b>Response Params</b> ( JSON format in body )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
SmdInfo	object[]	R	It is an array. Each element represents the historical data of a record data at a certain time.	
+Channel	int32	O	Channel number, starting from 0	1
+EndTime	char[32]	O	End time. "201	"2017-08-02 00:00:00"
+Type	char[32]	O	Video type. "smdTypeHuman": Human. "smdTypeVehicle": Vehicle. "smdTypeHumanAndVehicle": Human and vehicle.	"smdTypeHuman"
+StartTime	char[32]	O	Start time	2017-01-01 10:00:
<b>Response Example</b>				
<pre>{   "SmdInfo": [{     "Channel": 1,     "StartTime": "2017-08-01 00:00:00",     "EndTime": "2017-08-02 00:00:00", </pre>				

```

    "Type": "smdTypeHuman"
  },...{}]
}

```

### 8.7.3 End SMD Search

<b>Request URL</b>	http://<server>/cgi-bin/api/SmdDataFinder/stopFind			
<b>Method</b>	POST			
<b>Request Params (JSON format in body)</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Token	int	O	Search ID	1
<b>Request Example</b>				
<pre> {   "Token": 1 } </pre>				

<b>Response Params ( JSON format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
<b>Response Example</b>				
<pre> {} </pre>				

# 9

## Intelligent Traffic APIs

### 9.1 Traffic Snap

#### 9.1.1 Get the Specific Parking Space Status

Table 9-1

Syntax	http://<server>/cgi-bin/trafficSnap.cgi?action=getParkingSpaceStatus&channel=< <b>ChannelNo</b> >&< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Get specific parking space(s) status.
Example	http://192.168.1.108/cgi-bin/trafficSnap.cgi?action=getParkingSpaceStatus&condition.Lane[0]=0&condition.Lane[1]=255
Success Return	A list of parking space status status[0].Lane=0 status[0].PictureId=5 status[0].TrafficCar.CountInGroup=1 ... status[1].Lane=1 status[1].PictureId=4 status[1].TrafficCar.CountInGroup=1 ...
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>paramName</b> and <b>paramValue</b> : detail in table below.  In table below, <b>index</b> : The index of type array, start from 0  Parameters in Response : <b>TrafficCar</b> : the members refer to TrafficCar

#### Appendix

ParamName	ParamValue type	Description
condition.Lane[ <i>index</i> ]	int	The Lane value
condition.ResponseLevel	int	The Level value , refer to condition

#### 9.1.2 Open Strobe

Table 9-2

Syntax	http://<server>/cgi-bin/trafficSnap.cgi?action=openStrobe&channel=< <b>ChannelNo</b> >&inf
--------	--

	o.openType=< <b>OpenType</b> >&info.plateNumber=< <b>PlateNumber</b> >
Method	GET
Description	Open the strobe manually.
Example	http://192.168.1.108/cgi-bin/trafficSnap.cgi?action=openStrobe&channel=1&info.openType=Normal&info.plateNumber=046XRW
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>OpenType</b> : string, For now, the value is fixed to "Normal" <b>PlateNumber</b> : string, the plateNumber of the TrafficCar.

### 9.1.3 Open/Close Unlicensed Vehicle Detection

Table 9-3

Syntax	http://<server>/cgi-bin/trafficSnap.cgi?action=< <b>Action</b> >&channel=< <b>ChannelNo</b> >&name=UnlicensedVehicle
Method	GET
Description	Open or Close the unlicensed vehicle detection.
Example	http://192.168.1.108/cgi-bin/trafficSnap.cgi?action=open&channel=1&name=UnlicensedVehicle
Success Return	OK
Comment	Parameters in URL: <b>Action</b> : string, this value can be open and close. <b>ChannelNo</b> : integer, video channel index which starts from 1.

### 9.1.4 Snap

Table 9-4

Syntax	http://<server>/cgi-bin/trafficSnap.cgi?action=manSnap&channel=< <b>ChannelNo</b> >
Method	GET
Description	Take a snapshot manually. For intelligent traffic device, it should use this method to take a snapshot. But, the response is not image data.If you want to get the image data, please follow these steps: 1. Use the method mentioned chapter (4.4.3 Subscribe to snapshot) to subscribe the image data, and the eventcode is "TrafficManualSnap". 2. Use the "manSnap"to take a snapshot manually. 3. In the connection which built in the Step 1, the device will send the image data.
Example	http://192.168.1.108/cgi-bin/trafficSnap.cgi?action=manSnap&channel=1
Success Return	OK
Comment	Parameters in URL:

**ChannelNo:** integer, video channel index which starts from 1.

## 9.2 Traffic Parking

### 9.2.1 Get All Status of Parking Spaces

Table 9-5

Syntax	http://<server>/cgi-bin/trafficParking.cgi?action=getAllParkingSpaceStatus
Method	GET
Description	Get all valid parking spaces status of one device.
Example	http://192.168.1.108/cgi-bin/trafficParking.cgi?action=getAllParkingSpaceStatus
Success Return	<pre>status[0].Lane=0 status[0].CustomParkNo = A2701 status[0].Status = Park status[0].SpaceType = 0 ... status[1].Lane=1 status[1].Status = NoPark ... sceneType = ParkingStatistics statisticsMode = AreaMode areaStatus[0].Name = A01 areaStatus[0].Count = 10 areaStatus[0].SpaceType = 0 areaStatus[0].RemainCnt = 5 ... areaStatus[1].Name = A02 areaStauts[1].Count = 20 ...</pre>
Comment	<p>Parameters in Response :</p> <p><b>Status</b> : Park or NoPark</p> <p><b>CustomParkNo</b>: customized parking space number</p> <p><b>SpaceType</b>: 0: trolley parking space, 1: Crane parking space</p> <p><b>sceneType</b> : scene type; "parkingspace": parking space detection type; and "parkingstatistics": parking space statistics type</p> <p><b>statisticsMode</b>: Statistical mode, valid when the sceneType is "parkingstatistics".</p> <p>"areamode": area mode, "spacemode": parking space mode</p> <p><b>areaStatus</b>: area status. It is an array. It is valid when scenetype is "parkingstatistics" and statisticsmode is "areamode"</p> <p><b>SpaceType</b>: parking space type: 0: trolley parking space, 1: Crane parking space</p>

### 9.2.2 Parking Space Light State

- Get parking space light state

Table 9-6

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=ParkingSpaceLightState
Method	GET
Description	Use this method. It can get the light state config. For example, it can know that when the space is free, then the light should be green, and the space is full, the light should be red.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=ParkingSpaceLightState
Success Return	<pre> table.ParkingSpaceLightState.SpaceFree.Blue=0 table.ParkingSpaceLightState.SpaceFree.Green=1 table.ParkingSpaceLightState.SpaceFree.Pink=0 table.ParkingSpaceLightState.SpaceFree.Purple=0 table.ParkingSpaceLightState.SpaceFree.Red=0 table.ParkingSpaceLightState.SpaceFree.White=0 table.ParkingSpaceLightState.SpaceFree.Yellow=0 table.ParkingSpaceLightState.SpaceFull.Blue=0 table.ParkingSpaceLightState.SpaceFull.Green=0 table.ParkingSpaceLightState.SpaceFull.Pink=0 table.ParkingSpaceLightState.SpaceFull.Purple=0 table.ParkingSpaceLightState.SpaceFull.Red=1 table.ParkingSpaceLightState.SpaceFull.White=0 table.ParkingSpaceLightState.SpaceFull.Yellow=0 table.ParkingSpaceLightState.SpaceOrder.Blue=0 table.ParkingSpaceLightState.SpaceOrder.Green=0 table.ParkingSpaceLightState.SpaceOrder.Pink=0 table.ParkingSpaceLightState.SpaceOrder.Purple=0 table.ParkingSpaceLightState.SpaceOrder.Red=0 table.ParkingSpaceLightState.SpaceOrder.White=0 table.ParkingSpaceLightState.SpaceOrder.Yellow=1 table.ParkingSpaceLightState.SpaceOverLine.Blue=0 table.ParkingSpaceLightState.SpaceOverLine.Green=0 table.ParkingSpaceLightState.SpaceOverLine.Pink=0 table.ParkingSpaceLightState.SpaceOverLine.Purple=0 table.ParkingSpaceLightState.SpaceOverLine.Red=0 table.ParkingSpaceLightState.SpaceOverLine.White=0 table.ParkingSpaceLightState.SpaceOverLine.Yellow=1 table.ParkingSpaceLightState.SpaceSpecial.Blue=0 table.ParkingSpaceLightState.SpaceSpecial.Green=0 table.ParkingSpaceLightState.SpaceSpecial.Pink=0 table.ParkingSpaceLightState.SpaceSpecial.Purple=0 table.ParkingSpaceLightState.SpaceSpecial.Red=0 table.ParkingSpaceLightState.SpaceSpecial.White=0 table.ParkingSpaceLightState.SpaceSpecial.Yellow=1 </pre>
Comment	<p>Parameters in Response :</p> <p>There are five conditions about the space light state: SpaceFree, SpaceFull,</p>

	<p>SpaceOrder, SpaceOverLine, and SpaceSpecial. For each condition, you can choose the light state.</p> <p>The state value is integer. It can be 0, 1, 2.</p> <p>0 means close;</p> <p>1 means open;</p> <p>2 means twinkle.</p>
--	--

- Set parking space light state

Table 9-7

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set the light state config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&ParkingSpaceLightState.SpaceFree.Blue=1&ParkingSpaceLightState.SpaceFree.Green=0&ParkingSpaceLightState.SpaceFree.Pink=0&ParkingSpaceLightState.SpaceFree.Purple=0&ParkingSpaceLightState.SpaceFree.Red=0&ParkingSpaceLightState.SpaceFree.White=0&ParkingSpaceLightState.SpaceFree.Yellow=0
Success Return	OK
Comment	—

### 9.2.3 Set Order State

Table 9-8

Syntax	http://<server>/cgi-bin/trafficParking.cgi?action=setOrderState&state[0].Lane=<LaneNumber>&state[0].State=<State>
Method	GET
Description	Set the light order state, order or not
Example	http://192.168.1.108/cgi-bin/trafficParking.cgi?action=setOrderState&state[0].Lane=0&state[0].State=Ordered
Success Return	OK
Comment	<p>Parameters in URL:</p> <p><b>LaneNumber</b>: integer, for now, fixed to 0.</p> <p><b>State</b>: string, the value can be "Ordered" or "Free".</p>

### 9.2.4 Set Light State

Table 9-9

Syntax	http://<server>/cgi-bin/trafficParking.cgi?action=setLightState&state[<Index>].LightNo=<LaneNumber>&state[<Index>].Color=<Color>&state[<Index>].State=<State>&state[<Index>].Enable=<Enable>
Method	GET

Description	Set the light state.
Example	http://192.168.1.108/cgi-bin/trafficParking.cgi?action=setLightState&state[0].LightNo=0&state[0].Color=Red&state[0].State=0&state[0].Enable=true
Success Return	OK
Comment	Parameters in URL: <b>Index</b> : integer, start with 0. <b>LaneNumber</b> : integer, the No. of the Lane. <b>Color</b> : string, it can be Red, Yellow, Blue, Green, Purple, White, Pink. <b>State</b> : integer, it can be 0, 1, 2.0 means close;1 means open;2 means twinkle; <b>Enable</b> : true or false, enable or not.

## 9.2.5 [Config] Parking Space Access Filter Setting

- Get parking space access filter setting

Table 9-10

URL	http://<server>/cgi-bin/ConfigManager.cgi?action=getConfig&name=ParkingSpaceAccessFilter		
Method	GET		
Description	Get Parking Space Access Filter config. Using this method, we can get the accessible address of the device.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+ParkingSpaceAccessFilter	object	R	ParkingSpaceAccessFilter config object
++Enable	bool	R	Enable filter or not
++Type	string	R	Filter type, can be: " <b>BannedList</b> ", " <b>TrustList</b> ".
++TrustList	Array<string>	O	IP address list that trust
++BannedList	Array<string>	O	IP address list that banned.
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/ConfigManager.cgi?action=getConfig&name=ParkingSpaceAccessFilter		
Response	table.ParkingSpaceAccessFilter.Enable=false table.ParkingSpaceAccessFilter.Type=TrustList table.ParkingSpaceAccessFilter.TrustList[0]=172.24.2.14 table.ParkingSpaceAccessFilter.BannedList[0]=172.24.2.15		

- Set parking space access filter setting

Table 9-11

URL	http://<server>/cgi-bin/ConfigManager.cgi?action=setConfig
Method	GET

Description	Set Parking Space Access Filter config.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
ParkingSpaceAccessFilter	object	R	ParkingSpaceAccessFilter config object
+Enable	bool	R	Enable filter or not
+Type	string	R	Filter type, can be: " <b>BannedList</b> ", " <b>TrustList</b> ".
+TrustList	Array<string>	O	IP address list that trust
+BannedList	Array<string>	O	IP address list that banned.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/ConfigManager.cgi?action=setConfig&ParkingSpaceAccessFilter.Enable=true&ParkingSpaceAccessFilter.Type=TrustList&ParkingSpaceAccessFilter.TrustList[0]=172.24.2.14&ParkingSpaceAccessFilter.BannedList[0]=172.24.2.15		
Response	OK		

## 9.2.6 Set OverLine State

Table 9-12

Syntax	http://<server>/cgi-bin/trafficParking.cgi?action=setOverLineState&state[0].Lane=< <b>LaneNumber</b> >&state[0].State=< <b>State</b> >
Method	GET
Description	Set the spaceState overLine state or StopOverLine state
Example	http://192.168.1.108/cgi-bin/trafficParking.cgi?action=setOverLineState&state[0].Lane=0&state[0].State=OverLine
Success Return	OK
Comment	Parameters in URL: <b>LaneNumber</b> : integer, for now, fixed to 0. <b>State</b> : string, the value can be "OverLine" or "StopOverLine".

## 9.3 Traffic Record Import Export

### 9.3.1 Traffic BlackList / RedList

- Import blacklist or redlist record

Table 9-13

Syntax	http://<server>/cgi-bin/trafficRecord.cgi?action=uploadFile&Type=< <b>Type</b> >&format=< <b>Format</b> >&code=< <b>Code</b> >
Method	POST

Description	Upload a blacklist or redlist(whitelist) file into the device, and the device will import the data of the file.This operation may take a long time.
Example	<pre> POST /cgi-bin/trafficRecord.cgi?action=uploadFile&amp;Type=TrafficBlackList&amp;format=CSV&amp;code=UTF-8 HTTP/1.1 Host: 192.168.1.108 Connection: keep-alive Content-Length:XXXX Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryooT6JTCbuezAQeDy  -----WebKitFormBoundaryooT6JTCbuezAQeDy Content-Disposition: form-data; name="blackfile"; filename="TrafficBlackList.CSV" Content-Type: application/vnd.ms-excel File data....  -----WebKitFormBoundaryooT6JTCbuezAQeDy-- </pre>
Success Return	OK
Comment	Parameters in URL: <b>Type:</b> string, the value can be "TrafficBlackList" or "TrafficRedList". <b>Format:</b> string, for the blacklist or redlist(whitelist), the value is fixed to "CSV". <b>Code:</b> string, for the blacklist, the value can be "utf-8"or "GB2312".

- Export blacklist or redlist record async

Table 9-14

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=exportAsyncFile&name=<Name>&filename=<FileName>&format=<Format>&code=<Code>
Method	GET
Description	Let the device export the record into a file in the device. This method is non-blocking. To get the result, it should use the "getFileExportState" method.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=exportAsyncFile&name=TrafficBlackList&filename=RecordFile01&format=CSV&code=utf-8
Success Return	OK
Comment	Parameters in URL: <b>Name:</b> string, the value can be "TrafficBlackList" or "TrafficRedList". <b>FileName:</b> [optional], string, the name of the exported file. <b>Format:</b> string, for the blacklist or redlist(whitelist), the value is fixed to "CSV". <b>Code:</b> string, for the blacklist or whitelist, the value can be "utf-8" or "GB2312".

- Get export blacklist or redlist record state

Table 9-15

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=<Name>
Method	GET

Description	Get the blacklist or redlist(whitelist) export state
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=TrafficBlackList
Success Return	state=0
Comment	Parameters in URL: <b>Name</b> : string, the value can be "TrafficBlackList" or "TrafficRedList". <b>state</b> : integer, the value can be 0(success),1(fail), 2(running), 3(the file is invalid), 4(the file is too large) or 5(have duplicate info in the file).

- Get export blacklist or redlist record result

Table 9-16

Syntax	http://<server>/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=< <b>Type</b> >&filename=< <b>FileName</b> >
Method	GET
Description	Download the blacklist or redlist(whitelist) record
Example	http://192.168.1.108/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=TrafficBlackList&filename=RecordFile01
Success Return	File data
Comment	Parameters in URL: <b>Type</b> : string, the value can be "TrafficBlackList" or "TrafficRedList". <b>FileName</b> : [optional], string, the name of the downloaded file. The FileName must be same as the exported file name.

### 9.3.2 Traffic Flow

- Export traffic flow record

Table 9-17

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=exportAsyncFile&name=TrafficFlow&filename=< <b>FileName</b> >&format=< <b>Format</b> >&code=< <b>Code</b> >
Method	GET
Description	Let the device export the record into a file in the device. This method is non-blocking. To get the result, it should use the "getFileExportState" method.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=exportAsyncFile&name=TrafficFlow&filename=RecordFile01&format=CSV&code=utf-8
Success Return	OK
Comment	Parameters in URL: <b>FileName</b> : [optional], string, the name of the exported file. <b>Format</b> : string, for the TrafficFlow, the value is fixed to "CSV". <b>Code</b> : string, for the TrafficFlow, the value can be "utf-8" or "GB2312".

Table 9-18

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=TrafficFlow
Method	GET
Description	Get the export state.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=TrafficFlow
Success Return	state=0
Comment	Parameters in Response: <b>state</b> : integer, the value can be 0(success), 1(fail), 2(running), 3(the file is invalid), 4(the file is too large) or 5(have duplicate info in the file).

Table 9-19

Syntax	http://<server>/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=TrafficFlow&filename=<FileName>
Method	GET
Description	Download the traffic flow record.
Example	http://192.168.1.108/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=TrafficFlow&filename=RecordFile01
Success Return	File data
Comment	Parameters in URL: <b>FileName</b> : [optional], string, the name of the downloaded file. The FileName must be same as the exported file name.

### 9.3.3 Traffic Snap Event Info

- Export traffic snap event info record

Table 9-20

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=exportAsyncFileByCondition&name=TrafficSnapEventInfo&filename=<FileName>&format=<Format>&code=<Code>&condition.startTime=<StartTime>&condition.endTime=<EndTime>
Method	GET
Description	Let the device export the record into a file in the device. This method is non-blocking. To get the result, it should use the "getFileExportState" method.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=exportAsyncFileByCondition&name=TrafficSnapEventInfo&filename=RecordFile01&format=CSV&code=utf-8&condition.startTime=123456700&condition.endTime=123456800
Success Return	OK
Comment	Parameters in URL: <b>FileName</b> : [optional], string, the name of the exported file. <b>Format</b> : string, for the TrafficSnapEventInfo, the value is fixed to "CSV". <b>Code</b> : string, for the TrafficSnapEventInfo, the value can be "utf-8" or "GB2312".

	<p><b>StartTime:</b> string, the start of the record's Time.</p> <p><b>EndTime:</b> string, The end of the record's Time.</p>
--	---

Table 9-21

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=TrafficSnapEventInfo
Method	GET
Description	Get the export state.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=getFileExportState&name=TrafficSnapEventInfo
Success Return	state=0
Comment	<p>Parameters in Response:</p> <p><b>state:</b> integer, the value can be 0(success), 1(fail), 2(running), 3(the file is invalid), 4(the file is too large) or 5(have duplicate info in the file).</p>

Table 9-22

Syntax	http://<server>/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=TrafficSnapEventInfo&filename=<FileName>
Method	GET
Description	Download the traffic snap event info record.
Example	http://192.168.1.108/cgi-bin/trafficRecord.cgi?action=downloadFile&Type=TrafficSnapEventInfo&filename=RecordFile01
Success Return	File data
Comment	<p>Parameters in URL:</p> <p><b>FileName:</b> [optional], string, the name of the downloaded file. The FileName must be same as the exported file name.</p>

## 9.4 Vehicles Distribution

### 9.4.1 Subscribe Vehicles Distribution Data

Table 9-23

URL	http://<server>/cgi-bin/vehiclesDistribution.cgi?action=attach		
Method	GET		
Description	Subscribe the vehicles Distribution data.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Channel	int	R	Video channel index which starts from 1

heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat meage are "Heartbeat". If this parameter is not present, its default value is 60.
[ Response Params ] ( key=value format)			
Name	Type	R/O	Param Description
Channel	int	R	Video channel index which starts from 1
VehiclesData	array<object>	R	Vehicles distribution data info.
+PtzPresetId	int	R	The ptz preset index which starts from 1.
+RuleId	int	R	The rule id.
+RuleType	int	R	The rule type, 197 : vehicle congestion detection, 198 : vehicle limit detection.
+VehiclesNum	int	R	The vehicles number.
+QueueLen	int	R	The vehicle queue length.
+Region	array<array<int>>	R	The detection region, the first array is point list, max item is 32, the second array is point, must be two int, means x and y value, coordinate remap to 0 - 8192.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/vehiclesDistribution.cgi?action=attach&Channel=1&heartbeat=5		
Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Channel=1 VehiclesData[0].PtzPresetId=1 VehiclesData[0].RuleId=1 VehiclesData[0].RuleType=197 VehiclesData[0].VehiclesNum=25 VehiclesData[0].QueueLen=53 VehiclesData[0].Region[0][0]=1032 VehiclesData[0].Region[0][1]=1035 VehiclesData[0].Region[1][0]=1045 VehiclesData[0].Region[1][1]=5072 VehiclesData[0].Region[2][0]=6163 VehiclesData[0].Region[2][1]=5127 VehiclesData[0].Region[3][0]=6031 VehiclesData[0].Region[3][1]=1063  --&lt;boundary&gt; </pre>		

<pre>Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Channel=1 VehiclesData[0].PtzPresetId=2 ...</pre>
--

DAHUA HTTP API V3.05 for Whisper House Kft.

44638

2021-12-21

# 10

## Thermography and Radiometry APIs

### 10.1 Thermography Manager

#### 10.1.1 Get Capability of Thermography

Table 10-1

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=getCaps&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get thermography capability.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=getCaps&channel=1
Success Return	caps. <b>PresetModes</b> = Indoor caps.Brightness.Max = 100 caps.Brightness.Min = 0 caps.Brightness.Step = 1 caps.Sharpness.Max= 100 caps.Sharpness.Min = 0 caps.Sharpness.Step = 5 caps.EZoom.Max= 24 caps.EZoom.Min = 0 caps.EZoom.Step = 1 caps.ThermographyGamma.Max= 8 caps.ThermographyGamma.Min = -8 caps.ThermographyGamma.Step = 1 caps.SmartOptimizer.Max= 100 caps.SmartOptimizer.Min = 0 caps.SmartOptimizer.Step = 5 caps.Agc.Max= 255 caps.Agc.Min = 0 caps.Agc.Step = 5 caps.AgcMaxGain.Max= 255 caps.AgcMaxGain.Min = 0 caps.AgcMaxGain.Step = 5 caps.AgcPlateau.Max= 100 caps.AgcPlateau.Min = 0 caps.AgcPlateau.Step = 5 caps. <b>PresetColorization</b> [i]= Ironbow2 caps. <b>PresetROIModes</b> [j]= Full Screen
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

	<p>Parameters in Response:</p> <p><b>PresetModes:</b> the preset mode. Range is { "Indoor", "Outdoor", "Default" }</p> <p><b>PresetColorization:</b> Preset colorization mode. Range is { "WhiteHot", "BlackHot", "Fusion", "Rainbow", "Globow", "Ironbow1", "Ironbow2", "Sepia", "Color1", "Color2", "Icefire", "Rain", "RedHot", "GreenHot"}.</p> <p><b>PresetROIModes:</b> Preset ROI mode. Range is {"Full Screen", "Sky", "Ground", "Horizon", "Center 75%", "Center 50%", "Center 25%", "Custom"}</p>
--	---

## 10.1.2 Thermography Options

- Get thermography options config

Table 10-2

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=ThermographyOptions
Method	GET
Description	Thermography options contain EZoom, Colorization, and SmartOptimizer and so on.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=ThermographyOptions
Success Return	<pre> head.EZoom=0 head.Colorization=White Hot head.SmartOptimizer=10 head.OptimizedRegion.Type=Custom head.OptimizedRegion.Enable= true head.OptimizedRegion.Regions[1][0u]=0 head.OptimizedRegion.Regions[1][1u]=0 head.OptimizedRegion.Regions[1][2u]=0 head.OptimizedRegion.Regions[1][3u]=0 head.Agc=10 head.AgcMaxGain=10 head.AgcPlateau=10 head.Mode="HighTemperature" head.Auto.LowToHigh=13 head.Auto.LHROI=15 head.Auto.HighToLow=12 head.Auto.HLROI=95 </pre>
Comment	<p>Parameters in Response:</p> <p><b>head</b> = table.ThermographyOptions [<b>ChannelNo</b>][0]</p> <p><b>ChannelNo:</b> integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>Regions:</b> the region is a rectangle</p> <p><b>i:</b> the array index starts from 0.</p>

- Set thermography options config

Table 10-3

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramV
--------	---

	<b>alue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set thermography options.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&ThermographyOptions[0][0].OptimizedRegion.Type=Gound
Success Return	OK
Comment	<p>Parameters in URL: The paramName and paramValue are in the table below.</p> <p>In table below, <b>head</b> = ThermographyOptions[<b>ChannelNo</b>][0] <b>ChannelNo</b>: integer, array index starts from 0, which means video channel(equals to video channel index -1, and so 0 means channel 1). <b>i</b>: the array index starts from 0.</p>

#### Appendix

ParamName	ParamValue type	Description
<b>head</b> .EZoom	integer	Range is [0—24]. Range and step are got from interface in "10.1.1 Get Capability of Thermography".
<b>head</b> .Colorization	String	Range is {"White Hot", "Black Hot", "Ironbow2", "IceFire"...}. Range and step are got from interface in "10.1.1 Get Capability of Thermography".
<b>head</b> .SmartOptimizer	integer	Range is [0—100]. Range and step are got from interface in "10.1.1 Get Capability of Thermography".
<b>head</b> .OptimizedRegion.Type	String	Range is {"Full Screen", "Sky", "Ground", "Horizontal", "Center 75%", "Center 50%", "Center 25%", "Custom"}.
<b>head</b> .OptimizedRegion.Enable	bool	true: enable false: not enable
<b>head</b> .OptimizedRegion.Regions[ <i>i</i> ][0u]	integer	Range is [0—8191]. <b>i</b> : the region index, starts from 0.
<b>head</b> .OptimizedRegion.Regions[ <i>i</i> ][1u]	integer	Range is [0—8191]. <b>i</b> : the region index, starts from 0.
<b>head</b> .OptimizedRegion.Regions[ <i>i</i> ][2u]	integer	Range is [0—8191]. <b>i</b> : the region index, starts from 0.
<b>head</b> .OptimizedRegion.Regions[ <i>i</i> ][3u]	integer	Range is [0—8191]. <b>i</b> : the region index, starts from 0.
<b>head</b> .Agc	integer	Range is [0—255]. Range and step are got from interface in "10.1.1 Get Capability of Thermography".
<b>head</b> .AgcMaxGain	integer	Range is [0—255]. Range and step are got from interface in "10.1.1

ParamName	ParamValue type	Description
		Get Capability of Thermography".
<b>head</b> .AgcPlateau	integer	Range and step are got from interface in "10.1.1 Get Capability of Thermography".
<b>head</b> .Mode	string	Range is {"HighTemperature", "LowTemperature", "Auto"}.
<b>head</b> .Auto.LowToHigh	integer	UInt32
<b>head</b> .Auto.LHROI	integer	UInt32, percentage range is[0—100]
<b>head</b> .Auto.HighToLow	integer	UInt32
<b>head</b> .Auto.HLROI	integer	UInt32, percentage range is[0—100]

### 10.1.3 Get ExternSystem Information

Table 10-4

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=getExternSystemInfo&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get extern system Info.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=getExternSystemInfo&channel=1
Success Return	sysInfo.SerialNumber = 1111111123 sysInfo.SoftwareVersion = 222222222222 sysInfo.FirmwareVersion= 333333333333 sysInfo.LibVersion = 4444444444
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

### 10.1.4 Get Information of Preset Mode

Table 10-5

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=getPresetParam&channel=< <b>ChannelNo</b> >&mode=< <b>modeType</b> >
Method	GET
Description	Get preset mode info.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=getPresetParam&channel=1&mode=Default
Success Return	presetInfo.Brightness = 50 presetInfo.Sharpness= 50 presetInfo.EZoom= 12 presetInfo.ThermographyGamma= 0 presetInfo.Colorization= "White Hot" presetInfo.SmartOptimizer= 10 presetInfo.OptimizedRegion.Type= Full Screen presetInfo.OptimizedRegion.Enable= Full Screen

	<pre> presetInfo.OptimizedRegion.<b>Regions</b>[i][0u]=0 presetInfo.OptimizedRegion.<b>Regions</b>[i][1u]=0 presetInfo.OptimizedRegion.<b>Regions</b>[i][2u]=0 presetInfo.OptimizedRegion.<b>Regions</b>[i][3u]=0 presetInfo.Agc= 10 presetInfo.AgcMaxGain=10 presetInfo.AgcPlateau = 10 </pre>
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p><b>modeType</b>: depends on capability , get from interface in getCaps</p> <p>Parameters in Response :</p> <p><b>Regions</b> : the region is a rectangle</p> <p><b>i</b> : the array index.</p>

## 10.1.5 Get Optimized Region Information

Table 10-6

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=getOptimizedRegion&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get optimized region info.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=getOptimizedRegion&channel=1
Success Return	<pre> optimizedRegion.Type= Full Screen optimizedRegion.Enable= true optimizedRegion.<b>Regions</b>[i][0u]=0 optimizedRegion.<b>Regions</b>[i][1u]=0 optimizedRegion.<b>Regions</b>[i][2u]=0 optimizedRegion.<b>Regions</b>[i][3u]=0 </pre>
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p>Parameters in Response:</p> <p><b>Regions</b> : the region is a rectangle</p> <p><b>i</b>: the region index.</p>

## 10.1.6 Enable Shutter

Table 10-7

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=enableShutter&channel=< <b>ChannelNo</b> >&enable=< <b>Enable</b> >
Method	GET
Description	Shutter control, whether enable shutter.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=enableShutter&channel

	=1&enable=true
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>Enable</b> : true or false, enable or not.

## 10.1.7 Fix Focus

Table 10-8

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=fixFocus&linkVideoChannel[0]=< <b>ChannelNo</b> >&linkVideoChannel[1]=< <b>ChannelNo</b> >[&speed=< <b>SpeedValue</b> >]
Method	GET
Description	The visual channel change focus to the same as the thermography channel.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=fixFocus&linkVideoChannel[0]=1&linkVideoChannel[1]=2
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>SpeedValue</b> : float, range is 0.0-1.0.

## 10.1.8 Do Flat Field Correction

Table 10-9

Syntax	http://<server>/cgi-bin/ThermographyManager.cgi?action=doFFC&channel=< <b>ChannelNo</b> >
Method	GET
Description	Do flat field correction.
Example	http://192.168.1.108/cgi-bin/ThermographyManager.cgi?action=doFFC&channel=1
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

## 10.2 Radiometry

### 10.2.1 Get Capability of Radiometry

Table 10-10

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=getCaps[&channel=< <b>ChannelNo</b> >]
Method	GET

Description	Get the capabilities of radiometry manager.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=getCaps&channel=1
Success Return	caps.TotalNum.MaxNum=8 caps.TotalNum.Spot.MaxSpots=8 caps.TotalNum.Line.MaxLines=1 caps.TotalNum.Area.MaxAreas=8 caps.TemperPresets.MaxPresets=256 caps.MeterInfo.Type[0u]=Spot caps.MeterInfo.Type[1u]=Area caps.MeterInfo.ObjectEmissivity.Max=100 caps.MeterInfo.ObjectEmissivity.Min=0 caps.MeterInfo.ObjectEmissivity.Default=0 caps.MeterInfo.ObjectEmissivity.Step=1 caps.MeterInfo.ObjectDistanceMeter.Max=100 caps.MeterInfo.ObjectDistanceMeter.Min=0 caps.MeterInfo.ObjectDistanceMeter.Default=0 caps.MeterInfo.ObjectDistanceMeter.Step=1 caps.MeterInfo.ReflectedTemperature.Max=100 caps.MeterInfo.ReflectedTemperature.Min=0 caps.MeterInfo.ReflectedTemperature.Default=0 caps.MeterInfo.ReflectedTemperature.Step=1 caps.MeterInfo.RelativeHumidity.Max=100 caps.MeterInfo.RelativeHumidity.Min=0 caps.MeterInfo.RelativeHumidity.Default=0 caps.MeterInfo.RelativeHumidity.Step=1 caps.MeterInfo.AtmosphericTemperature.Max=100 caps.MeterInfo.AtmosphericTemperature.Min=0 caps.MeterInfo.AtmosphericTemperature.Default=0 caps.MeterInfo.AtmosphericTemperature.Step=1 caps.Statistics.MinPeriod=60 caps.Isotherm.MaxTemp=327.0 caps.Isotherm.MinTemp=-20.0
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

## 10.2.2 Heat Image Thermometry

- Get heat image thermometry config

Table 10-11

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=HeatImagingThermometry
Method	GET
Description	Get HeatImagingThermometry Config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=HeatImagingThermometry

Success Return	table.RelativeHumidity = 50 table.AtmosphericTemperature =20 table.ObjectEmissivity =1 table.ObjectDistance =100 table.ReflectedTemperature=20 table.TemperatureUnit= Centigrade table.Isotherm.Enable=true table.Isotherm.MaxValue=50 table.Isotherm.MinValue=0 table.Isotherm.ColorBarDisplay=true table.HotSpotFollow=true table.TemperEnable=true
Comment	—

- Set heat image thermometry config

Table 10-12

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;paramName&gt;=&lt;paramValue&gt;[&amp;&lt;paramName&gt;=&lt;paramValue&gt;...]</code>
Method	GET
Description	Set HeatImagingThermometry Config
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;HeatImagingThermometry.RelativeHumidity=50&amp;HeatImagingThermometry.ObjectDistance=20.3</code>
Success Return	OK
Comment	Parameters in URL: The paramName and paramValue are in the table below.

## Appendix

ParamName	ParamValue type	Description
HeatImagingThermometry.RelativeHumidity	integer	The Relative Humidity range and step are got from interface in <b>getCaps</b> .
HeatImagingThermometry.AtmosphericTemperature	float	The Atmospheric Temperature range and step are got from interface in <b>getCaps</b> .
HeatImagingThermometry.ObjectEmissivity	float	The Object Emissivity range and step are got from interface in <b>getCaps</b> .
HeatImagingThermometry.ObjectDistance	float	The Object Distance range and step are got from interface in <b>getCaps</b> . Unit is meter.
HeatImagingThermometry.ReflectedTemperature	float	The Reflected Temperature range and step are got from interface in <b>getCaps</b> .

ParamName	ParamValue type	Description
HeatImagingThermometry.TemperatureUnit	string	Range is {Centigrade, Fahrenheit}.
HeatImagingThermometry.Isotherm.Enable	bool	true or false
HeatImagingThermometry.Isotherm.MaxValue	float	MaxValue range is got form interface in getCaps. MaxValue must be bigger than MinVaue
HeatImagingThermometry.Isotherm.MinValue	float	MinValue range is got form interface in getCaps. MinValue must be smaller than MaxVaue.
HeatImagingThermometry.Isotherm.ColorBarDisplay	bool	true or false
HeatImagingThermometry.HotSpotFollow	bool	true or false
HeatImagingThermometry.TemperEnable	bool	true or false

### 10.2.3 Thermometry Rule

- Get thermometry rule config

Table 10-13

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=ThermometryRule
Method	GET
Description	Get thermometry rule.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=ThermometryRule
Success Return	<pre> <b>head</b>.Enable = true <b>head</b>.PresetId =0 <b>head</b>.RuleId=0 <b>head</b>.Name=SpotName <b>head</b>.Type=Spot <b>head</b>.MeterRegion.Coordinates[<b>PointNo</b>][0]= 0 <b>head</b>.MeterRegion.Coordinates[<b>PointNo</b>][1]= 0 ... <b>head</b>.T=3 <b>head</b>.<b>Alarm</b>.Id=0 <b>head</b>.<b>Alarm</b>.Enable=true <b>head</b>.<b>Alarm</b>.Result =Max <b>head</b>.<b>Alarm</b>.AlarmCondition=Below <b>head</b>.<b>Alarm</b>.Threshold=20.0 <b>head</b>.<b>Alarm</b>.PreThreshold=10.0 <b>head</b>.<b>Alarm</b>.PreDuration=30 <b>head</b>.<b>Alarm</b>.Hysteresis=0.1 <b>head</b>.<b>Alarm</b>.Duration=30 <b>head</b>.LocalParameters.Enable=true <b>head</b>.LocalParameters.ObjectEmissivity=0.95 </pre>

	<b>head.LocalParameters.ObjectDistance=0.95</b> <b>head.LocalParameters.RefalectedTemp=0</b>
Comment	Parameters in Response : <b>head</b> =table.ThermometryRule[ <b>ChannelNo</b> ][ <b>RuleNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>RuleNo</b> =rule index. <b>PointNo</b> = point index <b>Alarm</b> = AlarmSetting[ <b>AlarmNo</b> ] <b>AlarmNo</b> = alarm index

- Set thermometry rule config

Table 10-14

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set thermometry rule.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&ThermometryRule[0][0].Name=name1
Success Return	OK
Comment	Parameters in URL: The paramName and paramValue are in the table below.  In table below, <b>head</b> = ThermometryRule[ <b>ChannelNo</b> ][ <b>RuleNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1). <b>PointNo</b> = point index <b>RuleNo</b> =rule index. <b>Alarm</b> = AlarmSetting[ <b>AlarmNo</b> ] <b>AlarmNo</b> = alarm index

## Appendix

ParamName	ParamValue type	Description
<b>head.Enable</b>	bool	Enable/Disable
<b>head.PresetId</b>	integer	Range [0—PresetMax] PresetMax is got from interface in <b>GetCurrentProtocolCaps</b> .
<b>head.RuleId</b>	integer	Range [0—MaxNum] MaxNum is got from interface in <b>getCaps</b> .
<b>head.Name</b>	string	Radiometry rule name. char[64]
<b>head.Type</b>	string	Range is {Spot, Line, Area }.

ParamName	ParamValue type	Description
<b>head.MeterRegion.Coordinates[PointNo][0]</b>	integer	Range [0—8091] The Xscale of Region/Line point
<b>head.MeterRegion.Coordinates[PointNo][1]</b>	integer	Range [0—8091] The Yscale of Region/Line point
<b>head.T</b>	integer	Temperature Sample period. Unit is Second.
<b>head.Alarm.Id</b>	integer	Range [0 — 65535],unique alarm id
<b>head.Alarm.Enable</b>	bool	Enable/Disable
<b>head.Alarm.Result</b>	string	Depend on the vaule of Type Spot : {Vaule} Line: { Max, Min, Aver} Area: {Max, Min, Aver, Std, Mid, ISO}
<b>head.Alarm.AlarmCondition</b>	string	Range is {Below, Match , Above }
<b>head.Alarm.Threshold</b>	float	Alarm threshold
<b>head.Alarm.Hysteresis</b>	float	Alarm hysteresis
<b>head.Alarm.Duration</b>	integer	The duration time of alarm. Unit is second
<b>Head.Alarm.PreThreshold</b>	float	PreAlarm threshold
<b>head.Alarm.PreDuration</b>	integer	The duration time of preAlarm. Unit is second
<b>head.LocalParameters.Enable</b>	bool	Enable/Disable
<b>head.LocalParameters.ObjectEmissivity</b>	float	Range [0 — 1] Accuracy is 0.01
<b>head.LocalParameters.ObjectDistance</b>	float	Object distance The range is got from interface in <b>getCaps</b> .
<b>head.LocalParameters.ReflectedTemp</b>	float	Object Reflected Temperature The range is got from interface in <b>getCaps</b> .

## 10.2.4 Heat Image Temper Event

- Get heat image temper event config

Table 10-15

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=HeatImagingTemper
Method	GET
Description	Get Heat Imaging Temper config
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=HeatImagingT

	emper
Success Return	<b>head</b> .Enable=false <b>head</b> .EventHandler. paramName = paramValue
Comment	Parameters in Response: <b>head</b> = table.HeatImagingTemper[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set heat image temper event config

Table 10-16

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig< <b>paramName</b> >=< <b>paramValue</b> >[&< <b>paramName</b> >=< <b>paramValue</b> >...]
Method	GET
Description	Set Heat Imaging Temper config
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&HeatImagingTemper[0].Enable=false&HeatImagingTemper[0].EventHandler.BeepEnable=false
Success Return	OK
Comment	Parameters in URL: The paramName and paramValue are in the table below.  In table below, <b>head</b> = HeatImagingTemper[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

#### Appendix

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	Enable/Disable Heat Imaging Temper feature.
<b>head</b> .EventHandler	—	Setting of EventHandler is described in <u><a href="#">SetEventHandler</a></u> .

## 10.2.5 Get Temperature of Particular Point

Table 10-17

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=getRandomPointTemper&channel=< <b>ChannelNo</b> >&coordinate[0]=x &coordinate[1]=y
Method	GET
Description	Get temperature values of random point.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=getRandomPointTemper&channel=1&coordinate[0]=1024&coordinate[1]=1024
Success Return	TempInfo.Type=Spot TempInfo.TemperAver=27.5
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1.

	<b>x</b> : The Xscale of the point <b>y</b> : The Yscale of the point
--	--

## 10.2.6 Get Temperature of Particular Condition

Table 10-18

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=getTemper&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Get temperature values from rules which have been set.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=getTemper&condition.PresetId=0&condition.RuleId=0&condition.Type=Spot&condition.Name=Spot1&condition.channel=1
Success Return	TempInfo.Type=Spot TempInfo.TemperAver=27.5
Comment	Parameters in URL: The paramName and paramValue are in the table below.

### Appendix

ParamName	ParamValue type	Description
condition.Channel	integer	Video channel index. Start from 1
condition.PresetId	integer	Range [0- PresetMax] PresetMax is got from interface in <b><u>GetCurrentProtocolCaps</u></b> .
condition.RuleId	integer	Range [0- MaxNum] MaxNum is got from interface in <b><u>getCaps</u></b> .
condition.Type	string	Range is {Spot, Line, Area}.
condition.Name	string	Name is got from interface in <b><u>GetThermometryRuleConfig</u></b> .

## 10.2.7 Query Temperature Information

1. Start to query temperature information

Table 10-19

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=startFind&condition.StartTime=<StartTimeValue>&condition.EndTime=<EndTimeValue>&condition.Type=<TypeValue>&condition.channel=<ChannelValue>&condition.Period=<PeriodValue>
Method	GET
Description	Start to query the history data of temperature values.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=startFind&condition.StartTime=2010-04-01%200:00:00&condition.EndTime=2010-04-08%200:00:00&condition.Type=Spot&condition.channel=1&condition.Period=5
Success Return	token=46878 totalCount=333

Comment	The parameters in bold face are as table below.
---------	---

## Appendix

ParamName	ParamValue type	Description
condition.StartTime	string	The start time to find.
condition.EndTime	string	The end time to find.
condition.Type	string	The type of data. Range is {Spot, Line, Area}
condition.channel	integer	Video channel index. Start from 1
condition.Period	integer	Range is {5, 10, 15, 30}, minute

### 2. Get the data of temperature

Table 10-20

Syntax	<code>http://&lt;server&gt;/cgi-bin/RadiometryManager.cgi?action=doFind&amp;token=&lt;tokenvalue&gt;&amp;<b>beginNumber</b>=&lt;BeginNumber&gt;&amp;count=&lt;findNum&gt;</code>
Method	GET
Description	Get the history data of temperature.
Example	<code>http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=doFind&amp;token=46878&amp;beginNumber=16&amp;count=16</code>
Success Return	<pre>found=12 info[i].Time=2010-04-08 16:12:46 info[i].PresetId=0 info[i].RuleId=0 info[i].Type=Spot info[i].Name=xxxx info[i].Coordinate[0]=1024 info[i].Coordinate[1]=2048 info[i].Channel=0 info[i].TemperatureUnit=Centigrade info[i].QueryTemperInfo.TemperAve=50.1 info[i].QueryTemperInfo.TemperMax=50.2 info[i].QueryTemperInfo.TemperMin=50.0</pre>
Comment	<p>Parameters in URL:</p> <p><b>token</b>: query token, get from interface of the first step above.</p> <p><b>beginNumber</b>: the begin index in this query.</p> <p><b>count</b>: the number you want to query.</p> <p>Params in Resp:</p> <p><b>i</b>: the array index.</p>

### 3. Stop finding temperature information

Table 10-21

Syntax	<code>http://&lt;server&gt;/cgi-bin/RadiometryManager.cgi?action=stopFind&amp;token=&lt;tokenvalue&gt;</code>
Method	GET
Description	Stop to find the history data of temperature values.
Example	<code>http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=stopFind&amp;token=46878</code>
Success	OK

Return	
Comment	<b>token</b> : query token, get from interface of the first step.

## 10.2.8 Subscribe to Temperature Information

Table 10-22

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=attachTemper&channel=< <b>Chan</b> <b>nelNo</b> >[&heartbeat=< <b>PERIOD</b> >]
Method	GET
Description	Subscribe to temperature information of a channel.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=attachTemper&channel=2 &heartbeat=1
Success Return	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;data length&gt;  info[i].Time=2010-04-08 16:12:46 info[i].PresetId=0 info[i].RuleId=0 info[i].Type=Spot info[i].Name=xxxx info[i].Coordinate[0]=1024 info[i].Coordinate[1]=2048 info[i].Channel=0 info[i].TemperatureUnit=Centigrade info[i].QueryTemperInfo.TemperAve=50.1 info[i].QueryTemperInfo.TemperMax=50.2 info[i].QueryTemperInfo.TemperMin=50.0  --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt;  ..... </pre>
Comment	Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. Params in Resp : <b>i</b> : the array index. <b>PERIOD</b> : integer, period of heartbeat, starts from 1.

## 10.2.9 Subscribe to Radiometry Data

Table 10-23

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=attachProc&channel=< <b>Channel No</b> >[&heartbeat=< <b>heartbeat</b> >]
Method	GET
Description	Subscribe to radiometry data of a channel. It needs to cooperate with interface below.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=attachProc&channel=2&heartbeat=1
Success Return	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;data length&gt;  dataInfo.Height=0 dataInfo.Width=0 dataInfo.Channel=0 dataInfo.Time=2010-05-25 00:00:00 dataInfo.Length=0 dataInfo.sensorType="Tau" dataInfo.Unzip.ParamR=1 dataInfo.Unzip.ParamB=1 dataInfo.Unzip.ParamF=1 dataInfo.Unzip.ParamO=1  --&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;data length&gt;  &lt;Binary data&gt;  --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt; ..... </pre>
Comment	<p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p><b>heartbeat</b>: range is [1,60]</p>

## 10.2.10 To Fetch Radiometry Data

Table 10-24

Syntax	http://<server>/cgi-bin/RadiometryManager.cgi?action=toFetch&channel=< <b>ChannelNo</b> >
Method	GET
Description	Start to fetch radiometry data.
Example	http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=toFetch&channel=2
Success Return	status=Ready
Comment	<b>ChannelNo</b> : integer, video channel index which starts from 1. <b>status</b> : Range is {Ready, Busy}. "Ready" means service available and "Busy" means service busy.

## 10.2.11 Get FireWarning Config

Table 10-25

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=FireWarning
Method	GET
Description	Get FireWarning config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=FireWarning
Success Return	<pre> head.Enable=true head.PresetId=0 head.Row = 31 head.Col = 40 head.Mode="Auto" head.SmdFilterEnable=false head.MovingTargetFilterEnable=false head.SunReflectEnable=false head.TimeDurationEnable=false head.FireDuration= 15 head.DetectWindow[windowsNum].Regions[0]=123468789 head.DetectWindow[windowsNum].Regions[1]=123468789 head.DetectWindow[windowsNum].Regions[2]=123468789 head.DetectWindow[windowsNum].Regions[3]=123468789 head.DetectWindow[windowsNum].Postion[0]=0 head.DetectWindow[windowsNum].Postion[1]=0 head.DetectWindow[windowsNum].Postion[2]=0 head.DetectWindow[windowsNum].Postion[3]=0 head.DetectWindow[windowsNum].Sensitivity = 95 head. DetectWindow[windowsNum].Id=1 head.DetectWindow[windowsNum].Name="windName" head.EventHandler=(output of EventHandler is described in GetEventHandler)                     </pre>
Comment	Parameters in Response:

	<p><b>head</b>= table.FireWarning[ChannleNo][RuleNum]</p> <p><b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p> <p><b>RuleNum</b>: integer,array index starts from 0,which means rules in each Preset, If FireWarningMode is "SpaceExClude", then only the first rule used.</p> <p><b>windowsNum</b>: support 4 Nums if FireWarningMode is "PtzPreset", 8 if "SpaceExClude"</p> <p><b>Postion</b>: only valid in "SpaceExClude"Mode</p>
--	--

## 10.2.12 Set FireWarning Config

Table 10-26

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set FireWarning config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&FireWarning[0][0].Enable=false
Success Return	OK
Comment	head = FireWarning[ChNum][RuleNum]

### Appendix:

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	whether fire detect take effect
<b>head</b> .PresetId	integer	The PresetId
<b>head</b> .Row	integer	rows of fire detect area
<b>head</b> .Col	integer	cols of fire detect area
<b>head</b> .Mode	string	Range is {"Auto","Normal"}.
<b>Head</b> .SmdFilterEnable	bool	whether smd filter take effect
<b>Head</b> .MovingTargetFilterEnable	bool	whether moving target filter take effect
<b>Head</b> .SunReflectEnable	bool	whether sun reflect filter take effect
<b>head</b> .TimeDurationEnable	bool	whether include fire detect duration,only take effect in SpaceExClude mode
<b>head</b> .FireDuration	integer	fire last times
<b>head</b> .DetectWindow[windowsNum].Regions[0]	integer	detect area mask
<b>head</b> .DetectWindow[windowsNum].Sensitivity	integer	Rage {0,100}
<b>head</b> .DetectWindow[windowsNum].Id	integer	Rage {0,...}
<b>head</b> .DetectWindow[windowsNum].Name	string	detect window name
<b>head</b> .EventHandler	EventHandler	Setting of EventHandler is described in SetEventHandler.

## 10.2.13 Get FireWarningMode Config

Table 10-27

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=FireWarningMode
Method	GET
Description	Get FireWarningMode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=FireWarningMode
Success Return	head.Mode="PtzPreset"
Comment	Parameters in Response: head= table.FireWarningMode[ChannelNo] ChannelNo: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

## 10.2.14 Set FireWarningMode Config

Table 10-28

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig<paramName>=<paramValue>[&<paramName>=<paramValue> ...]
Method	GET
Description	Set FireWarningMode config.
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&FireWarningMode[0].Mode="SpaceExClude"
Success Return	OK
Comment	<b>head</b> = FireWarningMode[ChannelNo]

### Appendix:

ParamName	ParamValue type	Description
head.Mode	string	Range is {"PtzPreset", "SpaceExClude"}

## 10.2.15 Get Current Hot Cold Spot

Table 10-29

URL	http://<server>/cgi-bin/TemperCorrection.cgi?action=getCurrentHotColdSpot		
Method	GET		
Description	Get the max and min temperature values.		
[ Request Params ] ( key=value format in URL )			
channel	int	R	video channel index which starts from 1.
[ Response Params ] (key=value format)			

Name	Type	R/O	Param Description
info	object	R	The current hot code spot info.
+HotPoint	array<int>	R	The hot spot position, must be two int, means x and y value, coordinate remap to 0 — 8192.
+HotSpotValue	double	R	The hot spot temperature value.
+ColdPoint	array<int>	O	The cold spot position, must be two int, means x and y value, coordinate remap to 0 — 8192.
+ColdSpotValue	double	O	The cold spot temperature value.
+TemperatureUnit	int	O	The temperature unit : 0 Centigrade, 1 Fahrenheit
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/TemperCorrection.cgi?action=getCurrentHotColdSpot&channel=1		
Response	info.HotPoint[0]=1150 info.HotPoint[1]=2320 info.HotSpotValue=35.5 info.ColdPoint[0]=5452 info.ColdPoint[1]=6192 info.ColdSpotValue=24.3 info.TemperatureUnit=0		

## 10.2.16 Heat Image Temper PreAlarm Event

- Get heat image temper pre event config

Table 10-30

Syntax	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=PreAlarmEvent
Method	GET
Description	Get Heat Imaging Temper PreAlarm Event config
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=PreAlarmEvent
Success Return	<b>head</b> .Enable=false <b>head</b> .EventHandler.paramName = paramValue
Comment	Parameters in Response: <b>head</b> = table.PreAlarmEvent[ <b>ChannelNo</b> ] <b>ChannelNo</b> : integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).

- Set heat image temper pre event config

Table 10-31

Syntax	http://<server>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Method	GET
Description	Set Heat Imaging Temper PreAlarm Event config
Example	http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&PreAlarmEvent[0].En

	able=false&PreAlarmEvent[0].EventHandler.BeepEnable=false
Success Return	OK
Comment	<p>Parameters in URL: The paramName and paramValue are in the table below.</p> <p>In table below, <b>head=PreAlarmEvent[ChannelNo]</b> <b>ChannelNo</b>: integer, array index starts from 0, which means video channel (equals to video channel index -1, and so 0 means channel 1).</p>

#### Appendix

ParamName	ParamValue type	Description
<b>head.Enable</b>	bool	Enable/Disable Heat Imaging Temper feature.
<b>head.EventHandler</b>	—	Setting of EventHandler is described in <a href="#">SetEventHandler</a> .

## 10.2.17 Get Heat Map Info

Get Heat Map Info.

<b>Request URL</b>	http://<server>/cgi-bin/RadiometryManager.cgi?action=getHeatMapsDirectly&channel=<ChannelNo>			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	R	video channel index which starts from 1	2
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/RadiometryManager.cgi?action=getHeatMapsDirectly&channel=2				

Response Params	( multipart, binary in body )			
Name	Type	R/O	Description	Example
dataInfo	object	O	Data info	
+Height	integer	O	height	0
+Width	integer	O	width	0
+Channel	integer	O	Channel number	0
+Time	string	O	Time of getting data	2010-05-25 00:00:00
+Length	integer	O	Data length	0
+sensorType	string	O	Algorithm type	"Tau"
+Unzip	object	O	unzip parameters, it works when sensorType was "Tau".	
++ParamR	integer	O	Parameter R for temperature conversion	1
++ParamB	integer	O	Parameter B for	1

			temperature conversion	
++ParamF	integer	O	Parameter F for temperature conversion	1
++ParamO	integer	O	Parameter O for temperature conversion	1

**Response Example**

HTTP/1.1 200 OK  
 Server: Device/1.0  
 Content-Type: multipart/x-mixed-replace; boundary=<boundary>  
 Connection: closed

--<boundary>  
 Content-Type: text/plain  
 Content-Length: <data length>

dataInfo.Height=0  
 dataInfo.Width=0  
 dataInfo.Channel=0  
 dataInfo.Time=2010-05-25 00:00:00  
 dataInfo.Length=0  
 dataInfo.sensorType="Tau"  
 dataInfo.Unzip.ParamR=1  
 dataInfo.Unzip.ParamB=1  
 dataInfo.Unzip.ParamF=1  
 dataInfo.Unzip.ParamO=1

--<boundary>  
 Content-Type: application/octet-stream  
 Content-Length: <data length>

<Binary data>

--<boundary>

.....

## 10.3 TemperCustom

### 10.3.1 Set Environment Temperature

Syntax	http://<server>/cgi-bin/TemperCustom.cgi?action=setEnvTemp		
Method	GET		
Description	Set environment temperature		
[ Request Params ] (key=value format at URL )			
Name	Type	R/O	Param Description
EnvironmentTemp	int	R	environment temperature Magnified 100 times,the unit is 0.01Celsius degree
[ Response Params ] ( OK )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/TemperCustom.cgi?action=setEnvTemp&EnvironmentTemp=2800		
Response	OK		

# 11

## Access Control APIs

### 11.1 Door

#### 11.1.1 Open Door

Table 11-1

Syntax	http://<server>/cgi-bin/accessControl.cgi?action=openDoor&channel=< <b>ChannelNo</b> >[&UserID=< <b>UserID</b> >&Type=< <b>Type</b> >]
Method	GET
Description	Open the door.
Example	http://192.168.1.108/cgi-bin/accessControl.cgi?action=openDoor&channel=1&UserID=101&Type=Remote
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, the index of door, starts from 1. <b>UserID</b> : remote user ID. <b>Type</b> : the open type, default value is "Remote".

#### 11.1.2 Get Door Status

Table 11-2

Syntax	http://<server>/cgi-bin/accessControl.cgi?action=getDoorStatus&channel=< <b>ChannelNo</b> >
Method	GET
Description	Get status of the door.
Example	http://192.168.1.108/cgi-bin/accessControl.cgi?action=getDoorStatus&channel=1
Success Return	Info.status=Open
Comment	Parameters in URL : <b>ChannelNo</b> : integer, the index of door, starts from 1. Parameters in Response : <b>status</b> : the range is {Open, Break, Close}

#### 11.1.3 Close Door

Table 11-3

Syntax	http://<server>/cgi-bin/accessControl.cgi?action=closeDoor&channel=< <b>ChannelNo</b> >[&UserID=< <b>UserID</b> >&Type=< <b>Type</b> >]
Method	GET

Description	Close the door.
Example	http://192.168.1.108/cgi-bin/accessControl.cgi?action=closeDoor&channel=1&UserID=101&Type=Remote
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, the index of door, starts from 1. <b>UserID</b> : remote user ID. <b>Type</b> : the open type, default value is "Remote".

## 11.1.4 Capture Fingerprint

Only ID verification terminals such as ASHZ320/520 are supported. Common access control is not supported.

Table 11-4

Syntax	http://<server>/cgi-bin/accessControl.cgi?action=captureFingerprint		
Method	GET		
Description	Capture fingerprint data, then report it via 'Fingerprint' event.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
info	object	R	fingerprint data
+ReaderID	string	R	Reader machine ID
+FingerPrintName	string	O	Fingerprint name (For smart building products)
+ UserID	string	O	User Identity (For smart building products)
heartbeat	int	R	Heartbeat interval (sec)
timeout	int	R	Timeout (sec)
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/accessControl.cgi?action=captureFingerprint&info.ReaderID=101&info.FingerPrintName=aaaa&info.UserID=10221&heartbeat=5&timeout=10		
Response	OK		

## 11.1.5 Capture face picture

Only ID verification terminals such as ASHZ320/520 are supported. Common access control is not supported.

Table 11-5

URL	http://<server>/cgi-bin/accessControl.cgi?action=captureCmd
-----	---

Method	GET		
Description	Capture face data, then report it via 'CitizenPictureCompare' event.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
+type	int	R	Capture entity types: 0x01 - Face 0x02 - Identification card
+ UserID	string	O	User ID
heartbeat	int	R	Heartbeat interval (sec)
timeout	int	R	Timeout (sec)
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/accessControl.cgi?action=captureCmd&type=1&UserID=10221&heartbeat=5&timeout=10		
Response	OK		

## 11.2 Access Control

### 11.2.1 Add Access Control Custom Password

Table 11-6

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=AccessControlCustomPassword		
Method	GET		
Description	Insert access control custom password.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access control custom password record name, should be <b>"AccessControlCustomPassword"</b> .
UserID	string	R	The user's id.
OpenDoorPassword	string	R	This user's open door password.
AlarmPassword	string	O	This user's alarm password. ( This param is not supported by video talk device )
Doors	array<int>	O	The index of the doors that custom password can open. ( This param is not supported by video talk device )
TimeSections	array<int>	O	The index of the time sections of each door that this card can open. ( This param is not supported by video talk device )
VTOPosition	string	O	VTO position number. ( This param is not supported by access control device )
ValidDateStart	string	O	The start of valid date, format is <b>"yyyyMMdd hhmss"</b> .
ValidDateEnd	string	O	The end of valid date, format is <b>"yyyyMMdd hhmss"</b> .
ValidCounts	int	O	The password's valid counts.

OriginSmartGate Way	string	O	The origin smart gateway address. ( Video talk device customize. )
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
RecNo	int	R	The record id.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=insert&name=AccessControlCustomPassword&UserID=102&OpenDoorPassword=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&VTOPosition=01018001&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811&ValidCounts=30		
Response	RecNo=12345		

## 11.2.2 Modify Access Control Custom Password

Table 11-7

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=update&name=AccessControlCustomPassword		
Method	GET		
Description	Update access control custom password. <b>Note:</b> You should provide at lease one optional param to update.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access control custom password record name, should be " <b>AccessControlCustomPassword</b> ".
recno	int	R	The record id.
UserID	string	R	The user's id.
OpenDoorPassword	string	R	Ths user's open door password.
Doors	array<int>	O	The index of the doors that custom password can open.
...<other param>	—	—	...<See above insert command for other params of the record, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=update&name=AccessControlCustomPassword&recno=12345&UserID=102&OpenDoorPassword=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811		
Response	OK		

## 11.2.3 Delete Access Control Custom Password

- Delete access control custom password record by recno

Table 11-8

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=remove&name=AccessControlCustomPassword		
Method	GET		
Description	Remove the access control custom password record by recno.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be " <b>AccessControlCustomPassword</b> ".
recno	int	R	The record id.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=remove&name=AccessControlCustomPassword&recno=12345		
Response	OK		

- Delete all the access control custom password records

Table 11-9

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=clear&name=AccessControlCustomPassword		
Method	GET		
Description	Remove all the access control custom password records		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be " <b>AccessControlCustomPassword</b> ".
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=clear&name=AccessControlCustomPassword		
Response	OK		

## 11.2.4 Find Access Control Custom Password

- Find Access control custom password by condition

Table 11-10

URL	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=AccessControlCustomPassword		
Method	GET		
Description	Find Access control custom password by condition.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Record Name, should be " <b>AccessControlCustomPassword</b> ".

count	int	O	Max result to return, default is 1024.
StartTime	string	O	The start of the record's CreateTime.
EndTime	string	O	The End of the record's CreateTime.
condition	object	O	Search condition.
+UserID	string	O	The user's id.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
totalCount	int	O	The total record num that find.
found	int	O	The record num that returned.
records	array<object>	R	The records that returned.
+RecNo	int	R	The record id.
+CreateTime	int	O	The create time of record.
+UserID	string	R	The user's id.
+OpenDoorPassword	string	R	This user's open door password.
+AlarmPassword	string	O	This user's alarm password. ( This param is not supported by video talk device )
+Doors	array<int>	O	The index of the doors that custom password can open. ( This param is not supported by video talk device )
+TimeSections	array<int>	O	The index of the time sections of each door that this card can open. ( This param is not supported by video talk device )
VTOPosition	string	O	VTO position number. ( This param is not supported by video talk device )
+ValidDateStart	string	O	The start of valid date, format is "yyyyMMdd hhmmss".
+ValidDateEnd	string	O	The end of valid date, format is "yyyyMMdd hhmmss".
+ValidCounts	int	O	The password's valid counts.
+OriginSmartGateway	string	O	The origin smart gateway address. ( Video talk device customize. )
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=AccessControlCustomPassword&condition.UserID=103&StartTime=123456700&EndTime=123456800&count=100		

Response	<pre> totalCount=1000 found=100 records[0].RecNo=12345 records[0].CreateTime=123456789 records[0].UserID=103 records[0].OpenDoorPassword=123456 records[0].Doors[0]=1 records[0].Doors[1]=3 records[0].Doors[2]=5 records[0].VTOPosition=01018001 records[0].ValidStart=20151022 093811 records[0].ValidEnd=20151222 093811 ... records[1].RecNo=13579 records[1].CreateTime=123456799 records[1].UserID=103 records[0].OpenDoorPassword=123456 records[1].Doors[0]=2 records[1].Doors[1]=4 records[1].Doors[2]=6 records[1].VTOPosition=01018002 records[1].ValidStart=20151022 093811 records[1].ValidEnd=20151222 093811 ... </pre>
----------	--

- Find Access control custom password by recno

Table 11-11

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=get&name=AccessControlCustomPassword		
Method	GET		
Description	Find Access control custom password by recno.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Record Name, should be " <b>AccessControlCustomPassword</b> ".
recno	int	R	The record id.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
record	object	R	The record that returned.
+RecNo	int	R	The record id.
+CreateTime	int	O	The create time of record.
+UserID	string	R	The user's id.
...<other param>	—	—	...<See above find command for other params of the record.>
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=get&name=AccessControlCustomPassword&recno=3		

Response	<pre> record.RecNo=3 record.CreateTime=123456789 record.UserID=103 record.OpenDoorPassword=123456 record.Doors[0]=1 record.Doors[1]=3 record.Doors[2]=5 record.VTOPosition=01018001 record.ValidStart=20151022 093811 record.ValidEnd=20151222 093811 ... </pre>
----------	--

## 11.2.5 Get the Total Number of Records of Access Control Custom Password

Table 11-12

URL	http://<server>/cgi-bin/recordFinder.cgi?action=getQuerySize&name=AccessControlCustomPassword		
Method	GET		
Description	Get the total number of records of the access control custom password.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be "AccessControlCustomPassword".
[ Response Params ] ( key=value )			
count	int	R	The total number of records.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=getQuerySize&name=AccessControlCustomPassword		
Response	count=150		

## 11.2.6 Get Access Control Caps

Table 11-13

URL	http://<server>/cgi-bin/accessControlManager.cgi?action=getCaps		
Method	GET		
Description	Get access control capabilities.		
[ Request Params ] ( none )			
[ Response Params ] ( key=value )			
caps	object	R	The access control capabilities.
+AccessControl Channels	int	R	The access control channel number.

+AccessControlAlarmRecord	bool	0	Support log access control alarm record or not. ( This param is not supported by video talk device )
+CustomPasswordEncryption	int	0	The custom password crypt type, 0 : plain text, 1 : MD5
+SupportFingerPrint	int	0	Support fingerprint type, 0 : unknown, 1 : not support, 2 : support
+OnlySingleDoorAuth	int	0	Support single door auth type, 0 : not support, 1 : support, ( This param is not used by video talk device )
+AsynAuth	int	0	Support async auth type, 0 : not support, 1 : support, ( This param is not supported by video talk device )
+SpecialDaysSchedule	object	0	Special days schedule capabilities, ( This param is not supported by video talk device )
++Support	bool	0	Support special days schedule or not. ( This param is not supported by video talk device )
++MaxSpecialDaysSchedules	uint	0	Max special days schedule number. ( This param is not supported by video talk device )
++MaxTimePeriodsPerDay	uint	0	Max time periods per day. ( This param is not supported by video talk device )
++MaxSpecialDayGroups	uint	0	Max special day groups. ( This param is not supported by video talk device )
++MaxDaysInSpecialDayGroup	uint	0	Max special days in special day group. ( This param is not supported by video talk device )

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getCaps
Response	caps.AccessControlChannels=3 caps.AccessControlAlarmRecord=true caps.CustomPasswordEncryption=1 caps.SupportFingerPrint=1 caps.OnlySingleDoorAuth=1 caps.AsynAuth=1 caps.SpecialDaysSchedule.Support=true caps.SpecialDaysSchedule.MaxSpecialDaysSchedules=5 caps.SpecialDaysSchedule.MaxTimePeriodsPerDay=5 caps.SpecialDaysSchedule.MaxSpecialDayGroups=5 caps.SpecialDaysSchedule.MaxDaysInSpecialDayGroup=5

## 11.2.7 Get Lock Status

Syntax	http://<server>/cgi-bin/accessControl.cgi?action=getLockStatus&channel=<ChannelNo>
Method	GET
Description	Get status of the lock.
Example	http://192.168.1.108/cgi-bin/accessControl.cgi?action=getLockStatus &channel=1
Success Return	onLineStatus ="Online" status ="Close"
Comment	Parameters in URL :

	<p><b>ChannelNo:</b> integer, the index of lock, starts from 1. Default is 1 if not present.</p> <p>Parameters in Response :</p> <p><b>status:</b> the range is { "Open", "Close", "Abnormal", "FakeLocked", "Unknown" }</p> <p><b>onLineStatus:</b> the range is { "OnLine", "OffLine" }</p>
--	---

## 11.2.8 [Config] Access Control General Setting

- Get access control general setting

Table 11-14

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AccessControlGeneral		
Method	GET		
Description	Get access control general setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+AccessControlGeneral	object	R	AccessControlGeneral config object
++AccessProperty	string	O	Access property, can be : "unidirect", "bidirect" ( This param is supported by access control device. )
+ABLock	object	O	AB Lock setting ( This param is supported by access control device. )
+++Enable	bool	O	Enable AB Lock or not.
+++Doors	Array<Array<int> >	O	AB Lock groups, each group has several doors, one door can be opened only when all other doors in the AB lock group are in closed state.
++CustomPasswordEnable	bool	O	Whether to enable custom password. ( This param is supported by video talk device. )
++CommonPassword	string	O	The common password. ( This param is supported by video talk device. )
++ButtonExitEnable	bool	O	Whether to enable the open door button. ( This param is supported by video talk device. )
++CheckSensorBeforeLock	bool	O	Whether to check the sensor before lock the door. ( This param is supported by video talk device. )
++CheckSensorTime	int	O	The check sensor time, in seconds. ( This param is supported by video talk device. )
++DuressPassword	string	O	The duress password. ( This param is supported by video talk device. )
++DuressEnable	bool	O	Whether to enable duress password. ( This param is supported by video talk device. )
++UnlockRecordType	Array<string>	O	Unlock Record Type, can be {"Password", "Button", "FingerPrint", "QRCode", "Card", "Remote", "BlueTooth",

			"Face"}
++SensorType	integer	O	sensor type, 0 for always open; 1 for always close.
++CallLiftType	string	O	call lift protocol type
++CallLiftEnable	bool	O	whether to enable call lift
++AccessVoice	object	O	Door opening prompt tone
+++CurrentVoice ID	integer	O	the current prompt tone ID, default value 0
+++VoiceList	array<object>	O	voice list, max length is 16 voice ID can be: { 0: Validate succeed 1: door open succeed 2: unlock succeed 3: punch time clocks succeed 4: welcome 5: Welcome to come again 6: thank you 7: custom voice }
++++VoiceID	integer	O	voice id
++++VoiceName	string	O	voice info
++++FileName	string	O	file full path with name

[ Example ]

Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=AccessControlGeneral
Response	table.AccessControlGeneral.AccessProperty=bidirect table.AccessControlGeneral.ABLock.Enable=true table.AccessControlGeneral.ABLock.Doors[0][0]=1 table.AccessControlGeneral.ABLock.Doors[0][1]=2 table.AccessControlGeneral.ABLock.Doors[0][2]=3 table.AccessControlGeneral.ABLock.Doors[1][0]=4 table.AccessControlGeneral.ABLock.Doors[1][1]=5 table.AccessControlGeneral.ABLock.Doors[1][2]=6 table.AccessControlGeneral.CustomPasswordEnable=true table.AccessControlGeneral.CommonPassword=123456 table.AccessControlGeneral.ButtonExitEnable=true table.AccessControlGeneral.CheckSensorBeforeLock=true table.AccessControlGeneral.CheckSensorTime=30 table.AccessControlGeneral.DuressPassword=654321 table.AccessControlGeneral.DuressEnable=true

- Set access control general setting

Table 11-15

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig
Method	GET
Descripti	Set access control general setting.

on			
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
AccessControlGeneral	object	R	AccessControlGeneral config object
+AccessProperty	string	O	Access property, can be : "unidirect", "bidirect" ( This param is supported by access control device. )
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&AccessControlGeneral.AccessProperty=bidirect&AccessControlGeneral.ABLock.Enable=false&AccessControlGeneral.CustomPasswordEnable=true&AccessControlGeneral.CommonPassword=123456&AccessControlGeneral.ButtonExitEnable=true&AccessControlGeneral.CheckSensorBeforeLock=true		
Response	OK		

## 11.2.9 [Config] Access Control Setting

- Get access control setting

Table 11-16

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AccessControl		
Method	GET		
Description	Get access control setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	The config table
+AccessControl	array<object>	R	The AccessControl config array for every access control channel
++Enable	bool	O	Whether to enable config for this channel.
++State	string	O	Door state, can be : "Normal", "CloseAlways", "OpenAlways", "NoPersonNC", "NoPersonNO"
++Method	int	O	Open door method, can be following value, default is 2 : 0 : only by password 1 : only by access card 2 : by password or access card 3 : by access card first then password 4 : by password first then access card 5 : different method in different time range 6 : only by fingerprint 7 : by password or access card or fingerprint 8 : by password and access card and fingerprint together

			<p>9 : by password and fingerprint together</p> <p>10 : by access card and fingerprint together</p> <p>11 : by multiple access user</p> <p>12 : by face ( match with picture in ID card )</p> <p>13 : by face and ID card</p> <p>14 : by face ( match with picture in ID card ) or access card or fingerprint</p> <p>15 : by face and ID card, or access card or fingerprint</p> <p>16 : by UserID and password</p> <p>17 : only by face</p> <p>18 : by face and password together</p> <p>19 : by fingerprint and password together</p> <p>20 : by fingerprint and face together</p> <p>21 : by access card and face together</p> <p>22 : by face or password</p> <p>23 : by fingerprint or password</p> <p>24 : by fingerprint or face</p> <p>25 : by access card or face</p> <p>26 : by access card or fingerprint</p> <p>27 : by fingerprint and face and password together</p> <p>28 : by access card and face and password together</p> <p>29 : by access card and fingerprint and password together</p> <p>30 : by access card and fingerprint and face together</p> <p>31 : by fingerprint or face or password</p> <p>32 : by access card or face or password</p> <p>33 : by access card or fingerprint or face</p> <p>34 : by access card and fingerprint and face and password together</p> <p>35 : by access card or fingerprint or face or password</p> <p>36 : by face and ID card, or access card or face</p> <p>37 : by face ( match with picture in ID card ) or access card or face</p> <p>38 : by access card and password, or fingerprint and password</p> <p>39 : by ID card ( match with picture in ID card ) or face</p> <p>40 : by ID card ( match with fingerprint in ID card )</p> <p>41 : by ID card ( match with fingerprint and picture in ID card )</p> <p>42 : by ID card or access card or fingerprint or face or password</p> <p>43 : by multy user method</p> <p>44 : by ID card or health code</p>
++OpenAlwaysTime	int	O	The time range that working, value is index in AccessTimeSechdule config.
++CloseAlwaysTime	int	O	The time range that always closed, value is index in AccessTimeSechdule config.
++HolidayTime	int	O	The time range that working in holidays, value is index in

			AccessTimeSechdule config.
++UnlockHoldInterval	int	O	The Unlock holding interval, unit is milliseconds, value should between 250 to 20000. ( This param is also supported by video talk device. )
++UnlockReloadInterval	int	O	The Unlock reload interval, unit is milliseconds. ( This param is also supported by video talk device. )
++AccessProtocol	string	O	The access control process, can be : "Local", "Dahua", "Remote", "Private" ( This param is also supported by video talk device. )
++BreakInAlarmEnable	bool	O	Whether to enable the breakin alarm.
++RepeatEnterAlarm	bool	O	Whether to enable the repeat enter alarm.
++DoorNotClosedAlarmEnable	bool	O	Whether to enable the alarm when door not closed.
++DuressAlarmEnable	bool	O	Whether to enable the duress alarm.
++FirstEnter	object	O	The first enter setting.
+++Enable	bool	O	Whether to enable the first enter function,
+++Status	string	O	The status, can be : "KeepOpen" "Normal"
+++Time	int	O	The time range that enable first enter function, value is index in AccessTimeSechdule config.
++CardNoConvert	int	O	The card number convert setting : 0 – not convert, 1 – convert using NOT operation, 2 : use HIDpro convert
++MaliciousAccessControlEnable	bool	O	Whether to enable malicious access alarm.
+AutoRemoteCheck	object	O	auto open door remotely
+++Enable	bool		whether to enable the function
+++Time	uint		working period , value is index in AccessTimeSechdule config.
++Name	string	O	name
++RepeatEnterTime	integer	O	Repeat entry time, unit second, 0 means do not start
++CloseTimeout	integer	O	Closing overtime
++SensorEnable	bool	O	sensor enable
++TimeSchedule	Array<Array<object>>		Open the door in different periods it works when Method is "different method in different time range".
+++TimeSection	string	O	time section, format as "hh:mm:ss-hh:mm:ss"
+++Method	integer	O	Open door method, can be following value, default is 2 : 0 : only by password 1 : only by access card 2 : by password or access card 3 : by access card first then password

			4 : by password first then access card 5 : different method in different time range 6 : only by fingerprint 7 : by password or access card or fingerprint 8 : by password and access card and fingerprint together 9 : by password and fingerprint together 10 : by access card and fingerprint together 11 : by multiple access user 17 : only by face 35 : by access card or fingerprint or face or password
++CustomPasswordEnable	bool	O	whether to enable custom password
++RemoteCheck	bool	O	Is platform validation required
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=AccessControl		
Response	<pre> table.AccessControl[0].Enable=true table.AccessControl[0].State=Normal table.AccessControl[0].Method=2 table.AccessControl[0].OpenAlwaysTime=1 table.AccessControl[0].CloseAlwaysTime=2 table.AccessControl[0].UnlockHoldInterval=500 table.AccessControl[0].AccessProtocol=Local table.AccessControl[0].BreakInAlarmEnable=true table.AccessControl[0].DuressAlarmEnable=true table.AccessControl[0].CardNoConvert=0 table.AccessControl[0].MaliciousAccessControlEnable=true ... </pre>		

- Set access control setting

Table 11-17

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set access control setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
AccessControl	array<object>	R	AccessControl config object array
+Enable	bool	O	Whether to enable config for this channel.
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	<pre> GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&amp;AccessControl[0].Enable=true&amp; AccessControl[0].State=Normal&amp;AccessControl[0].Method=2&amp;AccessControl[0].OpenAlwaysTime=1&amp;AccessControl[0].CloseAlwaysTime=2&amp;AccessControl[0].UnlockHoldInterval=5 </pre>		

	00&AccessControl[0].AccessProtocol=Local&AccessControl[0].BreakInAlarmEnable=true&AccessControl[0].DuressAlarmEnable=true&AccessControl[0].CardNoConvert=0&AccessControl[0].MaliciousAccessControlEnable=true
Response	OK

## 11.2.10 [Config] Wiegand Setting

- Get Wiegand setting

Table 11-18

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=Wiegand		
Method	GET		
Description	Get Wiegand setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+Wiegand	array<object>	R	Wiegand config
++Mode	int	R	Wiegand mode, 0: wiegand input, 1 : wiegand output
++PulseWidth	int	R	The pulse width, unit is microseconds
++PulseStep	int	R	The pulse step, unit is microseconds
++TransferMode	int	R	The transfer mode, 0 : wiegand 34bit transfer, 4 byte card number, 2 bit checksum, 1 : wiegand 66bit transfer, 8 byte card number, 2 bit checksum, 2 : wiegand 26bit transfer, 3 byte card number, 2 bit checksum,
++OutType	int	R	The output type, 0 : output ID, 1 : output card number
++InputType	integer	O	the input type, the value is 32bits. It won't accept any input when value is 0. Bit0: card number input Bit1: password input bit2~Bit31:reserved
++Doors	integer	O	door index, starts from 0, and 0 means door 1
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=Wiegand		
Response	table.Wiegand[0].Mode=1 table.Wiegand[0].PulseWidth=200 table.Wiegand[0].PulseStep=1000 table.Wiegand[0].TransferMode=1 table.Wiegand[0].OutType=1 table.Wiegand[1].Mode=1 table.Wiegand[0].InputType=3 table.Wiegand[0].Doors=0 ...		

- Set Wiegand setting

Table 11-19

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set Wiegand setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Wiegand	array<object>	R	Wiegand config object
+Mode	int	O	Wiegand mode, 0 : wiegand input, 1 : wiegand output
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&Wiegand[0].Mode=1&Wiegand[0].PulseWidth=200&Wiegand[0].PulseStep=1000&Wiegand[0].TransferMode=1&Wiegand[0].OutType=1		
Response	OK		

### 11.2.11 [Config] Access Time Schedule Setting

The following Apis are not supported by video talk device.

- Get access time schedule setting

Table 11-20

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=AccessTimeSchedule		
Method	GET		
Description	Get access time schedule setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	array<object>	R	config table object
+AccessTimeSchedule	array<object>	R	AccessTimeSchedule config object array
++Name	string	O	The schedule name, max string length is 63.
++Enable	bool	R	Whether to enable this time schedule
++TimeSechedule	array< array< string > >	O	The time schedule array, first array has max 8 item, refer to 7 day in a week (the first one is Sunday)and last one is holiday, each item is an array has max 6 time section, each time section is a string, format is "enable, hour:minute:second – hour:minute:second", for example : "1 00:00:00-12:00:00"
[ Example ]			
Request	GET		

	http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=AccessTimeSchedule
Response	table.AccessTimeSchedule[0].Name=TS1 table.AccessTimeSchedule[0].Enable=true table.AccessTimeSchedule[0].TimeSchedule[0][0]=1 00:00:00-12:00:00 table.AccessTimeSchedule[0].TimeSchedule[0][1]=1 15:00:00-20:00:00 table.AccessTimeSchedule[0].TimeSchedule[1][0]=1 00:00:00-12:00:00 table.AccessTimeSchedule[0].TimeSchedule[1][1]=1 15:00:00-20:00:00 ...

- Set access time schedule setting

Table 11-21

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set access time schedule setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
AccessTimeSchedule	array<object>	R	The access time schedule config object array
+Name	string	O	The schedule name, max string length is 63.
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&AccessTimeSchedule[0].Name=TS1&AccessTimeSchedule[0].Enable=true&AccessTimeSchedule[0].TimeSchedule[0][0]=1%2000:00:00-12:00:00&AccessTimeSchedule[0].TimeSchedule[0][1]=1%2015:00:00-20:00:00&AccessTimeSchedule[0].TimeSchedule[1][0]=1%2000:00:00-12:00:00&AccessTimeSchedule[0].TimeSchedule[1][1]=1%2015:00:00-20:00:00		
Response	OK		

## 11.2.12 [Config] Special Day Group Setting

The following Apis are not supported by video talk device.

- Get special day group setting

Table 11-22

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=SpecialDayGroup		
Method	GET		
Description	Get special day group setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object

+SpecialDayGroup	array<object>	R	SpecialDayGroup config object array
++Name	string	O	The special day group name
++Enable	bool	R	Whether to enable this special day group
++Days	array<object>	O	The special days in group
+++SpecialDayName	string	O	The special day name.
+++StartTime	string	O	The special day start datetime
+++EndTime	string	O	The special day end datetime
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=SpecialDayGroup		
Response	table.SpecialDayGroup[0].Name=SpecialDayGroup1 table.SpecialDayGroup[0].Enable=true table.SpecialDayGroup[0].Days[0].SpecialDayName=NationalDay table.SpecialDayGroup[0].Days[0].StartTime=2017-10-01 00:00:00 table.SpecialDayGroup[0].Days[0].EndTime=2017-10-07 23:59:59 ...		

- Set special day group setting

Table 11-23

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set special day group setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
SpecialDayGroup	array<object>	R	The special day group config object array
+Name	string	O	The special day group name
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&SpecialDayGroup[0].Name=SpecialDayGroup1&SpecialDayGroup[0].Enable=true&SpecialDayGroup[0].Days[0].SpecialDayName=NationalDay&SpecialDayGroup[0].Days[0].StartTime=2017-10-01%2000:00:00&SpecialDayGroup[0].Days[0].EndTime=2017-10-07%2023:59:59		
Response	OK		

### 11.2.13 [Config] Special Days Schedule Setting

The following Apis are not supported by video talk device.

- Get special days schedule setting

Table 11-24

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=SpecialDaysSchedule		
Method	GET		
Description	Get special days schedule setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+SpecialDaysSchedule	array<object>	R	SpecialDaysSchedule config object array
++Name	string	O	The special day schedule name
++Enable	bool	R	Whether to enable this special day schedule
++GroupNo	int	R	The special day group number, value is the index of SpecialDayGroup setting array.
++TimeSection	array<string>	R	The time section array, the format of the item is "enable, hour:minute:second – hour:minute:second", for example : "1 00:00:00-12:00:00".
++Doors	array<int>	R	The doors array.
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=SpecialDaysSchedule		
Response	table.SpecialDaysSchedule[0].Name=SpecialDayGroup1 table.SpecialDaysSchedule[0].Enable=true table.SpecialDaysSchedule[0].GroupNo=1 table.SpecialDaysSchedule[0].TimeSection[0]=1 00:00:00-12:00:00 table.SpecialDaysSchedule[0].TimeSection[1]=1 15:00:00-20:00:00 table.SpecialDaysSchedule[0].Doors[0]=2 table.SpecialDaysSchedule[0].Doors[1]=3 ...		

- Set special days schedule setting

Table 11-25

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set special days schedule setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
SpecialDaysSchedule	array<object>	R	The special days schedule config object array
+Name	string	O	The special day schedule name
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&SpecialDaysSchedule[0].		

	Name=SpecialDayGroup1&SpecialDaysSchedule[0].Enable=true&SpecialDaysSchedule[0].GroupNo=1&SpecialDaysSchedule[0].TimeSection[0]=1%2000:00:00-12:00:00&SpecialDaysSchedule[0].TimeSection[1]=1%2015:00:00-20:00:00&SpecialDaysSchedule[0].Doors[0]=2&SpecialDaysSchedule[0].Doors[1]=3
Response	OK

## 11.2.14 [Config] MeasureTemperature Setting

The following Apis are not supported by video talk device.

- Get measure temperature setting

Table 11-26

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=MeasureTemperature		
Method	GET		
Description	Get measure temperature setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+MeasureTemperature	object	R	MeasureTemperature config object
++Enable	bool	R	whether to enable measure temperature function
++MaskOpt	uint	R	mask mode 0: don not detect mask 1: mask prompt 2: mask intercept
++OnlyTemperatureMode	bool	O	whether to enable the single function, temperature monitoring
++TemperatureDisplay	bool	R	whether to show the temperature
++TemperatureUnit	uint	O	temperature unit: 0: centigrade 1: Fahrenheit }
++Type	integer	R	Temperature monitoring method, it can be: 0: Single chip microcomputer infrared temperature monitoring 1: the thermal imaging principle of temperature monitoring 2: Temperature monitoring module of Guide 3: single point temperature monitoring at wrist
++GuideModuleParam	object	O	param of temperature monitoring module of Guide
+++Threshold	float	O	the temperature threshold, unit is °C
+++CalibrationM	uint32	O	Calibration Model, it can be:

odel			0: indoor mode 1: wall mounting mode 2: Gate mode 3: exclusive floor mode
+++Correct	Float	O	temperature correct value, unit is °C
+++DebugModel Enable	Bool	O	whether you enable debug mode of temperature monitoring, show temperature on the top of face
+++HeatDisplay Enable	bool	O	whether to show heat
+++MaxDistance	uint32	O	the max distance for temperature monitoring, unit: cm
+++ProjectDebugModel	bool	O	whether to enable project debug mode
+++RectEnable	Bool	O	whether to show the rectangle of temperature monitoring area on the video.
+++TempRandReplaceThreshold	Float	O	the threshold under which the temperature value will be replaced with valid one. When threshold value is 0, the function won't work.
+++ValidTemperatureLowerLimit	float	O	the lower limit of valid temperature value
++InfraredTemperatureParam	object	O	param of infrared temperature monitoring
+++Correct	float	O	temperature correct value, unit is °C
+++DebugModel Enable	bool	O	whether you enable debug mode of temperature monitoring, show temperature on the top of face
+++MaxDistance	Uint32	O	the max distance for temperature monitoring, unit: cm
+++OverTemperatureMaxDistance	Uint32	O	Maximum distance for reporting high temperature events, In this distance, the high temperature was measured and reported directly. Beyond this distance, the high temperature was measured, it will indicate closing to the point, and then measured again (CM)
+++RectEnable	bool	O	whether to show the rectangle of temperature monitoring area on the video.
+++RetentionTime	Uint32	O	temperature retention Time, unit: ms
+++SensorType	string	O	sensor type in temperature monitoring module "90641", "90640"
+++Threshold	Uint32	O	temperature threshold, unit : °C
+++ValidTemperatureLowerLimit	Flaot	O	the lower limit of valid temperature value
++ThermalImagingParam	object	O	param of thermalimaging
+++RetentionTime	Uint32	O	temperature retention Time, unit: ms
+++Threshold	Uint32	O	temperature threshold, unit : °C
++WristTemperature	Object	O	param of wrist temperature monitoring

tureParam			
+++Correct	Float	O	temperature correct value, unit is °C
+++InvalidTemperatureDistance	Uint32	O	invalid temperature monitoring distance
+++TemperatureTimeout	Uint32	O	timeout of temperature monitoring, unit: s
+++Threshold	Float	O	temperature threshold, unit : °C
+++ValidTemperatureDistance	Uint32	O	valid temperature monitoring distance,
+++ValidTemperatureLowerLimit	flaot	O	the lower limit of valid temperature value

[ Example ]

Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=MeasureTemperature
Response	<pre> table.MeasureTemperature.Enable=true table.MeasureTemperature.GuideModuleParam.CalibrationModel=1 table.MeasureTemperature.GuideModuleParam.Correct=0 table.MeasureTemperature.GuideModuleParam.DebugModelEnable=false table.MeasureTemperature.GuideModuleParam.HeatDisplayEnbale=false table.MeasureTemperature.GuideModuleParam.MaxDistance=0 table.MeasureTemperature.GuideModuleParam.ProjectDebugModel=false table.MeasureTemperature.GuideModuleParam.RectEnable=true table.MeasureTemperature.GuideModuleParam.TempRandReplaceThreshold=0 table.MeasureTemperature.GuideModuleParam.Threshold=37.300000 table.MeasureTemperature.GuideModuleParam.ValidTemperatureLowerLimit=30 table.MeasureTemperature.InfraredTemperatureParam.Correct=0 table.MeasureTemperature.InfraredTemperatureParam.DebugModelEnable=false table.MeasureTemperature.InfraredTemperatureParam.MaxDistance=100 table.MeasureTemperature.InfraredTemperatureParam.OverTemperatureMaxDistance=100 table.MeasureTemperature.InfraredTemperatureParam.RectEnable=true table.MeasureTemperature.InfraredTemperatureParam.RetentionTime=500 table.MeasureTemperature.InfraredTemperatureParam.SensorType= table.MeasureTemperature.InfraredTemperatureParam.Threshold=37.300000 table.MeasureTemperature.InfraredTemperatureParam.ValidTemperatureLowerLimit=30 table.MeasureTemperature.MaskOpt=0 table.MeasureTemperature.OnlyTemperatureMode=false table.MeasureTemperature.TemperatureDisplay=true table.MeasureTemperature.TemperatureUnit=0 table.MeasureTemperature.ThermallmagingParam.RetentionTime=30 table.MeasureTemperature.ThermallmagingParam.Threshold=60 table.MeasureTemperature.Type=3 table.MeasureTemperature.WristTemperatureParam.Correct=0 table.MeasureTemperature.WristTemperatureParam.InvalidTemperatureDistance=24 table.MeasureTemperature.WristTemperatureParam.TemperatureTimeout=10 table.MeasureTemperature.WristTemperatureParam.Threshold=37.300000 table.MeasureTemperature.WristTemperatureParam.ValidTemperatureDistance=5 table.MeasureTemperature.WristTemperatureParam.ValidTemperatureLowerLimit=35 </pre>

...
-----

- Set measure temperature setting

Table 11-27

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set measure temperature setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
MeasureTemperature	object	R	The measure temperature config object
+Name	string	O	The measure temperature name
...<other param>		—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&MeasureTemperature.Enable=true&MeasureTemperature.GuideModuleParam.Threshold=38.5&MeasureTemperature.MaskOpt=2		
Response	OK		

## 11.2.15 [Config] CitizenPictureCompare Setting

- Get citizen picture compare setting

Table 11-28

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=CitizenPictureCompare		
Method	GET		
Description	Get citizen picture compare setting		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	config table object
+CitizenPictureCompare	object	R	CitizenPictureCompare config object
++Threshold	uint8	R	threshold of citizen picture compare, range [1, 100]
++UnlockEnable	bool	R	whether enable unlock if false, it won't unlock, when citizen picture compare succeed.
++FuncEnable	bool	O	whether to enable citizen picture comparison
++CitizenIDCheck	bool	O	is citizen ID check supported

++SysMode	string	O	device work mode, it can be: { "FaceCollect" , "VisitorVerify", "CitizenCompare", "CitizenIDCheck", "Other" }
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=CitizenPictureCompare		
Response	table.CitizenPictureCompare.SysMode=CitizenCompare table.CitizenPictureCompare.Threshold=50 table.CitizenPictureCompare.UnlockEnable=false table.CitizenPictureCompare.FuncEnable=false table.CitizenPictureCompare.CitizenIDCheck=false ...		

- Set citizen picture compare setting

Table 11-29

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set citizen picture compare setting		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
CitizenPictureCompare	object	R	CitizenPictureCompare config object
...<other param>	—	—	...<See above getConfig command for other params of the config object, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&CitizenPictureCompare.SysMode=CitizenIDCheck&CitizenPictureCompare.Threshold=80&CitizenPictureCompare.UnlockEnable=true&CitizenPictureCompare.FuncEnable=true&CitizenPictureCompare.CitizenIDCheck=true		
Response	OK		

## 11.2.16 Sending User Information

Insert or update multiple new users. Update when the current user exists. This only applies to the AccessUser entity here.

Table 11-30

URL	http://<server>/cgi-bin/AccessUser.cgi?action=insertMulti
Method	POST
Description	Insert person information.
[ Request Params ] ( JSON format in body )	

Name	Type	R/O	Param Description
UserList	array<object>	R	User list, with up to 10 entries
+UserID	string	R	User ID
+UserName	string	O	User name
+UserType	uint16	O	0: General user, by default; 1: Blocklist user (report the blocklist event ACBlocklist); 2: Guest user; 3: Patrol user; 4: VIP user; 5: Extend time user
+UseTime	uint16	O	Limit of passing times for guest users
+IsFirstEnter	bool	O	First user authority or not. false: No; true: Yes
+FirstEnterDoors	int16[]	O	-1 indicates all channels.
+UserStatus	uint16	O	0: Normal; 1: Frozen
+Authority	uint8	O	User authority (attendance machine field). 1: Admin; 2: General user
+CitizenIDNo	string	O	ID card number
+Password	string	O	The password when unlocking by card + password. The password when unlocking by User ID + password
+Doors	int16[]	O	Door authority. The index in the controller is used with TimeSections, and the value corresponds to the subscript of the AC configuration.
+TimeSections	uint16[]	O	The door authority corresponds to the period index. For example, door 3 corresponds to period 2. Each element corresponds to the door in Doors.
+SpecialDaysSchedule	uint32[64]	O	Holiday plan identification. The value is the subscript number configured by SpecialDaysSchedule (defined in the configuration).
+ValidFrom	string	O	"yyyy-MM-dd HH:mm:ss", start of validity period. Note: The original "ValidDateStart" is deprecated.
+ValidTo	string	O	"yyyy-MM-dd HH:mm:ss", end of validity period. Note: The original "ValidDateEnd" is deprecated.

[ Response Params ] ( OK )	
[ Example ]	
Request	POST http://192.168.1.108/cgi-bin/AccessUser.cgi?action=insertMulti  <pre> {   UserList[{     "UserID": "100013",     "UserName": "",     "UserType": 0,     "UseTime": 1,     "IsFirstEnter": true,     "FirstEnterDoors": [0, 1],     "UserStatus": 0,     "Authority": 1,     "CitizenIDNo": "123456789012345678",     "Password": "xxxxxxxx",     "Doors": [1,3,5,7],     "TimeSections": [1,2,3,4],     "SpecialDaysSchedule": [1,2],     "ValidFrom": "2018-01-02 00:00:00",     "ValidTo": "2018-01-02 01:00:00",   } ,...,{}] }</pre>
Response	OK

## 11.2.17 Updating User Information

Update information of multiple users.

Table 11-31

URL	http://<server>/cgi-bin/AccessUser.cgi?action=updateMulti		
Method	POST		
Description	Update person information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
UserList	array<object>	R	User list, with up to 10 entries
+ UserID	string	R	User ID
+ UserName	string	O	User name
+ UserType	uint16	O	0: General user, by default; 1: Blocklist user (report the blocklist event ACBlocklist); 2: Guest user; 3: Patrol user; 4: VIP user; 5: Extend time user
+ UseTime	uint16	O	Limit of passing times for guest users

+IsFirstEnter	bool	O	First user authority or not. false: No; true: Yes
+FirstEnterDoors	int16[]	O	-1 indicates all channels.
+UserStatus	uint16	O	0: Normal; 1: Frozen
+Authority	uint8	O	User authority (attendance machine field). 1: Admin; 2: General user
+CitizenIDNo	string	O	ID card number
+Password	string	O	The password when unlocking by card + password. The password when unlocking by User ID + password
+Doors	int16[]	O	Door authority. The index in the controller is used with TimeSections, and the value corresponds to the subscript of the AC configuration.
+TimeSections	uint16[]	O	The door authority corresponds to the period index. For example, door 3 corresponds to period 2. Each element corresponds to the door in Doors.
+SpecialDaysSchedule	uint32[64]	O	Holiday plan identification. The value is the subscript number configured by SpecialDaysSchedule (defined in the configuration).
+ValidFrom	string	O	"yyyy-MM-dd HH:mm:ss", start of validity period. Note: The original "ValidDateStart" is deprecated.
+ValidTo	string	O	"yyyy-MM-dd HH:mm:ss", end of validity period. Note: The original "ValidDateEnd" is deprecated.

[ Response Params ] ( OK )

[ Example ]

Request	<pre>POST http://192.168.1.108/cgi-bin/AccessUser.cgi?action=insertMulti  {   UserList[{     "UserID": "100013",     "UserName": "",     "UserType": 0,     "UseTime": 1,     "IsFirstEnter": true,     "FirstEnterDoors": [0, 1],     "UserStatus": 0,</pre>
---------	---

	<pre>"Authority": 1, "CitizenIDNo": "123456789012345678", &gt;Password": "xxxxxxxx", "Doors": [1,3,5,7], "TimeSections": [1,2,3,4], "SpecialDaysSchedule": [1,2], "ValidFrom": "2018-01-02 00:00:00", "ValidTo": "2018-01-02 01:00:00", },...,{}]</pre>
Response	OK

## 11.2.18 Deleting All User Information

Delete all user information.

Table 11-32

URL	http://<server>/cgi-bin/AccessUser.cgi?action=removeAll		
Method	GET		
Description	Clear information of all persons.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=removeAll		
Response	OK		

## 11.2.19 Deleting Information of Multiple Users

Delete information of multiple users.

Table 11-33

URL	http://<server>/cgi-bin/AccessUser.cgi?action=removeMulti		
Method	GET		
Description	Delete user data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserIDLlist	array<string>	R	User ID list, with up to 10 entries
[ Response Params ] ( OK )			

[ Example ]	
Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=removeMulti&UserIDList[0]=102&UserIDList [1]=102
Response	OK

## 11.2.20 Searching for Information of Multiple Users

Search for information of multiple users.

Table 11-34

URL	http://<server>/cgi-bin/AccessUser.cgi?action=list		
Method	GET		
Description	Search for user data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserIDList	array<string>	R	User ID list, with up to 10 entries
[ Response Params ]			
Users	array<object>	R	The records that returned
+ UserID	string	R	User ID
+ UserName	string	O	User name
+ UserType	uint16	O	0: General user, by default; 1: Blocklist user (report the blocklist event ACBlocklist); 2: Guest user; 3: Patrol user; 4: VIP user; 5: Extend time user
+ UseTime	uint16	O	Limit of passing times for guest users
+ IsFirstEnter	bool	O	First user authority or not. false: No; true: Yes
+ FirstEnterDoors	int16[]	O	-1 indicates all channels
+ UserStatus	uint16	O	0: Normal; 1: Frozen
+ Authority	uint8	O	User authority (attendance machine field). 1: Admin; 2: General user
+ CitizenIDNo	string	O	ID card number
+ Password	string	O	The password when unlocking by card + password. The password when unlocking by UserID + password

+Doors	int16[]	O	Door authority. The index in the controller is used with TimeSections, and the value corresponds to the subscript of the AC configuration.
+TimeSections	uint16[]	O	The door authority corresponds to the period index. For example, door 3 corresponds to period 2. Each element corresponds to the door in Doors.
+SpecialDaysSchedule	uint32[64]	O	Holiday plan identification. The value is the subscript number configured by SpecialDaysSchedule (defined in the configuration).
+ValidFrom	string	O	"yyyy-MM-dd HH:mm:ss", start of validity period. Note: The original "ValidDateStart" is deprecated.
+ValidTo	string	O	"yyyy-MM-dd HH:mm:ss", end of validity period. Note: The original "ValidDateEnd" is deprecated.

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=list&UserIDList[0]=102&UserIDList[1]=102
Response	<pre> Users[0].UserID=100013 Users[0].UserName=Name Users[0].UserType=1 Users[0].UseTime=1 Users[0].IsFirstEnter=ZhangSan Users[0].FirstEnterDoors=0 Users[0].UserStatus=12345678 Users[0].Authority=1 Users[0].CitizenIDNo=1 Users[0].Password=ZhangSan Users[0].Doors=0 Users[0].TimeSections=12345678 Users[0].SpecialDaysSchedule=1 Users[0].ValidFrom=1 records[0].ValidTo=ZhangSan ... </pre>

## 11.2.21 Sending Information of Multiple Card Numbers

Send information of multiple card numbers.

Table 11-35

URL	http://<server>/cgi-bin/AccessCard.cgi?action=insertMulti		
Method	POST		
Description	Insert card number information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
CardList	array<object>	R	User list, with up to 10 entries
+CardNo	string	R	Card No.
+UserID	string	R	User ID
+CardType	uint16	O	Card type Enumint{ 0: Ordinary card 1: VIP card 2: Guest card 3: Patrol card 4: Blocklist card 5: Duress card
+CardName	string	O	Card name
+CardStatus	uint32	O	Card status. Different card status results in different person status. 0: Normal 1<<0: Reported for loss 1<<1: Canceled 1<<2: Frozen 1<<3: Arrearage 1<<4: Overdue
[ Response Params ] ( OK )			
[ Example ]			
Request	POST http://192.168.1.108/cgi-bin/AccessCard.cgi?action=insertMulti  <pre>{   CardList[{     "UserID" : "100013"     "CardNo" : ""     "CardType" : 0     "CardName" : "201 - Xiao Wang"     "CardStatus" : 0   } ,...,{}] }</pre>		
Response	OK		

## 11.2.22 Updating Information of Multiple Card Numbers

Update information of multiple card numbers.

Table 11-36

URL	http://<server>/cgi-bin/AccessCard.cgi?action=updateMulti		
Method	POST		
Description	Update card number information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
CardList	array<object>	R	User list, with up to 10 entries
+CardNo	string	R	Card No.
+UserID	string	R	User ID
+CardType	uint16	O	Card type Enumint{ 0: Ordinary card 1: VIP card 2: Guest card 3: Patrol card 4: Blocklist card 5: Duress card
+CardName	string	O	Card name
+CardStatus	uint32	O	Card status. Different card status results in different person status. 0: Normal 1<<0: Reported for loss 1<<1: Canceled 1<<2: Frozen 1<<3: Arrearage 1<<4: Overdue
[ Response Params ] ( OK )			
[ Example ]			
Request	POST ttp://192.168.1.108/cgi-bin/AccessCard.cgi?action=updateMulti  <pre>{   CardList[{     "UserID" : "100013"     "CardNo" : ""     "CardType" : 0     "CardName" : "201 - Xiao Wang"     "CardStatus" : 0   } ,...,{}] }</pre>		
Response	OK		

## 11.2.23 Removing All Card Number Information

Remove all card number information.

Table 11-37

URL	http://<server>/cgi-bin/AccessCard.cgi?action=removeAll		
Method	GET		
Description	Clear all card information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=removeAll		
Response	OK		

## 11.2.24 Removing Information of Multiple Card Numbers

Remove information of multiple card numbers.

Table 11-38

URL	http://<server>/cgi-bin/AccessCard.cgi?action=removeMulti		
Method	GET		
Description	Delete card number data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
CardNoList	array<string>	R	User card number list, with up to 10 entries
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=removeMulti&CardNoList[0]=12345678&CardNoList[1]=12345687		
Response	OK		

## 11.2.25 Searching for Information of Multiple Card Numbers

Search for information of multiple card numbers.

Table 11-39

URL	http://<server>/cgi-bin/AccessCard.cgi?action=list		
Method	GET		
Description	Search for card number data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
CardNoList	array<string>	R	User ID list, with up to 10 entries
[ Response Params ]			
records	array<object>	R	The records that returned.
+CardNo	string	R	Card No.
+UserID	string	R	User ID
+CardType	uint16	O	Card type Enumint{ 0: Ordinary card 1: VIP card 2: Guest card 3: Patrol card 4: Blocklist card 5: Duress card
+CardName	string	O	Card name
+CardStatus	uint32	O	Card status. Different card status results in different person status. 0: Normal 1<<0: Reported for loss 1<<1: Canceled 1<<2: Frozen 1<<3: Arrearage 1<<4: Overdue
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=list&CardNoList[0]=102&CardNoList[1]=102		
Response	Cards[0].CardNo=12345678 Cards[0].UserID=1 Cards[0].CardType=1 Cards[0].CardName=ZhangSan Cards[0].CardStatus=0 ...		

Cards[1].CardNo=12345679 Cards[1].UserID=2 Cards[1].CardType=1 Cards[1].CardName=LiSi Cards[1].CardStatus=0 ...
--

## 11.2.26 Sending Face Information

Insert multiple face information.

Table 11-40

URL	http://<server>/cgi-bin/AccessFace.cgi?action=insertMulti		
Method	POST		
Description	Insert face information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
FaceList	array<object>	R	Face list, with up to 10 entries
+UserID	string	R	User ID
+FaceData	array<string>	O	Base64 of red light face templates, 8192 Bytes/per face, maximum 20.
+PhotoData	array<string>	O	Base64 of white light face images, 200kx5.
+PhotoURL	array<string>	O	Either cloud storage URL of white light face images or PhotoData. If PhotoData exists, this field is invalid when it is subject to PhotoData. Only HTTP URL is available now.
[ Response Params ] ( OK )			
[ Example ]			
Request	<pre>POST http://192.168.1.108/cgi-bin/AccessFace.cgi?action=insertMulti  {   "FaceList":[     {       "UserID": "102",       "FaceData": [ "xxxx", "xxxx", ... ],       "PhotoData": [ "yyyy", "yyyy", ... ],       "PhotoURL": [ "yyyy", "yyyy", ... ],     },     ....   ] }</pre>		
Response	OK		

## 11.2.27 Updating Face Information

Update multiple face information.

Table 11-41

URL	http://<server>/cgi-bin/AccessFace.cgi?action=updateMulti		
Method	POST		
Description	Update face information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
FaceList	array<object>	R	Face list, with up to 10 entries
+UserID	string	R	User ID
+FaceData	array<string>	O	Base64 of red light face templates, 8192 Bytes/per face, maximum 20.
+PhotoData	array<string>	O	Base64 of white light face images, 200kx5.
+PhotoURL	array<string>	O	Either cloud storage URL of white light face images or PhotoData. If PhotoData exists, this field is invalid when it is subject to PhotoData. Only HTTP URL is available now.
[ Response Params ] ( OK )			
[ Example ]			
Request	POST <a href="http://192.168.1.108/cgi-bin/AccessFace.cgi?action=updateMulti">http://192.168.1.108/cgi-bin/AccessFace.cgi?action=updateMulti</a> <pre> {   "FaceList":[     {       "UserID": "102",       "FaceData": [ "xxxx", "xxxx", ... ],       "PhotoData": [ "yyyy", "yyyy", ... ],       "PhotoURL": [ "yyyy", "yyyy", ... ],     },     ....   ] }           </pre>		
Response	OK		

## 11.2.28 Deleting All Face Information

Delete all face information.

Table 11-42

URL	http://<server>/cgi-bin/AccessFace.cgi?action=removeAll
Method	GET

Description	Clear all face information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=removeAll		
Response	OK		

## 11.2.29 Deleting Multiple Face Information

Delete multiple face information.

Table 11-43

URL	http://<server>/cgi-bin/AccessFace.cgi?action=removeMulti		
Method	GET		
Description	Delete face data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserIDLlist	array<string>	R	User card number list, with up to 10 entries
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=removeMulti&UserIDLlist[0]=101&UserIDLlist[1]=102		
Response	OK		

## 11.2.30 Searching for Multiple Face Information

Search for multiple face information.

Table 11-44

URL	http://<server>/cgi-bin/AccessFace.cgi?action=list
Method	GET
Description	Search for user face data.

[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserIDList	array<string>	R	User ID list, with up to 10 entries
[ Response Params ]			
FaceDataList	array<object>	R	The records that returned.
+UserID	string	R	User ID
+FaceData	array<string>	O	Base64 of red light face templates, 8192 Bytes/per face, maximum 20.
+PhotoData	array<string>	O	Base64 of white light face images, 200kx5.
+PhotoURL	array<string>	O	Either cloud storage URL of white light face images or PhotoData. If PhotoData exists, this field is invalid when it is subject to PhotoData. Only HTTP URL is available now.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=list&UserIDList[0]=1&UserIDList[1]=2		
Response	FaceDataList[0].UserID=1 FaceDataList[0].PhotoData=[ "xxxx", "xxxx", ... ], FaceDataList[0].FaceData=[ "xxxx", "xxxx", ... ], ... FaceDataList[1].UserID=2 FaceDataList[1].PhotoData=[ "xxxx", "xxxx", ... ], FaceDataList[1].FaceData=[ "xxxx", "xxxx", ... ], ...		

### 11.2.31 Sending Fingerprint Information

Insert multiple fingerprint information.

Table 11-45

URL	http://<server>/cgi-bin/AccessFingerprint.cgi?action=insertMulti		
Method	POST		
Description	Insert fingerprint information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
vecPackets	array<string>	R	Fingerprint feature data

AccessFingerprints	array<object>	R	Fingerprint list, with up to 1 entry
+UserID	string	R	User ID
+FingerprintPacket	object	R	Send fingerprint information list.
++Length	uint32	R	Length of a single fingerprint package
++Count	uint32	R	Number of fingerprint packages
++DataURL	array<string>	O	Either cloud storage URL of fingerprint data or Length. If the Length exists and is not 0, this field is invalid when it is subject to Length. Only HTTP URL is available now.
++DuressIndex	Uint8	R	Duress fingerprint number, with a value range of [1, count]. This field is invalid if the value is illegal. That is, there is no duress fingerprint. For example, 0 indicates no duress fingerprint.

[ Response Params ] ( OK )

[ Example ]

Request	<pre>POST http://192.168.1.108/cgi-bin/AccessFingerprint.cgi?action=insertMulti {   "vecPackets": [ "xxxx", "xxxx", ... ],   "AccessFingerprints": [     {       "UserID": "102",       "FingerprintPacket" :         {           "Length" : 810,           "Count" : 3,           "DuressIndex" : 2         }     }   ] }</pre>
Response	OK

## 11.2.32 Updating Fingerprint Information

Update multiple fingerprint information.

Table 11-46

URL	http://<server>/cgi-bin/AccessFingerprint.cgi?action=updateMulti		
Method	POST		
Description	Update fingerprint information.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
vecPackets	array<string>	R	Binary fingerprint data (Binary Data).

AccessFingerprints	array<object>	R	Fingerprint list, with up to 1 entry
+UserID	string	R	User ID
+FingerprintPacket	object	R	Send fingerprint information list.
++Length	uint32	R	Length of a single fingerprint package
++Count	uint32	R	Number of fingerprint packages
++DataURL	array<string>	O	Either cloud storage URL of fingerprint data or Length. If the Length exists and is not 0, this field is invalid when it is subject to Length. Only HTTP URL is available now.
++DuressIndex	Uint8	R	Duress fingerprint number, with a value range of [1, count]. This field is invalid if the value is illegal. That is, there is no duress fingerprint. For example, 0 indicates no duress fingerprint.

[ Response Params ] ( OK )

[ Example ]

Request	<pre>POST http://192.168.1.108/cgi-bin/AccessFingerprint.cgi?action=updateMulti {   "vecPackets": [ "xxxx", "xxxx", ... ],   "AccessFingerprints": [     {       "UserID": "102",       "FingerprintPacket" :         {           "Length" : 810,           "Count" : 3,           "DuressIndex" : 2         }     }   ] }</pre>
Response	OK

### 11.2.33 Deleting All Fingerprint Information

Delete all fingerprint information.

Table 11-47

URL	http://<server>/cgi-bin/AccessFingerprint.cgi?action=removeAll		
Method	GET		
Description	Clear all fingerprint information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
[ Response Params ] ( OK )			

[ Example ]	
Request	GET http://192.168.1.108/cgi-bin/AccessFingerprint.cgi?action=removeAll
Response	OK

## 11.2.34 Deleting Fingerprint Information

Delete fingerprint information.

Table 11-48

URL	http://<server>/cgi-bin/AccessFingerprint.cgi?action=removeMulti		
Method	GET		
Description	Delete fingerprint data.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserIDList	array<string>	R	User card number list, with up to 1 entry
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFingerprint.cgi?action=removeMulti&UserIDList[0]=101&UserIDList[1]=102		
Response	OK		

## 11.2.35 Searching for Fingerprint Information

Search for user fingerprint information.

Table 11-49

URL	http://<server>/cgi-bin/AccessFingerprint.cgi?action=get		
Method	GET		
Description	Search for user fingerprint information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UserID	string	R	User ID list, with up to 1 entry
[ Response Params ]			

FingerprintData	BinaryData	R	Binary fingerprint data (Binary Data).
FingerprintPacket	object	R	Description of fingerprint data
+Length	uint32	R	Length of a single fingerprint package
+Count	uint32	R	Number of fingerprint packages
+DataURL	array<string>	O	Either cloud storage URL of fingerprint data or Length. If the Length exists and is not 0, this field is invalid when it is subject to Length. Only HTTP URL is available now.
+DuressIndex	UInt8	R	Duress fingerprint number, with a value range of [1, count]. This field is invalid if the value is illegal. That is, there is no duress fingerprint. For example, 0 indicates no duress fingerprint.

[ Example ]	
Request	GET http://192.168.1.108/cgi-bin/AccessFingerprint.cgi?action=get&UserID=1
Response	FingerprintPacket.Length=810 FingerprintPacket.Count=3 FingerprintPacket.DuressIndex=1 FingerprintData=xxx

## 11.2.36 Starting Searching for User Related Information

Start searching for user related information.

Table 11-50

URL	http://<server>/cgi-bin/AccessUser.cgi?action=startFind		
Method	GET		
Description	Start searching for user related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Condition	object	R	Search conditions. Users can perform conditional search according to the user information field.
+UserID	string	O	User ID

+UserName	string	O	User name
+UserType	uint16	O	0: General user, by default; 1: Blocklist user (report the blocklist event ACBlocklist); 2: Guest user; 3: Patrol user; 4: VIP user; 5: Extend time user
+UseTime	uint16	O	Limit of passing times for guest users
+IsFirstEnter	bool	O	First user authority or not. false: No; true: Yes
+FirstEnterDoors	int16[]	O	-1 indicates all channels.
+UserStatus	uint16	O	0: Normal; 1: Frozen
+Authority	uint8	O	User authority (attendance machine field). 1: Admin; 2: General user
+CitizenIDNo	string	O	ID card number
+Password	string	O	The password when unlocking by card + password. The password when unlocking by UserID + password
+Doors	int16[]	O	Door authority. The index in the controller is used with TimeSections, and the value corresponds to the subscript of the AC configuration.
+TimeSections	uint16[]	O	The door authority corresponds to the period index. For example, door 3 corresponds to period 2. Each element corresponds to the door in Doors.
+SpecialDaysSchedule	uint32[64]	O	Holiday plan identification. The value is the subscript number configured by SpecialDaysSchedule (defined in the configuration).
+ValidFrom	string	O	"yyyy-MM-dd HH:mm:ss", start of validity period. Note: The original "ValidDateStart" is deprecated.
+ValidTo	string	O	"yyyy-MM-dd HH:mm:ss", end of validity period. Note: The original "ValidDateEnd" is deprecated.
<b>[ Response Params ]</b>			
Token	uint32	R	Search token.
Total	uint32	R	Total number of entries found this time
Caps	uint32	R	Search capability: Maximum

		number of records that can be returned each time.
[ Example ]		
Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=startFind&Condition.UserID=1	
Response	<pre>{   "Token": 1234,   "Total": 20,   "Caps ": 20 }</pre>	

## 11.2.37 Getting User Related Information

Get user related information.

Table 11-51

URL	http://<server>/cgi-bin/AccessUser.cgi?action=doFind		
Method	GET		
Description	Get user related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Search token
Offset	int	R	Offset
Count	int	R	Number of entries obtained this time
[ Response Params ]			
info	array<object>	R	Person information
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=doFind&Token=1234&Offset=0&Count=20		
Response	<pre>{ "Info": [ {</pre>		

```

"UserID": "102",
...
}, { ... }, ... ]
}

```

### 11.2.38 Stopping Searching for User Related Information

Get user related information.

Table 11-52

URL	http://<server>/cgi-bin/AccessUser.cgi?action=stopFind		
Method	GET		
Description	Get user related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Search token.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessUser.cgi?action=stopFind&Token=1234		
Response	OK		

### 11.2.39 Starting Searching for Card Number Related Information

Start searching for card number related information.

Table 11-53

URL	http://<server>/cgi-bin/ AccessCard.cgi?action=startFind		
Method	GET		
Description	Start searching for card number related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Condition	object	R	Search conditions. Users can perform conditional search according to the user information

			field.
+UserID	string	O	User ID
+CardNo	string	O	User card number
+CardType	uint16	O	Card type Enumint{ 0: Ordinary card 1: VIP card 2: Guest card 3: Patrol card 4: Blocklist card 5: Duress card
+CardName	string	O	Card name
+CardStatus	uint32	O	Card status Different card status results in different person status. 0: Normal 1<<0: Reported for loss 1<<1: Canceled 1<<2: Frozen 1<<3: Arrearage 1<<4: Overdue
[ Response Params ]			
Token	uint32	R	Search token.
Total	uint32	R	Total number of entries found this time
Caps	uint32	R	Search capability: Maximum number of records that can be returned each time.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=startFind&Condition.UserID=1		
Response	<pre>{   "Token": 1234,   "Total": 20,   "Caps ": 20 }</pre>		

## 11.2.40 Getting Card Number Related Information

Get card number related information.

Table 11-54

URL	http://<server>/cgi-bin/AccessCard.cgi?action=doFind		
Method	GET		
Description	Get card number related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Search token.
Offset	int	R	Offset
Count	int	R	Number of entries obtained this time
[ Response Params ]			
info	array<object>	R	Person information
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=doFind&Token=1234&Offset=0&Count=20		
Response	<pre>{ "Info": [ {   "UserID": "102",   ... }, { ... }, ... ] }</pre>		

## 11.2.41 Stopping Searching for Card Number Related Information

Stop searching for card number related information.

Table 11-55

URL	http://<server>/cgi-bin/AccessCard.cgi?action=stopFind		
Method	GET		
Description	Stop searching for card number related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description

Token	int	R	Search token.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessCard.cgi?action=stopFind&Token=1234		
Response	OK		

## 11.2.42 Starting Searching for Face Related Information

Start searching for face related information.

Table 11-56

URL	http://<server>/cgi-bin/AccessFace.cgi?action=startFind		
Method	GET		
Description	Start searching for face related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Condition	object	R	Search conditions. Users can perform conditional search according to the user information field.
+UserID	string	O	User ID
[ Response Params ]			
Token	uint32	R	Search token.
Total	uint32	R	Total number of entries found this time
Caps	uint32	R	Search capability: Maximum number of records that can be returned each time.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=startFind&Condition.UserID=1		
Response	<pre>{   "Token": 1234,</pre>		

<pre> "Total": 20, "Caps ": 20 } </pre>
---

### 11.2.43 Getting Face Related Information

Get face related information.

Table 11-57

URL	http://<server>/cgi-bin/AccessFace.cgi?action=doFind		
Method	GET		
Description	Get face related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Search token.
Offset	int	R	Offset
Count	int	R	Number of entries obtained this time
[ Response Params ]			
info	array<object>	R	Person information
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=doFind&Token=1234&Offset=0&Count=20		
Response	<pre> { "Info": [ {   "UserID": "102",   ... }, { ... }, ... ] } </pre>		

### 11.2.44 Stopping Searching for Face Related Information

Stop searching for face related information.

Table 11-58

URL	http://<server>/cgi-bin/AccessFace.cgi?action=stopFind		
Method	GET		
Description	Stop searching for face related information.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Token	int	R	Search token.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/AccessFace.cgi?action=stopFind&Token=1234		
Response	OK		

### 11.2.45 Get access control capability

Table 11-59

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getCaps
Method	<b>GET</b>
Description	Get access control capability.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getCaps
Success Return	caps.AccessControlChannels=5
Comment	<p><b>Params in Response :</b></p> <p><b>AccessControlChannels:</b> the max channel count supported by control</p>

### 11.2.46 Add SubController

Table 11-60

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=addDevice&devAddr=<devAddr>&name=<name>
Method	<b>GET</b>
Description	Add SubController.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=addDevice&devAddr=1

	&name=Sub1
Success Return	FaiedCode=0
Comment	<p><b>Params in URL:</b></p> <p><b>devAddr:</b> SubController ID, range is [1...255]</p> <p><b>name:</b> SubController Name</p> <p><b>Params in Response:</b></p> <p><b>FaiedCode:</b> 0-Success; 1-Already exists; 2-Reach quantity limit; 3-Other reason</p>

## 11.2.47 Modify SubController

Table 11-61

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=modifyDevice&devAddr=<devAddr>&name=<name>
Method	<b>GET</b>
Description	Modify SubController.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=modifyDevice&devAddr=1&name=Sub11
Success Return	OK
Comment	<p><b>Params in URL:</b></p> <p><b>devAddr:</b> SubController ID, range is [1...255]</p> <p><b>name:</b> SubController Name</p>

## 11.2.48 Remove SubController

Table 11-62

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=removeDevice&devAddrs[0]=<devAddr0>[& devAddrs[1]=<devAddr1>...&devAddrs[n]=<devAddrn>]
Method	<b>GET</b>
Description	Remove SubController.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=removeDevice&devAddrs[0]=4&devAddrs[1]=5
Success Return	devInfo[0].DevAddr=4 devInfo[0].FaiedCode=0 devInfo[1].DevAddr=5

	devInfo[1].FaiedCode=1
Comment	<p><b>Params in URL:</b></p> <p><b>devAddr0...devAddrn:</b> SubController ID, range is [1...255]</p> <p><b>Params in Response:</b></p> <p><b>DevAddr:</b> the removed SubController ID</p> <p><b>FaiedCode:</b> 0-Success; 1-Fail; 2-Other</p>

## 11.2.49 Get SubController Info

Table 11-63

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getSubControllerInfo&devAd drs[0]=<devAddr0>[& devAdrs[1]=<devAddr1>&...&devAdrs[n]=<devAddrn>]
Method	<b>GET</b>
Description	Get SubController Info.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getSubControllerInfo&d evAdrs[0]=4&devAdrs[1]=5
Success Return	devInfo[0].DevAddr=4 devInfo[0].Name=Sub4 devInfo[0].AccessProperty=unidirect devInfo[0].DeviceType=ACS-1202B-S devInfo[0].VerSion=V1.00428 devInfo[0].DoorToReadID[0].Door=0 devInfo[0].DoorToReadID[0].ReaderID[0]=1 devInfo[0].DoorToReadID[0].ReaderID[1]=2 devInfo[0].DoorToReadID[1].Door=1 devInfo[0].DoorToReadID[1].ReaderID[0]=3 devInfo[0].DoorToReadID[1].ReaderID[1]=4 devInfo[1].DevAddr=5 devInfo[1].Name=Sub5 devInfo[1].AccessProperty=bidirect devInfo[1].DeviceType=ACS-1202B-S devInfo[1].VerSion=V1.00428 devInfo[1].DoorToReadID[0].Door=0 devInfo[1].DoorToReadID[0].ReaderID[0]=11 devInfo[1].DoorToReadID[0].ReaderID[1]=12
Comment	<p><b>Params in URL:</b></p> <p><b>devAddr0...devAddrn:</b> SubController ID, range is [1...255]</p> <p><b>Params in Response:</b></p> <p><b>DevAddr:</b> SubController ID, range is [1...255]</p>

	<b>AccessProperty:</b> range is [unidirect, bidirect] <b>DoorToReadID:</b> ReaderID used by the door
--	---

## 11.2.50 Get SubController States

Table 11-64

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getSubControllerStates&devAddrs[0]=<devAddr0>[& devAddrs[1]=<devAddr1>&...&devAddrs[n]=<devAddrn>]
Method	<b>GET</b>
Description	Get SubController States.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getSubControllerStates&devAddrs[0]=4&devAddrs[1]=5
Success Return	devStates[0].DevAddr=4 devStates[0].State=0 devStates[1].DevAddr=5 devStates[1].State=1
Comment	<b>Params in URL:</b> <b>devAddr0...devAddrn:</b> SubController ID, range is [1...255] <b>Params in Response:</b> <b>DevAddr:</b> SubController ID, range is [1...255] <b>State:</b> 0-Offline; 1-Online; 2-Other

## 11.2.51 Set RepeatEnter Route

Table 11-65

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=setRepeatEnterRoute&index=<index>&routeInfo.PointInfo[x0].ReaderID[0]=<ReadID0>&routeInfo.PointInfo[x0].ReaderID[1]=<ReadID1>[&routeInfo.PointInfo[x1].ReaderID[0]=<ReadID0>&routeInfo.PointInfo[x1].ReaderID[1]=<ReadID1>]&routeInfo.TimeSections=<TimeIndex>&routeInfo.Flag=<Flag>&routeInfo.Name=<Name>
Method	<b>GET</b>
Description	Set RepeatEnter Route.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=setRepeatEnterRoute&index=0&routeInfo.PointInfo[0].ReaderID[0]=11&routeInfo.PointInfo[0].ReaderID[1]=22&routeInfo.PointInfo[1].ReaderID[0]=33&routeInfo.PointInfo[1].ReaderID[1]=44&routeInfo.TimeSections=1&routeInfo.Flag=1&routeInfo.Name=123
Success	OK

Return	
Comment	<p><b>Params in URL:</b></p> <p><b>index:</b> route index</p> <p><b>x0,x1:</b> PointInfo index</p> <p><b>TimeIndex:</b> subscript of AccessTimeSchedule</p> <p><b>Flag:</b> 0-Disable this route; 1-Enable this route</p> <p><b>Name:</b> route name</p>

## 11.2.52 Get RepeatEnter Route

Table 11-66

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getRepeatEnterRoute&index=<index>
Method	<b>GET</b>
Description	Get RepeatEnter Route.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getRepeatEnterRoute&index=0
Success Return	<pre>routeInfo.PointInfo[0].ReaderID[0]=11 routeInfo.PointInfo[0].ReaderID[1]=22 routeInfo.PointInfo[1].ReaderID[0]=33 routeInfo.PointInfo[1].ReaderID[1]=44 routeInfo.TimeSections=1 routeInfo.Flag=1 routeInfo.Name=123</pre>
Comment	<p><b>Params in URL:</b></p> <p><b>index:</b> route index</p> <p><b>Params in Response:</b></p> <p><b>PointInfo:</b> is an array, stands for points in route</p> <p><b>TimeSections:</b> subscript of AccessTimeSchedule</p> <p><b>Flag:</b> 0-Disable this route; 1-Enable this route</p> <p><b>Name:</b> route name</p>

## 11.2.53 Set ABLOCK Route

Table 11-67

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=setABLOCKRoute&index=<index>&routeInfo.PointInfo[x0].Doors[0]=<DoorID0>&routeInfo.PointInfo[x0].Doors[1]=<
--------	--

	<b>DoorID1</b> >[&routeInfo.PointInfo[x1].Doors[0]=< <b>DoorID0</b> >&routeInfo.PointInfo[x1].Doors [1]=< <b>DoorID1</b> >]&routeInfo.TimeSections=< <b>TimeIndex</b> >&routeInfo.Flag=< <b>Flag</b> >&routeInfo.Name=< <b>Name</b> >
Method	<b>GET</b>
Description	Set ABLock Route.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=setABLockRoute&index =0&routeInfo.PointInfo[0].Doors[0]=11&routeInfo.PointInfo[0].Doors[1]=22&routeInfo.P ointInfo[1].Doors[0]=33&routeInfo.PointInfo[1].Doors[1]=44&routeInfo.TimeSections=1 &routeInfo.Flag=1&routeInfo.Name=123
Success Return	OK
Comment	<p><b>Params in URL:</b></p> <p><b>index:</b> route index</p> <p><b>x0,x1:</b> PointInfo index</p> <p><b>TimeIndex:</b> subscript of AccessTimeSchedule</p> <p><b>Flag:</b> 0-Disable this route; 1-Enable this route</p> <p><b>Name:</b> route name</p>

## 11.2.54 Get ABLock Route

Table 11-68

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getABLockRoute&index=< <b>index</b> >
Method	<b>GET</b>
Description	Get ABLock Route.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getABLockRoute&index =0
Success Return	routeInfo.PointInfo[0].Doors[0]=11 routeInfo.PointInfo[0].Doors [1]=22 routeInfo.PointInfo[1].Doors [0]=33 routeInfo.PointInfo[1].Doors [1]=44 routeInfo.TimeSections=1 routeInfo.Flag=1 routeInfo.Name=123
Comment	<p><b>Params in URL:</b></p> <p><b>index:</b> route index</p> <p><b>Params in Response:</b></p> <p><b>PointInfo:</b> is an array, stands for points in route</p>

	<b>TimeSections:</b> subscript of AccessTimeSchedule <b>Flag:</b> 0-Disable this route; 1-Enable this route <b>Name:</b> route name
--	---

## 11.2.55 Get Log Status

Table 11-69

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=getLogStatus&devAddr=<devAddr>
Method	<b>GET</b>
Description	Get Log Status.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=getLogStatus&devAddr=1
Success Return	OK
Comment	<b>Params in URL:</b> <b>devAddr:</b> SubController ID, range is [1...255]

## 11.2.56 Sync Offline Log

Table 11-70

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=syncOfflineLog&devAddr=<devAddr>
Method	<b>GET</b>
Description	Sync Offline Log.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=syncOfflineLog&devAddr=1
Success Return	OK
Comment	<b>Params in URL:</b> <b>devAddr:</b> SubController ID, range is [1...255]

## 11.2.57 Sync SubController Time

Table 11-71

Syntax	http://<server>/cgi-bin/accessControlManager.cgi?action=syncSubControllerTime&devAddrs[0]=<devAddr0>[& devAddrs[1]=<devAddr1>&...&devAddrs[n]=<devAddrn>]
Method	<b>GET</b>
Description	Sync SubController Time.
Example	http://192.168.1.108/cgi-bin/accessControlManager.cgi?action=syncSubControllerTime&devAddrs[0]=4&devAddrs[1]=5
Success Return	errorDevAddrs[0]=4
Comment	<b>Params in URL:</b> <b>devAddr0...devAddrn:</b> SubController ID, range is [1...255] <b>Params in Response:</b> <b>errorDevAddrs:</b> is an array, stands for DevAddr which failed in sync time

# 12

## Intelligent Building APIs

### 12.1 Video Talk

The following Apis are supported by VTT / VTA products.

#### 12.1.1 Subscribe Video Talk Status

Table 12-1

Syntax	<code>http://&lt;server&gt;/cgi-bin/VideoTalkPeer.cgi?action=attachState</code>
Method	GET
Description	Subscribe the video talk status. When client disconnect, it will unsubscribe.
Example	<code>http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=attachState</code>
Success Return	Notify the state:  SID=315 state.State=Answer state.Talkback.Pack=RTP state.Talkback.Protocol=UDP state.Talkback.Type=Talk state.Talkback.Audio.AudioPort=6000 state.Talkback.Audio.Format[0].Compression=PCM state.Talkback.Audio.Format[0].Frequency=44000 state.Talkback.Audio.Format[0].Depth=16 state.Talkback.Audio.Format[1].Compression=G.711A state.Talkback.Audio.Format[1].Frequency=44000 state.Talkback.Audio.Format[1].Depth=16 state.Talkback.Video.VideoPort=7000 state.Talkback.Video.Format[0].Compression=H.264 state.Talkback.Video.Format[0].Frequency=90000 state.Talkback.Video.Format[1].Compression=MJPEG  ... state.Talkback. MediaAddr=224.10.10.10
Comment	Parameters in Response: <b>State:</b> in range of {"Ringing", "Inviting", "Answer", "Refuse", "Cancel", "Hangup", "Busying" }

#### 12.1.2 Unsubscribe Video Talk Status

Table 12-2

Syntax	<code>http://&lt;server&gt;/cgi-bin/VideoTalkPeer.cgi?action=detachState&amp;SID=&lt;sid&gt;</code>
--------	---

Method	GET
Description	Unsubscribe the video talk status.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=detachState&SID=101
Success Return	OK
Comment	Parameters in URL: <b>sid</b> : the subscribe id, which is the response of attachState

### 12.1.3 Invite Server on Video Talk

Table 12-3

Syntax	http://<server>/cgi-bin/VideoTalkPeer.cgi?action=invite[&Talkback.Protocol=<protocol>&Talkback.Type=<type>&Talkback.MediaAddr=<addr>...]
Method	GET
Description	Start the video talk conversation.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=invite&Talkback.Protocol=UDP&Talkback.Type=Talk&Talkback.MediaAddr=224.10.10.10
Success Return	OK
Comment	Parameters in URL: <b>protocol</b> : the transmit protocol <b>type</b> : video talk type. <b>addr</b> : addr to get stream

### 12.1.4 Cancel the Video Talk

Table 12-4

Syntax	http://<server>/cgi-bin/VideoTalkPeer.cgi?action=cancel
Method	GET
Description	Cancel video talk conversation.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=cancel
Success Return	OK
Comment	—

### 12.1.5 Answer the Invitation

Table 12-5

Syntax	http://<server>/cgi-bin/VideoTalkPeer.cgi?action=answer&Talkback.Protocol=<protocol>&Talkback.Type=<type>&Talkback.MediaAddr=<addr>...
Method	GET
Description	Answer the call.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=answer&Talkback.Protocol=UDP

	&Talkback.Type=Talk&Talkback.MediaAddr=224.10.10.10
Success Return	OK
Comment	Parameters in URL: <b>protocol</b> : the transmit protocol <b>type</b> : video talk type. <b>addr</b> : addr to get stream

## 12.1.6 Refuse to Answer the Video Talk Invitation

Table 12-6

Syntax	http://<server>/cgi-bin/VideoTalkPeer.cgi?action=refuse
Method	GET
Description	Refuse answer the call.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=refuse
Success Return	OK
Comment	—

## 12.1.7 Hang Up

Table 12-7

Syntax	http://<server>/cgi-bin/VideoTalkPeer.cgi?action=hangup
Method	GET
Description	Close it when the conversation is over.
Example	http://192.168.1.108/cgi-bin/VideoTalkPeer.cgi?action=hangup
Success Return	OK
Comment	—

## 12.2 Video Talk Log

The following APIs are supported by VTO products.

### 12.2.1 Query Video Talk Log

Table 12-8

Syntax	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=VideoTalkLog[&condition.CallType=< <b>Type</b> >&condition.EndState=< <b>State</b> >&count=< <b>countNo</b> >]
Method	GET
Description	Find the VideoTalkLog record.
Example	http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=VideoTalkLog&condition.CallType=Incoming&condition.EndState=Missed&count=500

Success Return	<b>totalCount</b> =1000 <b>found</b> =500 records[0].RecNo=789 records[0].CreateTime=123456789 records[0]. <b>CallType</b> =Incoming records[0]. <b>EndState</b> =Received records[0].PeerNumber=501 ...
Comment	Parameters in URL: <b>Type</b> : call type <b>State</b> : end state of the call <b>countNo</b> : the number of records to get  Parameters in Response : <b>totalCount</b> : the record count which match condition <b>found</b> : the record count to return <b>CallType</b> : call type. The range is {"Incoming", "Outgoing"}. <b>EndState</b> : the range is {"Missed" , "Received"}

## 12.3 Announcement Record

The following Apis are supported by VTO products.

### 12.3.1 Insert Record

Table 12-9

Syntax	http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=Announcement&Content=< <b>Content</b> >&ExpirTime=< <b>ExpirTime</b> >&IssueTime=< <b>IssueTime</b> >&Title=< <b>Title</b> >&User=< <b>User</b> >&State=< <b>State</b> >&ReadFlag=< <b>ReadFlag</b> >
Method	GET
Description	Insert the Announcement record.
Example	http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=insert&name=Announcement&Content=stringData&ExpirTime=2012-01-01%2012:00:00&IssueTime=2012-01-01%2012:00:00&Title=Announce1&User=101&State=0&ReadFlag=0
Success Return	RecNo=< <b>RecNo</b> >
Comment	Parameters in URL: <b>Content</b> : Announcement Content <b>ExpirTime</b> : the time when the Announcement expire, format: 2012-01-01%2012:00:00 <b>IssueTime</b> : Announcement issue time, format: 2012-01-01%2012:00:00 <b>Title</b> : title of the announcement <b>User</b> : the number the Announcement issued to <b>State</b> : the state of the Announcement. 0 init, 1 send , 2 overdue <b>ReadFlag</b> : the read flag. 0 not read, 1 read. Parameters in Response : <b>RecNo</b> : the record index of the new record

## 12.4 Alarm Record

### 12.4.1 Query Alarm Record

Table 12-10

Syntax	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=AlarmRecord[&StartTime=< <b>startTime</b> >&EndTime=< <b>endTime</b> >&count=< <b>countNo</b> >]
Method	GET
Description	Find the AlarmRecord record. ( This api is supported by video talk device. )
Example	http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=AlarmRecord&StartTime=2014-8-25%2000:02:32&EndTime=2014-8-25%2001:02:32&count=500
Success Return	<b>totalCount</b> =1000 <b>found</b> =500 records[0].RecNo=789 records[0].CreateTime=123456789 records[0].Channel=0 records[0]. <b>SenseMethod</b> =DoorMagnetism records[0].RoomNumber=501 records[0].ReadFlag=0 records[0]. <b>Comment</b> =Friend ...
Comment	Parameters in URL: <b>startTime</b> : The start time ,format : 2014-8-25%2000:01:32 <b>endTime</b> : The end time, format: 2014-8-25%2000:02:32 <b>countNo</b> : the number of records to get, The record count, default 1024  Parameters in Response : <b>totalCount</b> : the record count which match condition <b>found</b> : the record count to return <b>SenseMethod</b> :the range is { "DoorMagnetism", "PassiveInfrared", "GasSensor", "SmokingSensor", "WaterSensor", "ActiveInfrared", "CallButton", "UrgencyButton", "Steal", "Perimeter", "PreventRemove", "DoorBell" }

### 12.4.2 Query Access Control Alarm Record

Table 12-11

Syntax	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=AccessControlAlarmRecord[&StartTime=< <b>startTime</b> >&EndTime=< <b>endTime</b> >&count=< <b>countNo</b> >]
Method	GET
Description	Find the AccessControlAlarmRecord record. ( This api is supported by access control device. )
Example	http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=AccessControlAlarmRecord&StartTime=2014-8-25%2000:02:32&EndTime=2014-8-25%2001:02:32&count=500

Success Return	<b>totalCount</b> =1000 <b>found</b> =500 records[0].RecNo=789 records[0].CreateTime=123456789 records[0].UserID=10113 records[0]. <b>EventCode</b> =DoorMagnetism records[0]. <b>DevAddrs</b> =1 records[0].IndexNum=0 records[0].Time=2017-05-10 16:00:01 ...
Comment	Parameters in URL: <b>startTime</b> : The start time ,format : 2014-8-25%2000:01:32 <b>endTime</b> : The end time, format: 2014-8-25%2000:02:32 <b>countNo</b> : the number of records to get, The record count, default 1024 Parameters in Response : <b>totalCount</b> : the record count which match condition <b>found</b> : the record count to return <b>EventCode</b> : alarm event code, the range is { "DoorNotClosed", "BreakIn", "RepeatEnter", "Duress", "AlarmLocal", "ChassisIntruded" }

## 12.5 SIP

The user needs to connect video intercom through CGI protocol, but do not know how to use this protocol to configure the device. This document provides the corresponding CGI protocol according to the configuration data format involved in the specific service.

### 12.5.1 Get SIP Configuration

Table 12-12

Syntax	<b>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=getConfig&amp;name=SIP</b>		
Method	<b>GET</b>		
Description	<b>Get SIP server configuration.</b>		
[ Request Params ] (None)			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	Configuration data table
+SIP	object	R	Configuration name
++AccoutName	string	O	Account name
++SIPServer	string	R	SIP server IP address or domain name
++SIPServerPort	int	R	SIP server port number
++OutboundProxy	string	O	Proxy server IP address or domain name
++OutboundProxyID	string	O	Proxy server ID

++OutboundProxyPort	Int	O	Proxy server port number
++UserID	String	R	User account ID Generally a phone number
++UserType	int	O	User type
++AuthID	String	R	Authentication ID
++AuthPassword	String	R	Authentication password
++STUNServer	String	O	STUN (Simple Traversal of UDP over NATs) server IP address or domain name
++RegisterRealm	String	R	Registration domain
++RegExpiration	int	O	Registration interval, in seconds
++LocalSIPPort	Int	R	Local SIP port 0-65535
++LocalRTPPort	int	R	Local RTP port 0-65535
++UnregisterOnReboot	bool	O	Restart to delete registration information. true: Delete false: Not delete
++DefaultCallNumber	string	O	Default call number
++MediaDetail	Object	O	Media configuration
+++VideoStream	string	O	Video stream enumchar[32]{ "Main": Main stream "Extra1": Sub stream 1 "Extra2": Sub stream 2 "Extra3": Sub stream 3 }
+++AudioStream	string	O	Audio stream enumchar[32]{ "Main": Main stream "Extra1": Sub stream 1 "Extra2": Sub stream 2 "Extra3": Sub stream 3 }
++RouteEnable	bool	O	Enable SIP cross-router or not.
++Route	string	O	Router address, which can be IP address or domain name.
++SIPServerLoginUserName	string	O	Username to log in to VTNC For intelligent building only
++SIPServerLoginPWD	string	O	Password to log in to VTNC For intelligent building only
++IsMainVTO	string	O	Whether the door station is a standby server For intelligent building only
++SIPServerRedundancy	string	O	Standby server IP address For intelligent building only

++SIPServerRedundancyUserNa me	string	O	Standby server login username For intelligent building only
++SIPServerRedundancyPassWo rd	string	O	Standby server login password For intelligent building only
++AnalogNumberStart	string	O	The start number of the analog indoor monitor supported in the analog system For intelligent building only
++AnalogNumberEnd	string	O	The end number of the analog indoor monitor supported in the analog system For intelligent building only
++UserEnable	bool	R	Enable registration For intelligent building only true: Register to the SIP server. false: Not register to the SIP server.

[ Example ]

Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=SIP
Response	<pre> table.SIP.AuthID=8001 table.SIP.AuthPassword=123456 table.SIP.IsMainVTO=0 table.SIP.LocalRTPPort=15000 table.SIP.LocalSIPPort=5060 table.SIP.OutboundProxy=192.168.1.111 table.SIP.OutboundProxyID=8000 table.SIP.OutboundProxyPort=5060 table.SIP.RegisterRealm=VDP table.SIP.Route[0]=sip:10.30.1.2:5060;lr table.SIP.RouteEnable=true table.SIP.SIPServer=192.168.1.111 table.SIP.SIPServerID=8000 table.SIP.SIPServerLoginPWD=admin table.SIP.SIPServerLoginUserName=admin table.SIP.SIPServerPort=5060 table.SIP.SIPServerRedundancyPassWord=admin table.SIP.SIPServerRedundancyUserName=admin table.SIP.UserEnable=true table.SIP.UserID=8001 table.SIP.UserType=2 </pre>

## 12.5.2 Modifying SIP Configuration

Table 12-13

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Modify SIP configuration.		
[ Request Params ] ( key=value at URL )			
Name	Type	R/O	Param Description

+SIP	object	R	Configuration name
++AccoutName	string	O	Account name
++SIPServer	string	R	SIP server IP address or domain name
++SIPServerPort	int	R	SIP server port number
++OutboundProxy	string	O	Proxy server IP address or domain name
++OutboundProxyID	string	O	Proxy server ID
++OutboundProxyPort	Int	O	Proxy server port number
++UserID	String	R	User account ID Generally a phone number
++UserType	int	O	User type
++AuthID	String	R	Authentication ID
++AuthPassword	String	R	Authentication password
++STUNServer	String	O	STUN (Simple Traversal of UDP over NATs) server IP address or domain name
++RegisterRealm	String	R	Registration domain
++RegExpiration	int	O	Registration interval, in seconds
++LocalSIPPort	Int	R	Local SIP port 0-65535
++LocalRTPPort	int	R	Local RTP port 0-65535
++UnregisterOnReboot	bool	O	Restart to delete registration information. true: Delete false: Not delete
++DefaultCallNumber	string	O	Default call number
++MediaDetail	Object	O	Media configuration
+++VideoStream	string	O	Video stream enumchar[32]{ "Main": Main stream "Extra1": Sub stream 1 "Extra2": Sub stream 2 "Extra3": Sub stream 3 }

+++AudioStream	string	O	Audio stream enumchar[32]{ "Main": Main stream "Extra1": Sub stream 1 "Extra2": Sub stream 2 "Extra3": Sub stream 3 }
++RouteEnable	bool	O	Enable SIP cross-router or not.
++Route	string	O	Router address, which can be IP address or domain name.
++SIPServerLog inUserName	string	O	Username to log in to VTNC For intelligent building only
++SIPServerLog inPWD	string	O	Password to log in to VTNC For intelligent building only
++IsMainVTO	string	O	Whether the door station is a standby server For intelligent building only
++SIPServerRedundancy	string	O	Standby server IP address For intelligent building only
++SIPServerRedundancyUserName	string	O	Standby server login username For intelligent building only
++SIPServerRedundancyPassWord	string	O	Standby server login password For intelligent building only
++AnalogNumberStart	string	O	The start number of the analog indoor monitor supported in the analog system For intelligent building only
++AnalogNumberEnd	string	O	The end number of the analog indoor monitor supported in the analog system For intelligent building only
++UserEnable	bool	R	Enable registration For intelligent building true: Register to the SIP server. false: Not register to the SIP server.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&SIP.SIPServer=192.168.1.108&SIP.SIPServerPort=5060&SIP.RegisterRealm=VDP&SIP.RouteEnable=true		
Response	OK		

## 12.6 Registrar

### 12.6.1 Get Registrar Configuration

Table 12-14

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=getConfig&amp;name=Registrar</code>		
Method	<b>GET</b>		
Description	<b>Get Registrar configuration.</b>		
[ Request Params ] (None)			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	Configuration data table
+Registrar	object[]	R	One-dimensional array. Each subscript represents a registration server.
++RegistrarName	char[]	O	Registration server name, which must be unique. (Note: For VTH, use the IP address and port in the configuration to connect VTO.)
++Enable	bool	O	Register to the registration server or not.
++ServerType	enumchar[32]	R	Server type, which is used only for SIP video talk, instead of VT video talk. Enumchar[32]{ "VTO" "H500" "VTNC" "ZYCOO" "ThirdParty" "3CXSystem" "Asterisk" }
++GeneralServerInfo	object	O	
+++Address	char[40]	O	IP address
+++Port	uint	O	Registration port number
+++Password	char[64]	O	Password registered to the register server, which will be used when registration authentication is required.
[ Example ]			
Request	GET <code>http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&amp;name=Registrar</code>		
Response	<code>table.Registar[0].ServerType=VTO</code>		

## 12.6.2 Modifying Registrar Configuration

Table 12-15

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Modify Registrar configuration.		
[ Request Params ] ( key=value at URL )			
Name	Type	R/O	Param Description
+Registrar	object[]	R	One-dimensional array. Each subscript represents a register server.
++RegistrarName	char[]	O	Registration server name, which must be unique. (Note: For VTH, use the IP address and port in the configuration to connect VTO.)
++Enable	bool	O	Register to the registration server or not.
++ServerType	enumchar[32]	R	Server type, which is used only for SIP video talk, instead of VT video talk. Enumchar[32]{ "VTO" "H500" "VTNC" "ZYCOO" "ThirdParty" "3CXSystem" "Asterisk" }
++GeneralServerInfo	object	O	
+++Address	char[40]	O	IP address
+++Port	uint	O	Registration port number
+++Password	char[64]	O	Password registered to the registration server, which will be used when registration authentication is required.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&Registrar.ServerType=VTO		
Response	OK		

## 12.7 Room Number Database Management

### 12.7.1 Adding Room Number

Table 12-16

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=VideoTalkContact		
Method	GET		
Description	Add room number to the contacts.		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
CreateTime	uint	O	Start time UTC seconds, read only
FirstName	string	O	First name
FamilyName	string	O	Last name
VTShortNumber	string	R	Short number for video intercom
VTMiddleNumber	string	O	Middle number for video intercom
VTLongNumber	string	O	Long number for video intercom or serial number of analog indoor monitor
VTNetAddress	string	O	Network address for video intercom
MacAdress	string	O	MAC address
VTOPosition	string	O	Door number linked with indoor monitor
VTSlaveBindMode	enumint	O	Mode when accessing to the analog indoor monitor for video talk Enumint{ 0: SubAddress 1: SubID+SubPort }
VTSlaveId	uint32	O	Allocator address when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
VTSlavePort	uint32	O	Allocator port when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
VTSlaveAddress	string	O	Address of the analog indoor monitor for video talk
NickName	string	O	Nickname
Notes	string	O	Notes
Type	enumchar[32]	O	User type Enumchar[32]{ "VTH": Indoor monitor "VTO": Door station }
RegisterType	enumchar[32]	R	Registration method Enumchar[32]{ "public" "local" }
VTHPassword	string	R	Registration password
VTOBuilding	string	O	Building number
VTOUnit	string	O	Unit number
GroupNumber	string	O	Group

Channel	uint32	O	Channel number, based on which the mobile phone subscribes to the call notification message.
Floors	char[256][4]	O	Floor number (lift control requirements), with up to 256 characters
LiftControlByVTH	bool	O	Lift control is triggered by the indoor monitor opening the door.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
RecNo	int	R	Record ID of successfully creation
[ Example ]			
Request	GET http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=VideoTalkContact&VTShortNumber=101&RegisterType=public&VTHPassword=123456		
Response	RecNo=12345		

## 12.7.2 Getting Records by recno

Table 12-17

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=get&name=VideoTalkContact		
Method	GET		
Description	Search for room number records by recno.		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
recno	int	R	Record ID
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
+RecNo	int	R	Record ID
+CreateTime	uint	O	Start time UTC seconds, read only
+FirstName	string	O	First name
+FamilyName	string	O	Last name
+VTShortNumber	string	R	Short number for video intercom
+VTMiddleNumber	string	O	Middle number for video intercom
+VTLongNumber	string	O	Long number for video intercom or serial number of VTH analog indoor monitor
+VTNetAddress	string	O	Network address for video intercom
+MacAddress	string	O	MAC address
+VTOPosition	string	O	Door number linked with indoor monitor
+VTSlaveBindMode	enumint	O	Mode when accessing to the analog indoor monitor for video talk Enumint{ 0: SubAddress

			1: SubID+SubPort }
+VTSlaveId	uint32	O	Allocator address when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
+VTSlavePort	uint32	O	Allocator port when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
+VTSlaveAddresses	string	O	Address of the analog indoor monitor for video talk
+NickName	string	O	Nickname
+Notes	string	O	Notes
+Type	enumchar[32]	O	User type Enumchar[32]{ "VTH": Indoor monitor "VTO": Door station }
+RegisterType	enumchar[32]	R	Registration method Enumchar[32]{ "public" "local" }
+VTHPassword	string	R	Registration password
+VTOBuilding	string	O	Building number
+VTOUnit	string	O	Unit number
+GroupNumber	string	O	Group
[ Example ]			
Request	GET http://<server>/cgi-bin/recordUpdater.cgi?action=get&name=VideoTalkContact&recno=12345		
Response	record.RecNo=12345 record.CreateTime=123456789 record.FirstName=Dafei record.FamilyName=Wang record.VTShortNumber=0101 record.VTMiddleNumber=11010101 record.VTLongNumber=330103001101010151 record.VTNetAddress=127.0.0.1 record.MacAdress=0A:3E:FF:2A:50:41 record.VTOPosition=01018001 record.VTSlaveBindMode=0 record.VTSlaveId=1258421 record.VTSlavePort=1258421 record.VTSlaveAddress=04:b3:01:f7 record.NickName=Nick record.Notes=Friend record.Type=VTH		

<pre> record.RegisterType=public record.VTHPassword=123456 record.VTOBuilding=01 record.VTOUnit=01 record.GroupNumber=301 record.Channel=1 record.Floors[0]=1 record.LiftControlByVTH=true </pre>
---

## 12.7.3 Updating Room Number Records

Table 12-18

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=update&name=VideoTalkContact		
Method	GET		
Description	Update Room number database records.		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
recno	int	R	Record ID
CreateTime	uint	O	Start time UTC seconds, read only
FirstName	string	O	First name
FamilyName	string	O	Last name
VTShortNumber	string	O	Short number for video intercom
VTMiddleNumber	string	O	Middle number for video intercom
VTLongNumber	string	O	Long number for video intercom or serial number of VTH analog indoor monitor
VTNetAddress	string	O	Network address for video intercom
MacAddress	string	O	MAC address
VTOPosition	string	O	Door number linked with indoor monitor
VTSlaveBindMode	enumint	O	Mode when accessing to the analog indoor monitor for video talk Enumint{ 0: SubAddress 1: SubID+SubPort }
VTSlaveId	uint32	O	Allocator address when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
VTSlavePort	uint32	O	Allocator port when accessing to the analog indoor monitor for video talk (Change string to uint32 for consistency)
VTSlaveAddress	string	O	Address of the analog indoor monitor for video talk
NickName	string	O	Nickname
Notes	string	O	Notes

Type	enumchar[32]	O	User type Enumchar[32]{ "VTH": Indoor monitor "VTO": Door station }
RegisterType	enumchar[32]	O	Registration method Enumchar[32]{ "public" "local" }
VTHPassword	string	O	Registration password
VTOBuilding	string	O	Building number
VTOUnit	string	O	Unit number
GroupNumber	string	O	Group
Channel	uint32	O	Channel number, based on which the mobile phone subscribes to the call notification message.
Floors	char[256][4]	O	Floor number (lift control requirements), with up to 256 characters
LiftControlByVTH	bool	O	Lift control is triggered by the indoor monitor opening the door.
[ Response Params ] (OK)			
[ Example ]			
Request	GET http://<server>/cgi-bin/recordUpdater.cgi?action=update&name=VideoTalkContact&recno=12345&FirstName=Green&FamilyName=Jim&VTHPassword=654321		
Response	OK		

## 12.7.4 Deleting Records by recno

Table 12-19

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=remove&name=VideoTalkContact		
Method	GET		
Description	Delete records by recno.		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
recno	int	R	Record ID
[ Response Params ] (OK)			
[ Example ]			
Request	GET http://<server>/cgi-bin/recordUpdater.cgi?action=remove&name=VideoTalkContact&recno=12345		
Response	OK		

## 12.7.5 Clearing All Room Numbers

Table 12-20

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=clear&name=VideoTalkContact		
Method	GET		
Description	Delete all records .		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
name	string	R	Database name, which must be VideoTalkContact.
[ Response Params ] (OK)			
[ Example ]			
Request	GET http://<server>/cgi-bin/recordUpdater.cgi?action=clear&name=VideoTalkContact		
Response	OK		

## 12.7.6 Getting Total Quantity of Room Number

Table 12-21

URL	http://<server>/cgi-bin/recordFinder.cgi?action=getQuerySize&name=VideoTalkContact		
Method	GET		
Description	Get the total number of records of the database VideoTalkContact.		
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
name	string	R	Database name, which must be VideoTalkContact.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
count	int	R	Total number of records
[ Example ]			
Request	GET http://<server>/cgi-bin/recordFinder.cgi?action=getQuerySize&name=VideoTalkContact		
Response	Count=200		

## 12.8 ElevatorFloorCounter

### 1.1.1 setElevatorFloorInfo

URL	http://<server>/cgi-bin/ElevatorFloorCounter.cgi?action=setElevatorFloorInfo&CheckBaseFloor=<Second>&channel=<ChannelNo>
Method	GET
Description	

[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
Channel	int	R	ChannelNo,from 1, default 1
CheckBaseFloor	string	R	Calibrate base floor, unit floor, It's not necessarily numbers
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
[ Example ]			
Request	http://192.168.1.108/cgi-bin/ElevatorFloorCounter.cgi?action=setElevatorFloorInfo&CheckBaseFloor=5&Channel=1		
Response	OK		

### 12.8.1 getElevatorWorkInfo

URL	http://<server>/cgi-bin/ElevatorFloorCounter.cgi?action=getElevatorWorkInfo&Channel=<ChannelNo>		
Method	GET		
Description			
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
Channel	int	R	ChannelNo,from 1, default 1
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
table	object	R	Operation information, including the current floor, status, speed, whether to calibrate
[ Example ]			
Request	http://192.168.1.108/cgi-bin/ElevatorFloorCounter.cgi?action=getElevatorWorkInfo&Channel=1		
Response	table.ElevatorFloor=20 table.ElevatorWorkState=1 table.ElevatorWorkSpeed=20 table.IsCheckFloor=true		

### 12.8.2 getCaps

URL	http://<server>/cgi-bin/ElevatorFloorCounter.cgi?action=getCaps&Channel=<ChannelNo>		
Method	GET		
Description			
[ Request Params ] (key=value format in URL)			
Name	Type	R/O	Param Description
Channel	int	R	ChannelNo,from 1, default 1
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
caps	object	R	

+IsElevatorFloor Counter	bool	R	Whether the floor configuration function is supported
[ Example ]			
Request	http://192.168.1.108/cgi-bin/ ElevatorFloorCounter.cgi?action=getCaps&channel=1		
Response	caps.IsElevatorFloorCounter=true		

DAHUA HTTP API V3.05 for Whisper House Kft.

44638

## 13.1 File Finder

### 13.1.1 Create a File Finder

Table 13-1

Syntax	<code>http://&lt;server&gt;/cgi-bin/FileFindHelper.cgi?action=startFind&amp;condition.channel=&lt;ChannelNo&gt;&amp;condition.startTime=&lt;start&gt;&amp;condition.endTime=&lt;end&gt;&amp;condition.streamType=&lt;stream&gt;[&amp;condition.flags[0]=&lt;flag&gt;&amp;condition.events[0]=&lt;event&gt;&amp;combineMode.granularity=&lt;granularityValue&gt;&amp;combineMode.types[0]=&lt;combineType&gt;]</code>
Method	GET
Description	Start to find files.
Example	Find a file in channel 1. Event type is "AlarmLocal" or "VideoMotion". Time lines between 2014-1-1 12:00:00 and 2015-1-10 12:00:00. Combine "AlarmLocal" or "VideoMotion" files with granularity 16. URL is: <code>http://172.23.1.66/cgi-bin/fileFindHelper.cgi?action=startFind&amp;condition.channel=1&amp;condition.startTime=2014-1-1%2012:00:00&amp;condition.endTime=2015-1-10%2012:00:00&amp;condition.streamType=Main&amp;condition.flags[0]=Event&amp;condition.events[0]=AlarmLocal&amp;condition.events[1]=VideoMotion&amp;combineMode.granularity=16&amp;combineMode.types[0]=AlarmLocal&amp;combineMode.types[1]=VideoMotion</code>
Success Return	result=08137
Comment	Start to find a file with the above condition and combine files with certain type. If it succeeds, return to find id. Otherwise return to Error. Parameters in URL: <b>ChannelNo</b> : integer, video channel index which starts from 1. <b>start / end</b> : the start/end time when recording. <b>flag</b> : which flags of the file you want to find. It is an array. The index starts from 0. The range of flag is {"Timing", "Marked", "Event", "Restrict"}. If omitted, find files with all the flags. <b>event</b> : by which event the record file is triggered. It is an array. The index starts from 0. The range of event is {"AlarmLocal", "VideoMotion", "VideoLoss"}. This condition can be omitted. If omitted, find files of all the events. <b>stream</b> : which video stream type you want to find. The range of stream is {"Main", "Extra1", "Extra2", "Extra3"}. <b>combineType</b> : which types of the file you want to combined. It is an array. The index starts from 0. The range of combine type is {"AlarmLocal", "VideoMotion", "Timing", "VideoLoss"}. This condition can be omitted. If omitted, file will not be combined. <b>granularityValue</b> : by which granularity to combine files Example:

<p>File 1:</p> <pre>items[0].Channel =1 items[0].StartTime =2011-1-1 12:00:00 items[0].EndTime =2011-1-1 13:00:00 items[0].Events[0]=AlarmLocal items[0].VideoStream=Main items[0].Length =790 items[0].Duration = 3600</pre> <p>File 2:</p> <pre>items[0].Channel =1 items[0].StartTime =2011-1-1 13:00:00 items[0].EndTime =2011-1-1 14:00:00 items[0].Events[0]=AlarmLocal items[0].VideoStream=Main items[0].Length =790 items[0].Duration = 3600</pre> <p>file1 and file2 will be combined to file3</p> <p>File 3:</p> <pre>items[0].Channel =1 items[0].StartTime =2011-1-1 12:00:00 items[0].EndTime =2011-1-1 14:00:00 items[0].Events[0]=AlarmLocal items[0].VideoStream=Main items[0].Length =1580 items[0].Duration = 7200</pre>
---

### 13.1.2 Create a Motion File Finder

Table 13-2

Syntax	<pre>http://&lt;server&gt;/cgi-bin/FileFindHelper.cgi?action=startMotionFind&amp;condition.channel=&lt;ChannelNo&gt;&amp;condition.startTime=&lt;start&gt;&amp;condition.endTime=&lt;end&gt;&amp;condition.streamType=&lt;stream&gt;&amp;motionRegion.senseLevel=&lt;level&gt;[&amp;motionRegion.rects[rectNo][0]=&lt;rect0&gt;&amp;motionRegion.rects[rectNo][1]=&lt;rect1&gt;&amp;motionRegion.rects[rectNo][2]=&lt;rect2&gt;&amp;motionRegion.rects[rectNo][3]=&lt;rect3&gt;]</pre>
Method	GET
Description	Start to find motion files.
Example	<p>Find a file in channel 1, event type is "AlarmLocal" or "VideoMotion", and time between 2014-1-1 12:00:00 and 2015-1-10 12:00:00, motion region is [0,0,21,17]</p> <p>URL is:</p> <pre>http://172.23.1.66/cgi-bin/fileFindHelper.cgi?action=startMotionFind&amp;condition.channel=1&amp;condition.startTime=2014-1-1%2012:00:00&amp;condition.endTime=2015-1-10%2012:00:00&amp;condition.streamType=Main&amp;condition.flags[0]=Event&amp;condition.events[0]=AlarmLocal&amp;condition.events[1]=VideoMotion&amp;motionRegion.senseLevel=1&amp;motionRegion.rects[1][0]=0&amp;motionRegion.rects[1][1]=0&amp;motionRegion.rects[1][2]=21&amp;motionRegion.rects[1][</pre>

	3]=17
Success Return	result=08137
Comment	<p>Start to find a file with the above condition and combine files with certain type. If success, return find id, else return Error.</p> <p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, video channel index which starts from 1.</p> <p><b>start / end</b>: the start/end time when recording.</p> <p><b>flag</b>: which flags of the file you want to find. It is an array. The index starts from 0. The range of flag is {"Timing", "Marked", "Event", "Restrict"}. If omitted, find files with all the flags.</p> <p><b>event</b>: by which event the record file is triggered. It is an array. The index starts from 0. The range of event is {"AlarmLocal", "VideoMotion"}. This condition can be omitted. If omitted, find files of all the events.</p> <p><b>stream</b>: which video stream type you want to find. The range of stream is {"Main", "Extra1", "Extra2", "Extra3"}.</p> <p><b>level</b>: the motion sensitive level, range is 0–6, 0 represent all level</p> <p><b>rectNo</b>: the rects array index, start from 1</p> <p><b>rect0 &amp; rect1 &amp; rect2 &amp; rect3</b>: relative coordinates, rect0 and rect2 range is 0—21, rect1 and rect3 range is 0—17. {0,0,0,0} top—left, {21,0,0,0} top—right, {0,17,0,0} bottom—left, {21,17,0,0} bottom—right</p>

### 13.1.3 Get the File Information Found by the Finder

Table 13-3

Syntax	http://<server>/cgi-bin/FileFindHelper.cgi?action=findNext&findId=< <b>findId</b> >&count=< <b>fileCount</b> >
Method	GET
Description	Find the next files no more than <i>fileCount</i> number.
Example	http://192.168.1.108/cgi-bin/FileFindHelper.cgi?action=findNext&findId=08137&count=100
Success Return	<pre>found=1 items[0].channel =1 items[0].startTime =2011-1-1 12:00:00 items[0].endTime =2011-1-1 13:00:00 items[0].fileType =dav items[0].events[0]=AlarmLocal items[0].streamType=Main items[0].length =790 items[0].duration = 3600</pre>
Comment	<b>findId</b> : The find Id is created by API <a href="#">Create a file finder</a> or API <a href="#">Create a motion file finder</a> . Must create a finder before finding files.

## 13.1.4 Stop the Finder

Table 13-4

Syntax	http://<server>/cgi-bin/FileFindHelper.cgi?action=stopFind&findId=< <b>findId</b> >
Method	GET
Description	Stop the searching operation.
Example	http://192.168.1.108/cgi-bin/FileFindHelper.cgi?action=stopFind&findId=08137
Success Return	OK
Comment	<b>findId</b> : The find Id is created by API <a href="#">Create a file finder</a> or API <a href="#">Create a motion file finder</a> . Must create a finder before finding files.

## 13.1.5 Get Bound Files

Table 13-5

Syntax	http://<server>/cgi-bin/FileFindHelper.cgi?action=getBoundFile&condition.channel=< <b>ChannelNo</b> >&condition.startTime=< <b>start</b> >&condition.endTime=< <b>end</b> >&condition.streamType=< <b>stream</b> >[&condition.flags[0]=< <b>flag</b> >&condition.events[0]=< <b>event</b> >]
Method	GET
Description	Get bound files.
Example	http://<server>/cgi-bin/FileFindHelper.cgi?action=getBoundFile&condition.channel=1&condition.startTime=2014-1-1%2012:00:00&condition.endTime=2015-1-10%2012:00:00&condition.streamType=Main&condition.flags[0]=Timing
Success Return	found=2 items[0].channel =1 items[0].startTime =2011-1-1 12:00:00 items[0].endTime =2011-1-1 13:00:00 items[0].flags [0]= Timing items[0].streamType=Main items[0].length =790 items[0].duration = 3600 items[1].channel =1 items[1].startTime =2011-1-1 13:00:00 items[1].endTime =2011-1-1 14:00:00 items[1].events[0]= Timing items[1].streamType=Main items[1].length =790 items[1].duration = 3600
Comment	Params is same as FileFindHelper.startFind

## 13.2 BandLimit

### 13.2.1 Get Bandwidth Limit State

Table 13-6

Syntax	http://<server>/cgi-bin/BandLimit.cgi?action=getLimitState
Method	GET
Description	Get bandwidth limit state.
Example	http://192.168.1.108/cgi-bin/bandLimit.cgi?action=getLimitState
Success Return	limit=true
Comment	—

## 13.3 Record Files Protection

### 13.3.1 Add Protection

Table 13-7

Syntax	http://<server>/cgi-bin/FileManager.cgi?action=addConditionList&condition.Types[0]=< <b>paramValue</b> >&condition.StartTime=< <b>paramValue</b> >&condition.EndTime=< <b>paramValue</b> >&condition.Channel[0]=< <b>paramValue</b> >
Method	GET
Description	Add protection or access control for record files.
Example	http://192.168.1.108/cgi-bin/FileManager.cgi?action=addConditionList&condition.Types[0]=RecordRestrict&condition.Types[1]=RecordProtect&condition.StartTime=2014-7-3%2021:02:32&condition.EndTime=2014-7-3%2023:02:32&condition.Channel[0]=1&condition.Channel[1]=3
Success Return	OK
Comment	In table below: <b>TypeIndex</b> : The index of type array <b>ChIndex</b> : The index of channel number array

#### Appendix:

ParamName	ParamValue type	Description
condition.Type[ <b>TypeIndex</b> ]	string	An array. The range is {"RecordProtect", "RecordRestrict"}.
condition.StartTime	string	The time format is "Y-M-D H-m-S", example 2011-7-3%2021:02:32
condition.EndTime	string	The time format is "Y-M-D H-m-S"
condition.Channel[ <b>ChIndex</b> ]	integer	Video channel index which starts from 1.

## 13.3.2 Cancel Protection

Table 13-8

Syntax	http://<server>cgi-bin/FileManager.cgi?action=cancelConditionList&condition.Types[0]=<paramValue>&condition.StartTime=<paramValue>&condition.EndTime=<paramValue>&condition.Channel[0]=<paramValue>
Method	GET
Description	Cancel protection of record files.
Example	http://192.168.1.108/cgi-bin/FileManager.cgi?action=cancelConditionList&condition.Types[0]=RecordRestrict&condition.Types[1]=RecordProtect&condition.StartTime=2014-7-3%2021:02:32&condition.EndTime=2014-7-3%2023:02:32&condition.Channel[0]=1&condition.Channel[1]=3
Success Return	OK
Comment	<i>paramValue</i> as <a href="#">Appendix</a> above.

## 13.3.3 Remove Protection

Table 13-9

Syntax	http://<server>/cgi-bin/FileManager.cgi?action=removeConditionList&condition.Types[0]=<paramValue>&condition.StartTime=<paramValue>&condition.EndTime=<paramValue>&condition.Channel[0]=<paramValue>
Method	GET
Description	Remove protection of record files.
Example	http://192.168.1.108/cgi-bin/FileManager.cgi?action=removeConditionList&condition.Types[0]=RecordRestrict&condition.Types[1]=RecordProtect&condition.StartTime=2014-7-3%2021:02:32&condition.EndTime=2014-7-3%2023:02:32&condition.Channel[0]=0&condition.Channel[1]=3
Success Return	OK
Comment	<i>paramValue</i> as <a href="#">Appendix</a> above.

## 13.3.4 DownloadFile

download file

<b>Request URL</b>	http://<server>/cgi-bin/FileManager.cgi?action=downloadFile			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
fileName	string	R	filename or path	download.jpg
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/FileManager.cgi?action=downloadFile&fileName=download.jpg				

Response Params (multipart binary data )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
HTTP/1.1 200 OK Content-type: text/plain;charset=utf-8 CONNECTION: close Set-Cookie:secure; HttpOnly CONTENT-LENGTH: <length>  <Binary Data>				

### 13.3.5 UploadFile

Upload radar photos

<b>Request URL</b>	http://<server>/cgi-bin/FileManager.cgi?action=uploadFile&fileName=< <b>FileName</b> >&Path=< <b>Path</b> >			
<b>Method</b>	POST			
<b>Request Params ( key=value format in url )</b>				
Name	Type	R/O	Description	Example
fileName	string	R	File name, only the types bmp and jpg are available for radars.	"xxxxxx.bmp"
Path	string	R	The file path to store	/upload_pic
<b>Request Example</b>				
POST http://192.168.1.108/cgi-bin/FileManager.cgi?action=uploadFile&fileName=xxxxxx.bmp&fileLength=124879 HTTP/1.1 Host: 192.168.1.108 Connection: keep-alive Content-Type: multipart/form-data;boundary=-----8655433224198 Content-Length: xxxxxxxx  -----8655433224198 Content-Disposition:form-data;name="upload"; filename="xxxxxx.bmp" Content-Type: image/jpeg or application/x-MS-bmp  photo data....  -----8655433224198--				

Response Params ( OK in body )				
Parameter	Type	Required	Description	Example
<b>Response Example</b>				
OK				

## 13.4 Get Daylight

Table 13-10

Syntax	http://<server>/cgi-bin/global.cgi?action=getDST
Method	GET
Description	Get daylight saving time state.
Example	http://192.168.1.108/cgi-bin/global.cgi?action=getDST
Success Return	result = 1
Comment	<b>result:</b> 1/0, yes or not in daylight saving time

DAHUA HTTP API V3.05 for Whisper House Kft.

## 14.1 Coaxial Control IO

### 14.1.1 Control White Light or Speaker

- control white light or speaker

Table 14-1

Syntax	http://<server>/cgi-bin/coaxialControlIO.cgi?action=control&[channel=<channelno>]&info[0].Type=<type>&info[0].IO=<io>&info[0].TriggerMode=<triggermode>
Method	GET
Description	Control white light or speaker
Example	http://192.168.1.108/cgi-bin/coaxialControlIO.cgi?action=control&channel=1&info[0].Type=1&info[0].IO=0&info[0].TriggerMode=1
Success Return	OK
Comment	<b>channelno</b> : video channel index for white light,for siren it's audio output channel, The channel number is default 0 if the request is not carried the param

#### Appendix:

ParamName	ParamValue type	Description
<b>type</b>	integer	Required. 1:wight light, 2:speaker
<b>io</b>	integer	Required. 1:open, 2:close
<b>triggermode</b>	integer	Required. 1:trigger by linkage, 2:trigger by manual

- get white light and speaker status

Table 14-2

Syntax	http://<server>/cgi-bin/coaxialControlIO.cgi?action= getstatus&[channel=< <b>channelno</b> >]
Method	GET
Description	Get white light and speaker stauts
Example	http://192.168.1.108/cgi-bin/coaxialControlIO.cgi?action= getStatus&channel=1
Success Return	status.whitelight=on status.speaker=on
Comment	<b>channelno</b> : video channel index

## 14.2 Pir Alarm

### 14.2.1 Configure Pir Parameter

- Get pir parameter

Table 14-3

Syntax	http://<server>/cgi-bin/pirAlarm.cgi?action=getPirParam&[channel=<channelNo>]
Method	GET
Description	Get pir parameter.
Example	http://192.168.1.108/cgi-bin/pirAlarm.cgi?action=getPirParam&channel=1
Success Return	<pre> <b>head</b>.Enable=true <b>head</b>.PirLink.LightingLink.Enable=true <b>head</b>.PirLink.LightingLink.LightLinkType=Filcker <b>head</b>.PirLink.LightingLink.FilckerIntevalTime=5 <b>head</b>.PirLink.LightingLink.LightDuration=10 <b>head</b>.PirLink.LightingLink.WhiteLightTimeSection=TimeSection <b>head</b>.PirLink.TimeSection[<b>weekday</b>][0]=1 00:00:00-24:00:00 <b>head</b>.PirLink.TimeSection[<b>weekday</b>][1]=0 02:00:00-24:00:00 <b>head</b>.PirLink.TimeSection[<b>weekday</b>][2]=0 03:00:00-24:00:00 <b>head</b>.PirLink.TimeSection[<b>weekday</b>][3]=0 04:00:00-24:00:00 <b>head</b>.PirLink.TimeSection[<b>weekday</b>][4]=0 05:00:00-24:00:00 <b>head</b>.PirLink.TimeSection[<b>weekday</b>][5]=0 06:00:00-24:00:00 <b>head</b>.RecordEnable=true <b>head</b>.RecordChannels=[0, 1, 2] <b>head</b>.RecordLatch=10 <b>head</b>.AlarmOutEnable=true <b>head</b>.AlarmOutChannels=[1, 4] <b>head</b>.AlarmOutLatch=10 <b>head</b>.SnapshotEnable=true <b>head</b>.SnapshotChannels=[2, 4] <b>head</b>.MailEnable=true <b>head</b>.AlarmBellEnable=true <b>head</b>.AlarmBellLatch=10 <b>head</b>.Dejitter=0 <b>head</b>.LogEnable=true <b>head</b>.DetectWindow[0].Level=3 <b>head</b>.DetectWindow[0].Id=0 <b>head</b>.DetectWindow[0].Name=Region0 <b>head</b>.DetectWindow[0].Sensitive=58 <b>head</b>.DetectWindow[0].Threshold=4 <b>head</b>.DetectWindow[0].Region[0]=3932160 <b>head</b>.DetectWindow[0].Region[1]=3932160 ... ... </pre>

	<b>head</b> . DetectWindow [1]...
Comment	if a device has one or more pir, PirParam is a array <b>channel</b> : video channel index <b>weekday</b> : range is [0-6] (Sunday - Saturday) <b>head</b> =configEx[channelNo]

- set pir parameter

Table 14-4

Syntax	http://<server>/cgi-bin/pirAlarm.cgi?action=setPirParam&[channel=< <b>channelNo</b> >]&< <b>paramName</b> >=< <b>paramValue</b> >...&< <b>paramName</b> >=< <b>paramValue</b> >
Method	GET
Description	Set pir parameter.
Example	http://192.168.1.108/cgi-bin/pirAlarm.cgi?action=setPirParam&channel=1&configEx[1].Enable=true&configEx[1].PirLink.LightingLink.Enable=true&...
Success Return	OK
Comment	<b>channel</b> : video channel index <b>head</b> =configEx[channelNo]

#### Appendix:

ParamName	ParamValue type	Description
<b>head</b> .Enable	bool	Enable/Disable motion detect feature in a channel.
<b>head</b> .DetectWindow[ <b>WinNum</b> ].Level	integer	<b>WinNum</b> Index of detect window, there are 4 detect windows at present. Each window is divided into 18 lines and 22 blocks/line. Range is [1—6]. Sensitivity of pir 1: lowest sensitivity. 6: highest sensitivity.
<b>head</b> .DetectWindow[ <b>WinNum</b> ].Id	integer	It is the Id of a detect window.
<b>head</b> .DetectWindow[ <b>WinNum</b> ].Name	string	It is the name of a detect window.
<b>head</b> .DetectWindow[ <b>WinNum</b> ].Sensitive	integer	Range is [0—100]. It presents more sensitive if the value is larger.
<b>head</b> .DetectWindow[ <b>WinNum</b> ].Threshold	integer	Range is [0—100]. It presents the threshold value when trigger motion detect.

ParamName	ParamValue type	Description
<b>head.DetectWindow[WinNum].Region[LineNum]</b>	integer	<p><i>LineNum</i> Index of region, region is divided into lines and each line has several blocks, a line is described by a 32 bit integer, a bit for a block.</p> <p>0=Line 1 1=Line 2 ... ...</p> <p>Currently, region is divided into 18 lines and 22 blocks/line. A bit describes a block in the line. Bit = 1: motion in this block is monitored. Example: MotionDetect[0].Region[0] = 4194303 (0x3FFFFFF):: motion in channel 0 line 0's 22 blocks is monitored. MotionDetect[0].Region[1] =0: motion in line 1's 22 blocks is not monitored. MotionDetect[0].Region[17] = 3: in the last line of channel 0, motion in the left two blocks is monitored.</p>
<b>head.TimeSection[wd][ts]</b>	string	<p><i>wd</i> (week day) range is [0—6] (Sunday - Staurday) <i>ts</i> (time section) range is [0 — 23], timesection table index.</p> <p>Format: mask hh:mm:ss-hh:mm:ss Mask: [0—65535], hh: [0—24], mm: [0—59], ss: [0—59] Mask indicates record type by bits: Bit0: regular record Bit1: motion detection record Bit2: alarm record Bit3: card record</p>
<b>head.PirLink.RecordChannels[ch]</b>	Integer	<p>Range is {0, 1} 0 – do not record on video channel <i>ch</i> 1 – record on video channel <i>ch</i></p>
<b>head.PirLink.RecordEnable</b>	bool	Enable/Disable record function.
<b>head.PirLink.RecordLatch</b>	integer	<p>Range is [10—300]. Unit is seconds, indicates the time to record after input alarm is cleared.</p>

ParamName	ParamValue type	Description
<b>head.PirLink.AlarmOutChannels[<i>ch</i>]</b>	integer	Range is {0, 1}, <i>ch</i> is alarm out channel index. 0 — do not output alarm at alarm out channel <i>ch</i> 1 — output alarm at alarm out channel <i>ch</i>
<b>head.PirLink.AlarmOutEnable</b>	bool	Enable/Disable alarm out function.
<b>head.PirLink.AlarmOutLatch</b>	Integer	Range is [10—300]. Unit is seconds, indicates the time to output alarm after input alarm is cleared.
<b>head.PirLink.SnapshotChannels[<i>ch</i>]</b>	integer	Range is {0, 1} 0 — do not snapshot on video channel <i>ch</i> 1 — snapshot on video channel <i>ch</i>
<b>head.PirLink.SnapshotEnable</b>	bool	Enable/Disable snapshot function.
<b>head.PirLink.Dejitter</b>	integer	Range is [0—255]. Alarm signal dejitter seconds. Alarm signal change during this period is ignored.
<b>head.PirLink.MailEnable</b>	bool	Enable/Disable mail send for alarm.
<b>head.PirLink.AlarmBellEnable</b>	bool	Enable/Disable mail send for alarm.
<b>head.PirLink.AlarmBellLatch</b>	integer	Range is [10, 300] Unit is seconds
<b>head.PirLink.LogEnable</b>	bool	Enable/Disable log for alarm.

## 15.1 Discover Devices

### 15.1.1 Discover Devices on Internet

Table 15-1

Syntax	http://<server>/cgi-bin/deviceDiscovery.cgi?action=attach[&DeviceClass=<deviceClass>]
Method	GET
Description	Discover devices on internet.
Example	http://192.168.1.108/cgi-bin/deviceDiscovery.cgi?action=attach&DeviceClass=VTO
Success Return	<pre> deviceInfo[index].AlarmInputChannels=8 deviceInfo[index].AlarmOutputChannels=0 deviceInfo[index].DeviceClass=VTO deviceInfo[index].DeviceType=VTO2000A deviceInfo[index].HttpPort=80 deviceInfo[index].IPv4Address.DefaultGateway=172.12.0.1 deviceInfo[index].IPv4Address.DhcpEnable=false deviceInfo[index].IPv4Address.IPAddress=172.12.7.102 deviceInfo[index].IPv4Address.SubnetMask=255.255.0.0 deviceInfo[index].IPv6Address.DefaultGateway=2008::1 deviceInfo[index].IPv6Address.DhcpEnable=false deviceInfo[index].IPv6Address.IPAddress=2008::6/112 deviceInfo[index].Mac=00:01:5b:01:44:77 deviceInfo[index].MachineName=YZZ4DZ008D00031 deviceInfo[index].Port=37777 deviceInfo[index].RemoteVideoInputChannels=0 deviceInfo[index].SerialNo=YZZ4DZ008D00031 deviceInfo[index].Vendor=Multi deviceInfo[index].Version=1.200.0.0 deviceInfo[index].VideoInputChannels=1 deviceInfo[index].VideoOutputChannels=16 </pre>
Comment	<p>Parameters in URL:</p> <p><b>deviceClass</b>: in range of {VTO, VTH, VTT, VTS, VTNC, SHG}</p> <p>Parameters in Response:</p> <p><b>index</b>: the array index which starts from 0.</p> <p><b>Version</b>: Software Version</p>

## 15.2 Flashlight

### 15.2.1 Flashlight Config

- Get Flashlight Config

Table 15-2

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=getConfig&amp;name=FlashLight</code>
Method	GET
Description	Get Flashlight config. It does not recommend using it.
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&amp;name=FlashLight</code>
Success Return	<pre> <i>head</i>.Brightness=50 <i>head</i>.Enable=false <i>head</i>.TimeSection[0][0]=1 00:00:00-23:59:59 <i>head</i>.TimeSection[0][1]=0 00:00:00-23:59:59 ... <i>head</i>.TimeSection[6][5]=0 00:00:00-23:59:59         </pre>
Comment	Parameters in Response: <i>head</i> = table.FlashLight

- Set flashlight config

Table 15-3

Syntax	<code>http://&lt;server&gt;/cgi-bin/configManager.cgi?action=setConfig&lt;paramName&gt;=&lt;paramValue&gt;[&amp;&lt;paramName&gt;=&lt;paramValue&gt;...]</code>
Method	GET
Description	Set Flashlight config.
Example	<code>http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&amp;FlashLight.Enable=true&amp;FlashLight.TimeSection[1][0]=1%2012:00:00-18:00:00</code>
Success Return	OK
Comment	—

#### Appendix:

ParamName	ParamValue type	Description
FlashLight.Enable	bool	Enable
FlashLight.Brightness	integer	Brightness
FlashLight.TimeSection[ <b>wd</b> ][ <b>ts</b> ]	string	It is an effective time period for flash light every day. <b>wd</b> (week day) range is [0 — 6] (Sunday — Saturday) <b>ts</b> (time section) range is [0—23], it's index of time section table.

ParamName	ParamValue type	Description
		Format: mask hh:mm:ss-hh:mm:ss Mask: {0,1}, hh: [0—24], mm: [00—59], ss: [00—59] Mask 0: this time section is not used. Mask 1: this time section is used.  Example: TimeSection[1][0]=1 12:00:00—18:00:00 Means flash light is effective between 12:00:00 and 18:00:00 at Monday.

## 15.3 Open Platform

### 15.3.1 Application Start and Stop

- Start Application

Table 15-4

Syntax	http://<server>/cgi-bin/installManager.cgi?action=start&appname=<appname>&appid=<appid>
Method	GET
Description	Start application.
Example	http://192.168.1.108/cgi-bin/installManager.cgi?action=start&appname=FaceDemo&appid=1
Success Return	OK
Comment	Parameters in URL: <b>appname</b> : the application name, <b>appid</b> : optional, the application id

- Stop Application

Table 15-5

Syntax	http://<server>/cgi-bin/installManager.cgi?action=stop&appname=<appname>&appid=<appid>
Method	GET
Description	Stop application.
Example	http://192.168.1.108/cgi-bin/installManager.cgi?action=stop&appname=FaceDemo&appid=1
Success Return	OK
Comment	Parameters in URL: <b>appname</b> : the application name, <b>appid</b> : optional, the application id

## 15.3.2 Install Application

- Install application with app data

Table 15-6

URL	http://<server>/cgi-bin/dhop.cgi?action=uploadApp
Method	POST
Description	Install application.
[ Request Params ] (None)	
[ Response Params ] ( OK )	
[ Example ]	
Request	<pre>POST /cgi-bin/dhop.cgi?action=uploadApp HTTP/1.1 Host: 172.29.2.176 Content-Length: 413124 Content-Type: multipart/form-data; boundary=&lt;boundary&gt;  --&lt;boundary&gt; Content-Disposition: form-data; name="dev_upgrade"; filename="demo.bin" Content-Type: application/octet-stream  &lt;app data&gt; --&lt;boundary&gt;--</pre>
Response	OK

- Install Application with app download url

Table 15-7

URL	http://<server>/cgi-bin/dhop.cgi?action=installAppByUrl
Method	GET
Description	Install application.
[ Request Params ] (key=value format in URL)	
Url	string R The app download url
[ Response Params ] ( OK )	
[ Example ]	
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=installAppByUrl&Url=https://aaa/bbb/cc/app.bin
Response	OK

## 15.3.3 Update Application and License

- Update Application with app download url

Table 15-8

URL	http://<server>/cgi-bin/dhop.cgi?action=updateAppByUrl
Method	GET

Description	Update Application		
[ Request Params ] (key=value format in URL)			
appName	string	R	The app name
Url	string	R	The app download url
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=updateAppByUrl&appName=xxx&Url=https://aaa/bbb/ccc/app.bin		
Response	OK		

- Update firmware with firmware download url

Table 15-9

URL	http://<server>/cgi-bin/dhop.cgi?action=updateFirmwareByUrl		
Method	GET		
Description	Update firmware		
[ Request Params ] (key=value format in URL)			
Url	string	R	The firmware download url
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=updateFirmwareByUrl&Url=https://aaa/bbb/ccc/firmware.bin		
Response	OK		

- Update application license with license download url

Table 15-10

URL	http://<server>/cgi-bin/dhop.cgi?action=updateLicenseByUrl		
Method	GET		
Description	Update application license.		
[ Request Params ] (key=value format in URL)			
appName	string	R	The app name
Url	string	R	The license download url
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=updateLicenseByUrl&appName=xxx&Url=https://aaa/bbb/ccc/license.bin		
Response	OK		

- Update application license with license data

Table 15-11

URL	http://<server>/cgi-bin/dhop.cgi?action=uploadLicense		
Method	POST		
Description	Update application license.		
[ Request Params ] (key=value format in URL)			
appName	string	R	The app name

[ Response Params ] ( OK )	
[ Example ]	
Request	POST /cgi-bin/dhop.cgi?action=uploadLicense&appName=xxx HTTP/1.1 Host: 172.29.2.176 Content-Type: application/octet-stream Content-Length: <length>  <app license data>
Response	OK

### 15.3.4 Uninstall Application

URL	http://<server>/cgi-bin/dhop.cgi?action=uninstall		
Method	GET		
Description	Uninstall application.		
[ Request Params ] (key=value format in URL)			
appName	string	R	The app name
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=uninstall&appName=xxx		
Response	OK		

### 15.3.5 Download Application Log

URL	http://<server>/cgi-bin/dhop.cgi?action=downloadLog		
Method	GET		
Description	Download application log.		
[ Request Params ] (key=value format in URL)			
appName	string	R	The app name
[ Response Params ] ( log file content )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/dhop.cgi?action=downloadLog&appName=xxx		
Response	HTTP/1.1 200 OK Server: Device/1.0 Content-Type: Application/octet-stream Content-Length: <length>  <log file data>		

## 15.4 LensFunc

### 15.4.1 getCaps

URL	http://<server>/cgi-bin/LensFunc.cgi?action=getCaps		
Method	GET		
Description	Get Caps		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
Caps	object	R	The capacity.
+DepthField	uint8	R	Whether support depth field adjust, 0 : not support, 1 : support
+AngleAdjust	object	O	The angel adjust capacity.
++Support	uint8	R	Whether support angel adjust, 0 : not support, 1 : only support horizontal adjust, 2 : only support vertical adjust, 3 : support both horizontal and vertical adjust.
++MaxVelocity	uint8	R	The max velocity, range from 1 to 8
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=getCaps&Channel=1		
Response	Caps.DepthField=1 Caps.AngelAdjust.Support=3 Caps.AngelAdjust.MaxVelocity=8		

### 15.4.2 adjustAngleContinuously

URL	http://<server>/cgi-bin/LensFunc.cgi?action=adjustAngleContinuously		
Method	GET		
Description	Adjust the angel continuously.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
Direction	uint8	R	The moving direction, 0 : negative X axis direction, 1 : positive X axis direction, 2 : negative Y axis direction, 3 : positive Y axis direction
Velocity	uint8	R	The adjust velocity, range from 1 to MaxVelocity
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=adjustAngleContinuously&Channel=1&Direction=2&Velocity=2		
Response	OK		

### 15.4.3 stopAdjustingAngle

URL	http://<server>/cgi-bin/LensFunc.cgi?action=stopAdjustingAngle		
Method	GET		
Description	Stop adjusting angel		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=stopAdjustingAngle&Channel=1		
Response	OK		

### 15.4.4 adjustDepthField

URL	http://<server>/cgi-bin/LensFunc.cgi?action=adjustDepthField		
Method	GET		
Description	Adjust depth position		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
DepthField	float	R	The sensor depth position, range from 0 to 1, -1 means reset.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=adjustDepthField&Channel=1&DepthField=0.3		
Response	OK		

### 15.4.5 adjustDepthFieldContinuously

URL	http://<server>/cgi-bin/LensFunc.cgi?action=adjustDepthFieldContinuously		
Method	GET		
Description	Adjust depth position continuously.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
DepthFieldSpeed	float	R	The sensor depth position adjust velocity, range from -1 to 1, positive and negative value means difference direction, 0 means stop adjust.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=adjustDepthFieldContinuously&Channel=1&DepthFieldSpeed=0.03		

Response	OK
----------	----

## 15.4.6 getDepthFieldStatus

URL	http://<server>/cgi-bin/LensFunc.cgi?action=getDepthFieldStatus		
Method	GET		
Description	Get depth position status.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
Status	object	R	The depth position status.
+Status	string	R	The depth position status, "Normal" : normal status, "AutoDepthField" : adjusting depth position.
+DepthField	float	R	The sensor board relative position, range from 0 to 1
+DepthFieldSteps	uint	R	The depth position total step.
+ResetResult	string	O	The depth position reset result, only valid when reset. "Success" : reset success, "Failed" : reset failed
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=getDepthFieldStatus&Channel=1		
Response	Status.Status=Normal Status.DepthField=0.3 Status.DepthFieldSteps=1000 Status.ResetResult=Success		

## 15.4.7 autoAdjustDepthField

URL	http://<server>/cgi-bin/LensFunc.cgi?action=autoAdjustDepthField		
Method	GET		
Description	Automatic adjust depth position		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/LensFunc.cgi?action=autoAdjustDepthField&Channel=1		
Response	OK		

## 15.5 positionManager

### 15.5.1 getStatus

Syntax	http://<server>/cgi-bin/positionManager.cgi?action=getStatus
Method	GET
Description	Get gps info
Example	http://192.168.1.108/cgi-bin/positionManager.cgi?action=getGPSStatus
Success Return	status.Time=2011-07-03 21:02:32 status.Longitude=120.175556 status.Latitude=30.186389 status.Speed=30.00 status.Altitude=45.0 status.Bearing=45.3 status.SatelliteCount=11 status.WorkStatus=1
Comment	<b>Time:</b> time, ex: 2012-01-04 23:30:30. <b>Longitude:</b> Longitude, in degrees, positive East, negative West <b>Latitude:</b> Latitude, in degrees, positive north, negative South <b>Speed:</b> speed, unit: km/h <b>Altitude:</b> Altitude, unit: m <b>Bearing:</b> Direction angle, unit: degree <b>SatelliteCount:</b> Satellite number <b>WorkStatus:</b> GPS working status, 0= unlocated, 1= un-differenced positioning, 2= differential positioning, 3= invalid PPS, 6= estimating

## 15.6 Radar Adaptor

### 15.6.1 getCaps

Please use the interface getCapsEx.

- ~~Radar equipment detection capability request~~

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=getCaps&channel=< <b>ChannelNo</b> >
Method	GET
Description	Request radar device detection capabilities
Example	http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=getCaps&channel=1
Success-Return	eaps.DetectionRange = 5000 //Zoom in 100 times, while the true value is 50.00 eaps.DetectionAngle = 8003 //Zoom in 100 times eaps.DetectionHuman = 4000 //Zoom in 100 times
Comment	Parameters in URL: <b>ChannelNo:</b> integer, channel index which starts from 1.

## 15.6.2 getCapsEx

- Radar equipment caps request

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=getCapsEx&channel=< <b>ChannelNo</b> >
Method	GET
Description	Request radar device detection capabilities
Example	http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=getCapsEx&channel=1
Success Return	<pre>caps.DetectionRange = 5000 // Radar detection range caps.DetectionAngle = 8003 // Radar detection angle caps.DetectionHuman = 4000 // Radar human detection range caps.Capacity.Support = true //Power setting enable caps.Capacity.List[0] = 150 // Power allows setting list, no more than 24 choices ... caps.Capacity.List[23] = 300 caps.RadarScene.Support = true // Scene setting enable caps.RadarChannel.Support = true // Radar channel setting enable caps.RadarChannel.List[0] = 1 // Channel allows setting list, no more than 24 choices ... caps.RadarChannel.List[23] = 2 caps.MovedDetect.Support = true // Device motion detection setting enabled</pre>
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1.

## 15.6.3 getStatus

- Radar Equipment Status Request

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=getStatus&channel=< <b>ChannelNo</b> >
Method	GET
Description	Request radar device status
Example	http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=getCaps&channel=1
Success Return	status.State = Normal
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1.

### Appendix:

ParamName	ParamValue type	Description
State	string	Radar state, the range is { "Normal" , "Abnormal" }

## 15.6.4 calculateRealSize

- Radar real map size calculation

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=calculateRealSize&channel=< <b>Channel</b> >
--------	--

	<b>No</b> &distance=< <i>distance</i> >&startPoint[0]=< <i>startX</i> >&startPoint[1]=< <i>startY</i> >&endPoint[0]=< <i>endX</i> >&endPoint[1]=< <i>endY</i> >
Method	GET
Description	Calculate radar real map size
Example	http://192.168.0.108/cgi-bin/radarAdaptor.cgi?action=calculateRealSize&channel=1&distance=10.01&startPoint[0]=10&startPoint[1]=20&endPoint[0]=30&endPoint[1]=40
Success Return	realSize.realMap[0]=300 realSize.realMap[1]=400
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1. <b>Distance</b> : Segment distance, Unit: m <b>startPoint</b> : Line segment start coordinates <b>endPoint</b> : Line segment end coordinates

#### Appendix:

ParamName	ParamValue type	Description
realMap	array<integer>	Length and width of actual map, unit:meter

### 15.6.5 attachAlarmPointInfo

- Radar track point request

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=attachAlarmPointInfo&channel=< <b>ChannelNo</b> >&[&heartbeat=< <b>Heartbeat</b> >]		
Method	GET		
Description	Request radar track points		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Channel	int	R	Video channel index which starts from 1
heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat message are "Heartbeat". If this parameter is not present, its default value is 60.
Example	http://192.168.0.108/cgi-bin/radarAdaptor.cgi?action=attachAlarmPointInfo&channel=1&heartbeat=5		
Success Return	HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=< <i>boundary</i> > Connection: closed  --< <i>boundary</i> > Content-Type: application/octet-stream		

	<pre> Content-Length: &lt;length&gt;  Channel=1 info[0].PointType=0x80 info[0].RegionNumber=1 info[0].ObjectType=1 info[0].TrackID=1 info[0].Distance=1001 //Zoom in 100 times, while the true value is 10.01 info[0].Angle=2010 //Zoom in 100 times info[0].Speed=501 //Zoom in 100 times ..... info[3].PointType=0x80 .....  --&lt;boundary&gt; Content-Type: text/plain Content-Length: 11  Heartbeat -&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;  Channel=1 info[0].PointType=0x80 .....  --&lt;boundary&gt; ... </pre>
Comment	<p>Parameters in URL:</p> <p><b>ChannelNo</b>: integer, channel index which starts from 1.</p> <p><b>Notice</b>:</p> <p>A detach occurs when the connection is disconnected.</p>

## 15.6.6 getLinkSDState

- Linked device status request

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=getLinkSDState&channel=< <b>ChannelNo</b> >&ip[0]=< <i>ip</i> >&ip[1]=< <i>ip</i> >.....&ip[23]=< <i>ip</i> >
Method	GET
Description	Request linkage device status
Example	http://192.168.0.108/cgi-bin/radarAdaptor.cgi?action=getLinkSDState&channel=1&ip[0]=192.168.1.115&ip[1]=192.168.1.116
Success	state[0].SDLinkIP=192.168.1.115

Return	state[0].State=0 state[0].Channel=1 state[1].SDLinkIP=192.168.1.116 state[1].State=0 state[1].Channel=2
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1. <b>ip</b> : IP address of the linkage device

#### Appendix:

ParamName	ParamValue type	Description
State	integer	device state, 0: out of time, 1: offline, 2: disconnected

### 15.6.7 manualLocate

- Radar manual positioning request

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=manualLocate&channel=< <b>ChannelNo</b> >&point[0]=< <b>x</b> >&point[1]=< <b>y</b> >
Method	GET
Description	Request manual radar positioning
Example	http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=manualLocate&channel=1&point[0]=10&point[1]=10
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1. <b>point</b> : Pixel coordinates in the code stream

### 15.6.8 startRadarCalibration

- Radar equipment calibration

Syntax	http://<server>/cgi-bin/radarAdaptor.cgi?action=startRadarCalibration&channel=< <b>ChannelNo</b> >&mode=< <b>mode</b> >&direction=< <b>direction</b> >&step=< <b>step</b> >
Method	GET
Description	Calibration of radar equipment
Example	http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=startRadarCalibration&channel=1&mode=1&direction=1&step=5
Success Return	OK
Comment	Parameters in URL: <b>ChannelNo</b> : integer, channel index which starts from 1. <b>mode</b> : Calibration mode, 0:auto, 1: manual,2:init <b>direction</b> : Calibration direction, effective in manual mode , 0: invalid, 1: top, 2: bottom,

3: left, 4: right, 5: left\_top, 6: right\_top, 7: left\_bottom, 8: right\_bottom  
**step:** step size

## 15.6.9 addRadarLinkSD

- Radar link SD added

<b>Request URL</b>	http://<server>/cgi-bin/radarAdaptor.cgi?action=addRadarLinkSD			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Devices	array<object>	是	Add or update the added device login information	
+PassWord	char[32]	是	password, plaintext	"admin123"
+UserName	char[128]	是	user name	"admin"
+Port	integer	是	port	5000
+SDLinkIP	char[32]	是	device ip adress	"192.168.1.108"
Channel	integer	是	video channel, which starts from 1.	1
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=addRadarLinkSD&Devices[0].SDLinkIP=10.11.17.98&Devices[0].Port=5000&Devices[0].Username=admin&Devices[0].PassWord=admin123&Channel=1				

<b>Response Params ( key=value format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
SD	object	是	dome camera	
+Info	object	是	information	
++ErrorCode	integer	是	errorcode 0: normal 1: overtime 2: Wrong user name or password 3: The device does not support linkage	0
<b>Response Example</b>				
SD.Info.ErrorCode=0				

## 15.6.10 delRadarLinkSD

- Remove radar link SD

<b>Request URL</b>	http://<server>/cgi-bin/radarAdaptor.cgi?action=delRadarLinkSD			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
Devices	array<object>	是	Delete the SD added in radar equipment Empty array will delete	

			all linkage equipment by default	
+SDLinkIP	char[32]	是	device to be deleted	"192.168.1.108"
Channel	integer	是	video channel, which starts from 1.	1

#### Request Example

http://192.168.1.108/cgi-bin/radarAdaptor.cgi?action=delRadarLinkSD&Devices[0].SDLinkIP=10.11.17.98&Channel=1

#### Response Params ( key=value format in body )

Name	Type	R/O	Description	Example
<b>Response Example</b>				
OK				

## 15.6.11 [Config]MapPara

map para

Parameter	Type	Required	Description	Example
MapPara	object[]	Yes	One-dimensional array, with one configuration for each video channel	
+RadarCoordinate	object	Yes	Configuration of radar position on the map	null
++RadarDirectionAngle	double	Yes	Configuration of radar direction (°)	70.5
++RadarPixelPoint	integer	Yes	Pixel coordinates of radar on the map	[10, 20]
+MapSize	object	Yes	Configuration of map size	null
++PixelLine	integer[][]	Yes	Pixel coordinates of line segment in the image	[[10,10], [20,20]]
++Distance	double	Yes	The actual length represented by the line segment in the image (m)	10.1

#### Example

http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=MapPara

### Appendix: configuration operation method

**configname** and **configobject** in the following table correspond to specific configuration names and configuration contents.

special explanation:

In the return of **getConfig**, there is a "table" level above the specific configuration.

In the request of **setConfig**, there is no "table" level above the specific configuration.

- getConfig

<b>Syntax</b>	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name= <b>configName</b>			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
Name	Type	R/O	Description	Example
name	string	R	config name	MapPara
<b>Request Example</b>				
http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name= MapPara				

Response Params ( key=value format in body )				
Name	Type	R/O	Description	Example
table	object	是	table prefix	
+configObject	configObject	是	Specific configuration information	configObject

#### Response Example

```
table.MapPara[0].RadarCoordinate.RadarDirectionAngle=70.5
table.MapPara[0].RadarCoordinate.RadarPixelPoint[0]=10
table.MapPara[0].RadarCoordinate.RadarPixelPoint[1]=20
table.MapPara[0].MapSize.PixelLine[0][0]=10
table.MapPara[0].MapSize.PixelLine[0][1]=10
table.MapPara[0].MapSize.PixelLine[1][0]=20
table.MapPara[0].MapSize.PixelLine[1][1]=20
table.MapPara[0].MapSize.Distance=10.1
table.MapPara[1].RadarCoordinate.RadarDirectionAngle=70.5
...
```

- setConfig

<b>Syntax</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
Name	Type	R/O	Description	Example
configObject	string	R	Specific configuration content	configObject

#### Request Example

```
http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&MapPara[0].RadarCoordinate.RadarDirectionAngle=70.5&MapPara[0].RadarCoordinate.RadarPixelPoint[0]=10&MapPara[0].RadarCoordinate.RadarPixelPoint[1]=20&MapPara[0].MapSize.PixelLine[0][0]=10&MapPara[0].MapSize.PixelLine[0][1]=10&MapPara[0].MapSize.Distance=10.1
```

Response Params ( OK in body )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
OK				

## 15.6.12 [Config]RadarAnalyseRule

RadarAnalyseRule

Parameter	Type	Required	Description	Example
RadarAnalyseRule	object[][]	Yes	Radar alarm linkage configuration object[][24] Two-dimensional array. The first dimension indicates video channel, and the second dimension indicates multiple linkage configurations.	

+EventHandler	object	Yes	Alarm linkage configuration	null
++LogEnable	bool	Yes	Whether to record alarm logs. The type is fixed to EventStart/EventStop/EventPulse and the original event type is recorded in the Detail.Code field in the log.	false
++TimeSection	TimeSection[][]	Yes	Alarm period. For the definition of alarm period, see <i>Dahua Information Exchange Format</i> . TimeSection is used for the filtering of notify action. No data is sent if it is not within the period. All fields except this one are used for listener response.	[[["100:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59", "00:00:00-23:59:59"],]
++SnapshotChannels	int[]	Yes	List of snapshot channel numbers. One-dimensional array. Each member indicates that the corresponding channel needs to take snapshot, and the channel number starts from 0.	[0]
++SnapshotEnable	bool	Yes	Enable snapshot	false
++RecordLatch	integer	Yes	Record delay time (second) Range [10, 300]	10
++RecordChannels	int[]	Yes	List of record channel numbers. One-dimensional array. Each member indicates that the corresponding channel needs to record, and the channel number starts from 0.	[0]
++RecordEnable	bool	Yes	Record enabling, with RecordChannels. Record is enabled if it is true; start recording when the event action is start and stop recording when it is stop. Record is disabled if it is false.	false
++VoiceEnable	bool	Yes	Voice prompt	false
++BeepEnable	bool	Yes	Buzzer	false
++MessageEnable	bool	Yes	Upload to the alarm center server.	false
++MailEnable	bool	Yes	Send email.	false
++TipEnable	bool	Yes	Local message box prompt	false
++AlarmOutL	integer	Yes	Output delay time (second) after the alarm	10

atch			output stops. Range [10, 300]	
++AlarmOutEnable	bool	Yes	Enable alarm output	false
++AlarmOutChannels	int[]	Yes	List of alarm output channel numbers. One-dimensional array. Each member indicates that the corresponding channel needs to output alarm, and the channel number starts from 0.	[0]
+Enable	bool	Yes	Enable alarm enabling	false
+AlarmOutNumber	integer	Yes	Alarm configuration number, which is unique.	1
+SDLinkIP	char[32]	No	Controlled device IP (which is optional and reserved. Ignore this field due to business adjustment).	"192.168.1.108"

#### Example

<http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarAnalyseRule>

#### Note:

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.13 [Config]RadarCalibration

### RadarCalibration

Parameter	Type	Required	Description	Example
RadarCalibration	object[]	Yes	Configuration of radar and PTZ cameras calibration. One-dimensional array, with one configuration for each video channel.	
+SlopeAngle	double	Yes	Slope compensation angle.	10.0
+InstallHeight	double	Yes	Radar installation height (m).	10.5
+CalibrationParas	object[]	Yes	Calibration parameter.	null
++TiltRecoupAngle	double	Yes	Vertical compensation angle of linked PTZ camera. The range is -90° to 90°; 0° in the horizontal direction; 0 to 90° below the horizontal direction; -90° to 0° above the horizontal direction.	5.0
++LinkSDHeight	double	Yes	Installation height of linked PTZ camera (m).	10.5
++CalibrationPos	object[]	Yes	Array of calibration points.	null
+++Ptz	float[3]	Yes	PTZ value (normalized value) of calibration points. The first element is the horizontal angle, which is normalized to -1 to	[0.95, 0.43, 0.12]

			1. The second element is the vertical angle, which is normalized to -1 to 1. The third element is the zoom times, which is normalized to 0 to 1.	
+++AlarmPixelPoint	int[2]	Yes	Pixel coordinates of calibration points.	[10, 20]
++SDLinkIP	char[32]	Yes	Controlled device IP.	"192.168.1.108"
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarCalibration				

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.14 [Config]RadarGuardLine

RadarGuardLine

Parameter	Type	Required	Description	Example
RadarGuardLine	object[[[10]	Yes	Two-dimensional array. The first dimension indicates video channel, and the second dimension indicates multiple rule lines (up to 10).	
+AlarmOutNumber	integer	Yes	Alarm output configuration. Configure the AlarmOutNumber field according to RadarAnalyseRule.	1
+TargetFilter	integer	Yes	Target filtering bit0: Reserved bit1: Human. bit2: Vehicle. bit3: Animal (Enable by setting it to 1)	0x01
+Polygon	integer	Yes	Rule point, pixel coordinates [width, height]. Range of number of points [3, 24]	[[0, 0], [128, 128],...,[]]
+Type	char[32]	Yes	Rule type, alarm or shield type. High Alarm Alarm Shield	"Alarm"
+Enable	bool	Yes	Whether to enable the rules	true
+RegionNumber	integer	Yes	Rule line number Range [1, 10]	1

+Name	char[128]	Yes	Rule name, which cannot be repeated.	"GuardLine1"
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarGuardLine				

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.15 [Config]RadarLink

RadarLink

Parameter	Type	Required	Description	Example
RadarLink	object[]	Yes	One-dimensional array, with one configuration for each video channel	
+RadarLink	bool	Yes	Enable radar-PTZ linkage	true
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarLink				

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.16 [Config]RadarLinkDevice

RadarLinkDevice

Parameter	Type	Required	Description	Example
RadarLinkDevice	object[][]	Yes	Remote device linkage configuration. object[][24] Two-dimensional array. The first dimension indicates video channel, and the second dimension indicates multiple remote linkage devices.	
+DeviceName	char[128]	Yes	Device name	"Dahua North Gate"
+DeviceType	char[32]	Yes	Device type	"200W PTZ camera"
+Vendor	char[32]	Yes	Device source	"DH"
+SDLinkIP	char[32]	Yes	Controlled device IP	"192.168.1.108"
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarLinkDevice				

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.17 [Config]RadarPara

RadarPara

Parameter	Type	Required	Description	Example
-----------	------	----------	-------------	---------

RadarPara	object[]	Yes	Radar function configuration. One-dimensional array, with one configuration for each video channel.	
+Sensitivity	object	No	Recognition sensitivity information.	null
++Level	integer	No	Optional. Skip if this field does not exist. The range is -5 to 5; 0 is the default value; the smaller the value, the higher the human recognition rate; the larger the value, the higher the vehicle recognition rate.	0
+TargetRatio	object	Yes	Tracking screen ratio for radar-PTZ linkage.	null
++Ratio	integer	Yes	Reciprocal of screen ratio. Required. Calculate the reciprocal. For example, when the screen ratio is 0.5, Ratio = 2; when the screen ratio is 0.25, Ratio = 4.	1
+RadarChannel	object	Yes	Radar channel configuration.	null
++Route	integer	Yes	Channel selection. 1: Channel 1 2: Channel 2	1
+Capacity	object	Yes	Configuration of radar transmitting power.	null
++Power	integer	Yes	Radar transmitting power is measured by detection distance (m).	300
+Structured	object	Yes	Structured information configuration	null
++TrackDisplayTime	integer	No	Range [1, 30] (s).	5
++TrackType	integer	No	0: Disable 1: Trajectory box 2: Trajectory	0
++Enable	bool	Yes	Enable structured information display	false
+Scene	object	Yes	Scene configuration	null
++Type	integer	Yes	Scene selection. 1: Default. 2: Shrub. 3: Spacious. 4: Custom.	1

#### Example

<http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarPara>

#### Note:

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.18 [Config]RadarTrackGlobal

RadarTrackGlobal

Parameter	Type	Required	Description	Example
RadarTrackGlobal	object[]	Yes	One-dimensional array, with one configuration for each video channel	
+TrackSwitchMode	char[32]	Yes	Tracking switching mode Tour/time priority/distance priority Rotation/TimePriority/DistancePri	"Rotation"

			ority	
+TrackSwitchTime	integer	Yes	Tracking switching time (s) Range [1, 15]	5
+TrajectoryTime	integer	Yes	Trajectory time (s) Range [3, 30]	5
+SectorDisable	bool	Yes	Enable protection area hiding	false

**Example**

<http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RadarTrackGlobal>

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.6.19 [Config]RemoteSDLink

RemoteSDLink

Parameter	Type	Required	Description	Example
RemoteSDLink	object[][24]	Yes	Remote device linkage enabling configuration. Two-dimensional array. The first dimension indicates video channel, and the second dimension indicates multiple remote linkage devices.	
+RadarLink	bool	Yes	Enable linkage.	true
+SDLinkIP	char[32]	Yes	Controlled device IP.	"192.168.1.108"

**Example**

<http://192.168.1.108/cgi-bin/configManager.cgi?action=getConfig&name=RemoteSDLink>

**Note:**

Please refer to appendix in chapter 15.6.11 for configuration operation method

## 15.7 GpsControl

### 15.7.1 getCaps

Get the capability of position system.

Syntax	<a href="http://&lt;ip&gt;/cgi-bin/GpsControl.cgi?action=getCaps">http://&lt;ip&gt;/cgi-bin/GpsControl.cgi?action=getCaps</a>
Method	<b>GET</b>
Description	Get GpsControl caps.
Example	<a href="http://192.168.1.108/cgi-bin/GpsControl.cgi?action=getCaps">http://192.168.1.108/cgi-bin/GpsControl.cgi?action=getCaps</a>
Success	caps. SUPPORTMODE[0]="GPS"
Return	caps. SUPPORTMODE[1]="BEIDOU"

### 15.7.2 GPS config

- getConfig of position system

Syntax	http://<ip>/cgi-bin/configManager.cgi?action=getConfig&name= <b>GPS</b>
Method	<b>GET</b>
Description	Get the config of position system.
Example	http://192.168.1.108/cgi-bin/global.cgi?action= getConfig&name= <b>GPS</b>
Success Return	<b>Table.GPS[0].Enable[0]=true</b> <b>Table.GPS[1].Enable[1]=false</b> <b>Table.GPS[0].CurMode[0]="GPS"</b> <b>Table.GPS[1].CurMode[1]="BEIDOU"</b> <b>Table.GPS[0].SyncTime[0]=true</b> <b>Table.GPS[1].SyncTime[1]=false</b>

- setConfig of position system.

Syntax	http://<ip>/cgi-bin/configManager.cgi?action=setConfig&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
Comment	In below table, <b>head</b> =GPS
Success Return	<b>OK or ERROR</b>

Appendix:

ParamName	ParamValue type	Description
<b>head.Enable</b>	bool	true/false
<b>head.CurMode</b>	string	"GPS"/"BEIDOU"
<b>head.SyncTime</b>	bool	true/false

## 15.8 LensManager

### 15.8.1 resetAngle

- Reset camera lens angle

URL	http://< server >/cgi-bin/LensManager.cgi?action=resetAngle		
Method	GET		
Description	Reset camera lens angle		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
Channel	uint	R	Video channel index which starts from 1.
[ Response Params ] ( OK )			
[ Example ]			
Request	http://172.27.1.153/cgi-bin/LensManager.cgi?action=resetAngle&Channel=1		

Response	OK
----------	----

## 15.9 Advertisement

### 15.9.1 Getting Ad Placement Capability

Get advertisement placement capability. For access control device.

#### 15.9.1.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=getAdvertisementCaps			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
params	object	No		
<b>Example</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=getAdvertisementCaps			

#### 15.9.1.2 Response

<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
Support	bool	Yes	Supports Ad placements or not	true
FileStorageType	array/array<>	Yes	Set of advertising file storage types; array, up to 16 types Enumchar[32] { "Local": The device directly gets the file from the local storage path (The platform can upload this file through FileManager.uploadFile to the local path of the device). "FTP": Advertising files are stored on the FTP server and need to be downloaded by the device from the FTP server (This type was used for intelligent building	"Local","FTP","HTTP"

			products). "HTTP": The device downloads the file through the http URL provided by the cloud platform. }	
LocalStorage	object	No	This exists when FileStorageType has the Local type.	
+Path	string	Yes	Local path of the device where advertising files are stored. The client uploads advertising files to the local path of the device; for example, when uploading through the FileManager.uploadFile.Path field, enter this field (up to 256 characters).	"/mnt/mtd/PublishInfo"
Video	object	No	Format capabilities corresponding to the video file contained in the advertising video file.	
+Support	bool	Yes	Supports playing advertisement videos or not.	true
+StreamType	int	Yes	Set of encoding formats supported by the device; array, up to 16 formats Enumchar[32] { "H264": H264 format "H264Private": H264 encoded dav video "H265": H265 format "H265Private": H265 encoded dav video "MPEG4": MPEG4 encoding "MPEG4Private": MPEG4 encoded dav format }	"H264Private", "H265Private"
+MinWidth	int	Yes	Minimum pixels of video width.	50
+MaxWidth	int	Yes	Maximum pixels of video width.	800
+MinHeight	int	Yes	Minimum pixels of video height.	50
+MaxHeight	int	Yes	Maximum pixels of video	1280

			height.	
Audio	object	No	Capabilities corresponding to the audio file contained in the advertising file.	
+Support	bool	Yes	Supports playing advertisement audios or not.	true
+Formats	array/array<>	Yes	List of supported audio decoding formats; the array with up to 64 returns. Each member represents a combination of decoding parameters.	
++Compression	string	Yes	Audio compression format enumchar[32]{ /* PCM format with header */ "PCM" "G.711A" "G.711Mu" "AAC" }	"PCM"
++Frequency	int	Yes	Audio sampling frequency; value range: 8 K–192 K; for example, 16000 indicates 16K.	16000
++Depth	int	Yes	Sampling depth enumint{ 8, 16, 24, }	16
++PacketPeriod	int	Yes	Packaging cycle, in ms; value range: [10, 250]	40
OneAVFileMaxSize	int	Yes	Size of a single audio or video advertising file, in KB.	102400
MaxAVFileNums	int	Yes	Maximum number of audio or video advertising files supported.	5
Pictrue	object	Yes	Format capabilities corresponding to the image file contained in the advertising file.	
+Support	int	Yes	Supports playing advertisement images or not.	true
+Types	array/array<>	Yes	Set of encoding format capabilities of the image	"PNG","JPG","BMP"

			advertising file; array, up to 16 formats Enumchar[32] { "PNG": Image in PNG format "JPG": Image in JPG format "BMP": Image in BMP format }	
+MinWidth	int	Yes	Minimum pixels of image width.	
+MaxWidth	int	Yes	Maximum pixels of image width.	
+MinHeight	int	Yes	Minimum pixels of image height.	
+MaxHeight	int	Yes	Maximum pixels of image height.	
+OnePicMaxSize	int	Yes	Maximum size of a single image file, in KB.	
+MaxPicNums	int	Yes	Maximum number of images supported.	

#### Example

```

Support=true,
FileStorageType=Local,FTP,HTTP
LocalStorage.Path="/mnt/mtd/PublishInfo"
Video.Support=true
Video.StreamType="H264Private","H265Private"
Video.MinWidth=50
Video.MaxWidth=800
Video.MinHeight=50
Video.MaxHeight=1280
Audio.Support=true
Audio.Formats[0].Compression="PCM"
Audio.Formats[0].Frequency=16000
Audio.Formats[0].Depth=16
Audio.Formats[0].PacketPeriod=40
OneAVFileMaxSize=102400
MaxAVFileNums=5
Pictrue.Support=true
Pictrue.Types="PNG","JPG","BMP"
Pictrue.MinWidth=50
Pictrue.MaxWidth=800
Pictrue.MinHeight=50
Pictrue.MaxHeight=1280
Pictrue.OnePicMaxSize=100
Pictrue.MaxPicNums=50

```

## 15.9.2 Acquiring Advertisement Resource List

CGI command of VideoOutput.list protocol: Acquiring advertisement resource list.

### 15.9.2.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=list&path=/mnt/dvr/sdao/2010/8/11/dav			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
path	char[260]	Yes	Acquire path	"/mnt/dvr/sdao/2010/8/11/dav"
<b>Complete Example</b>				
http://<server>/cgi-bin/VideoOutput.cgi?action=list&path=/mnt/dvr/sdao/2010/8/11/dav				

### 15.9.2.2 Response

<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
elementInfo	object[]	No	File element information.	
+type	enumchar[16]	No	File element type. enumchar[16] { "File": file "Directory": Directory}	"File"
+file	object	No	File information (valid when Type= "File").	
++FileType	char[64]	No	File type. enumchar[16]{ "Video": Video file "Image": Image file "Audio": Audio file }	"Video"
++CreateTime	char[20]	No	Creation time	"2010-4-15 9:58:32"
++ModifyTime	char[20]	No	Modification time	"2010-4-15 9:58:32"
++Size	double	No	File size. The decimal part is meaningless. Unit: Byte.	1873.0
++path	char[260]	No	Absolute path.	"/mnt/dvr/sdao/2010/8/11/dav.jpg"

++Desc	char[128]	No	Customized file description.	"xxxxxxx"
+directory	object	No	Directory information (valid when Type = "Directory").	
++CreateTime	char[20]	No	Creation time. Format: "Y-M-D H-M-S".	"2010-4-15 9:58:32"
++path	char[260]	No	Absolute path.	"/mnt/dvr/sdao/2010/8/11/dav"

#### Complete Example

```

elementInfo[0].type=File
elementInfo[0].file.FileType=Video
elementInfo[0].file.CreateTime=2010-4-15 9:58:32
elementInfo[0].file.ModifyTime=2010-4-15 9:58:32
elementInfo[0].file.Size=1873.0
elementInfo[0].file.path=/mnt/dvr/sdao/2010/8/11/dav.jpg
elementInfo[0].file.Desc=xxxxxxx
elementInfo[0].directory.CreateTime=2010-4-15 9:58:32
elementInfo[0].directory.path=/mnt/dvr/sdao/2010/8/11/dav
...
elementInfo[1].type=File
elementInfo[1].file.FileType=Wireshark/tcpdump
elementInfo[1].file.CreateTime=2010-4-15 9:58:32
elementInfo[1].file.ModifyTime=2010-4-15 9:58:32
elementInfo[1].file.Size=1873.0
elementInfo[1].file.path=/mnt/dvr/sdao/2010/8/11/dav.jpg
elementInfo[1].file.Desc=xxxxxxx
elementInfo[1].directory.CreateTime=2010-4-15 9:58:32
elementInfo[1].directory.path=/mnt/dvr/sdao/2010/8/11/dav
...

```

## 15.9.3 Deleting Advertisement Files

Advertisement resource deletion interface

### 15.9.3.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=removeFiles&fileName[o]=a.dav			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
fileName	char[32][128]	Yes	File name	["a.dav", "b.dav"]
<b>Complete Example</b>				
http://<server>/cgi-bin/VideoOutput.cgi?action=removeFiles&fileName[o]=a.dav				

## 15.9.4 Sending Advertisement Resources

Call VideoOutput.deliveryFileEx component interface internally.

### 15.9.4.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=deliveryFile			
<b>Method</b>	POST			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
Port	int	No	Video output port. The value represents the array index of the VideoOut configuration.	0
Number	int	No	The current advertisement plan number. The caller can use this number to set up different advertisement plans.	0
TimeSection	TimeSchedule	No	Advertisement playing period. It is a two-dimensional array in which the first seven elements correspond to seven days of a week, and the eighth element corresponds to holidays, with a maximum of six periods per day. The eighth element can be left blank or entered as null, indicating that the holiday period is not supported.	[["100:00:00-07:00:00", "209:00:00-17:30:00", "317:30:00-23:59:59" ], ..., [, ]
Enable	bool	No	Enable play function.	true
StartTime:	char[20]	No	Start time	"2016-08-10 10:00:00"
Name	char[128]	No	Advertisement name	"Happy holliday"
EndTime:	char[20]	No	End time. It is meaning only for loop video play.	"2016-08-10 12:00:00"
Mode	char[16]	No	Video play mode. Once: Each file in the list is played once. Repeat: Files in the list are played repeatedly until end time. Alone: Cut-in play (playing alone).	"Once"
Type	char[16]	No	Operation type. If the field does not exist, it is Replace by	"Replace"

			<p>default.</p> <p>Replace: After it is cleared, the advertisement will be sending again.</p> <p>Add: Add advertisement resources.</p> <p>Remove: Remove advertisement resources.</p> <p>Operate according to the ID field of FileList..</p> <p>Clear: Clear all advertisement resources.</p>	
FileList	object[]	No	Video file list	
+FileType	enumchar[16]	No	<p>File type.</p> <pre>enumchar[16]{ "Video": Video file "Image": Image file "Audio": Audio file }</pre>	"Video"
+PlayWithMode	int	No	<p>It is 1 by default. File mode:</p> <p>1: Recognition mode.</p> <p>2: Information release mode.</p> <p>3: Home page mode.</p> <p>The file can only be played in the corresponding working mode. The recognition mode is the standard function of the program, which will automatically preform object recognition when detecting a object. The information release mode requires human-device interaction before recognition.</p>	1
URL	char[128]	No	The resource address of the file.	"ftp://192.168.1.108/1.dav"
+URLEx	char[512]	No	<p>File extension resource address: The original URL field is too short and not enough for the supporting ICC platform (currently it has reached 160 bytes), which needs to be increased. The maximum length of URLEx is 512 bytes. Compatibility logic of platform: When the Support field of the device</p>	"http://10.35.81.187:8927/1952fdd7-48e8-11e1-b015-08eded28a344/20210315/1/cdce2e98-855d-11eb-ac3f-08eded28a344.jpg?token=fe65ac56-ff42-4597-

			capability interface VideoOutput.getAdvertisementCaps returns true, the real download address is filled in the URLEx field, and the URL field is filled with an empty string. Compatibility logic of device: If the URL is not an empty string, the URL is used directly; only when the URL field is an empty string and URLEx is not an empty string, the URLEx field is used.	8of4-33e571deab22"
+Sustain	int	No	Stay time of each Image, which is valid only when FileType is Image. Unit: Second.	5
+TimeSection	TimeSchedule	No	(The new advertisement program might not be implemented) Play period. It is a two-dimensional array in which the first seven elements correspond to seven days of a week, and the eighth element corresponds to holidays, with a maximum of six periods per day. The eighth element can be left blank or entered as null, indicating that the holiday period is not supported.	[["100:00:00-07:00:00", "109:00:00-17:30:00", "117:30:00-23:59:59"], ..., [], ]
+Size	int	No	Files size. Unit: Byte. It is convenient to verify whether the file can be downloaded successfully before downloading. The actual download might fail, because the space limit of the device folder (used to save advertisements) has been reached If there is no such field, no verification is performed.	102400
+ID	int32	No	File number.	3
<b>Complete Example</b>				
POST http://192.168.1.108/cgi-bin/VideoOutput.cgi?action=deliveryFile {				

```

"Port" : 0,
"Number" : 0
"TimeSection": [
    ["1 00:00:00-07:00:00",
    "1 09:00:00-17:30:00",
    "1 17:30:00-23:59:59"
    ], ..., []],
],
"Enable" : true,
"StartTime" : "2016-08-10 10:00:00",
"EndTime" : "2016-08-10 12:00:00",
"Mode" : "Once",
    "FileList" : [{
        "FileType" : "Video",
        "URL" : "ftp://192.168.1.108/1.dav",
        "Sustain" : 5
        "Size": 102400,
        "ID": 3
    }, {}]
}

```

## 15.9.5 Changing Stay Time of Advertisement Image

Change the stay time of advertisement image.

### 15.9.5.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=changeSustain&Sustain=5			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
Sustain	int	Yes	Stay time of each Image.	5
<b>Complete Example</b>				
http://<server>/cgi-bin/VideoOutput.cgi?action=changeSustain&Sustain=5				

## 15.9.6 Acquiring Video and Image Files Played by the Device

Acquire video and image files played by the device.

### 15.9.6.1 Request

<b>Syntax</b>	http://<server>/cgi-bin/VideoOutput.cgi?action=queryDeliveredFile			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>

<b>Complete Example</b>
http://<server>/cgi-bin/VideoOutput.cgi?action=queryDeliveredFile

## 15.9.6.2 Response

Parameter Format	key=value format at URL			
Parameter	Type	Required	Description	Example
Enable	bool	No	Enable play function.	true
Number	int	Yes	The current advertisement plan number. The caller can use this number to set up different advertisement plans.	0
TimeSection	TimeSchedule	Yes	Advertisement playing period. It is a two-dimensional array in which the first seven elements correspond to seven days of a week, and the eighth element corresponds to holidays, with a maximum of six periods per day. The eighth element can be left blank or entered as null, indicating that the holiday period is not supported.	[["1 00:00:00-07:00:00", "2 09:00:00-17:30:00", "3 17:30:00-23:59:59"], ..., [], ]
Name	char[128]	No	Advertisement name	"Happy holiday"
StartTime:	char[20]	No	Start time	"2016-08-10 10:00:00"
EndTime:	char[20]	No	End time	"2016-08-10 12:00:00"
Mode	char[16]	No	Video play mode. Once: Each file in the list is played once. Repeat: Files in the list are played repeatedly until end time. Alone: Cut-in play (playing alone).	"Once"
FileList	object[]	No	Video file list.	Up to 20 file lists are supported.

+Downloaded	bool	No	Check whether the file has been downloaded to the device.	true
+FileType	enumchar[16]	No	File type. enumchar[16]{ "Video": Video file "Image": Image file "Audio": Audio file }	"Video"
+LocalPath	char[128]	No	The path where the file is downloaded to the device.	"/mnt/appdata/Publish/1.dav"
+Sustain	int	No	Stay time of each Image, which is valid only when FileType is Image. Unit: Second.	5
+URL	char[512]	No	The resource address of the file, and the maximum length is 512 bytes. It has the same length as the VideoOutput.delivery File interface (used by video intercom and delivered at the same time).	"ftp://192.168.1.108/1.dav"
+TimeSection	TimeSchedule	No	Advertisement playing period. It is a two-dimensional array in which the first seven elements correspond to seven days of a week, and the eighth element corresponds to holidays, with a maximum of six periods per day. The eighth element can be left blank or entered as null, indicating that the holiday period is not supported.	[["1 00:00:00-07:00:00", "1 09:00:00-17:30:00", "1 17:30:00-23:59:59"], ..., [], ]
+Size	int	No	Files size. Unit: Byte.	102400
+ID	int	No	File number.	3

### Complete Example

```
Enable=true
Number=0
TimeSection[0][0]=6 00:00:00-23:59:59
TimeSection[0][1]=0 00:00:00-23:59:59
TimeSection[0][2]=0 00:00:00-23:59:59
TimeSection[0][3]=0 00:00:00-23:59:59
TimeSection[0][4]=0 00:00:00-23:59:59
TimeSection[0][5]=0 00:00:00-23:59:59
TimeSection[1][0]=6 00:00:00-23:59:59
TimeSection[1][1]=0 00:00:00-23:59:59
TimeSection[1][2]=0 00:00:00-23:59:59
TimeSection[1][3]=0 00:00:00-23:59:59
TimeSection[1][4]=0 00:00:00-23:59:59
TimeSection[1][5]=0 00:00:00-23:59:59
...
Name="Happy holiday"
StartTime="2016-08-10 10:00:00"
EndTime="2016-08-10 12:00:00"
Mode="Once"
FileList.Downloaded=true
FileList.FileType="Video"
FileList.LocalPath="/mnt/appdata/Publish/1.dav"
FileList.Sustain=5
FileList.URL="ftp://192.168.1.108/1.dav"
FileList.Size=102400
FileList.ID=3
```

## 16.1 Video Analyse APIs

### 16.1.1 People Counting

#### 16.1.1.1 Get Summary

Get summary information of people statistics in video.

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=getSummary			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	O	Video channel index which starts from 1, default is 1.	1
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=getSummary&channel=1				

<b>Response Params ( key=value format in body )</b>				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
summary	object	R	Summary information of people statistics in video .	
+Channel	int	R	Video channel index which starts from 0.	1
+RuleName	string	R	Rule type, it can be: <ul style="list-style-type: none"> <li><b>"ManNumDetection"</b>: count the people num in region, and the detail stat is in "InsideSubtotal" param.</li> <li><b>"NumberStat"</b>: count the people entering and leaving the region, and the detail stat is in "EnteredSubtotal" and "ExitedSubtotal"param.</li> </ul>	"NumberStat"
+EnteredSubtotal	object	O	People enter stat.	
++Total	int	R	Total enter num.	14
++Today	int	R	Today enter num.	0
++Hour	int	R	This hour enter num.	0
++TotalInTimeSection	int	O	Today enter num after call clearSectionStat.	0
+ExitedSubtotal	object	O	People leave stat.	
++Total	int	R	Total leave num.	32
++Today	int	R	Today leave num.	0
++Hour	int	R	This hour leave num.	0
++TotalInTime	int	O	Today leave num after call clearSectionStat.	0

Section				
+InsideSubtotal	object	O	People inside region stat.	
++Total	int	R	Today inside num.	65
++ManStayStat	array<object>	O	The entering and leaving stat of people that leaved.	
+++EnterTime	string	O	People enter time.	2012-01-04 00:00:00
+++ExitTime	string	O	People leave time.	2012-01-04 00:00:45

### Response Example

```
summary.Channel=0
summary.RuleName=NumberStat
summary.EnteredSubtotal.Today=0
summary.EnteredSubtotal.Total=14
summary.EnteredSubtotal.TotalInTimeSection=0
summary.ExitedSubtotal.Today=0
summary.ExitedSubtotal.Total=32
summary.ExitedSubtotal.TotalInTimeSection=0
summary.InsideSubtotal.Total=65
summary.InsideSubtotal.ManStayStat[0].EnterTime=2012-01-04 00:00:00
summary.InsideSubtotal.ManStayStat[0].ExitTime=2012-01-04 00:00:45
summary.InsideSubtotal.ManStayStat[1].EnterTime=2012-01-04 00:00:00
summary.InsideSubtotal.ManStayStat[1].ExitTime=2012-01-04 00:00:45
```

### 16.1.1.2 Query the Count of People

- **Start to find**

Start to find video stat info, in response, there is a token for further info finding process, and there is a totalCount shows how many data count(s).

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=startFind			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	O	Video channel index which starts from 1, default is 1.	1
condition	object	R	Find condition.	
+StartTime	string	R	Find time range start	2012-01-04 00:00:00
+EndTime	string	R	Find time range end	2012-01-04 00:02:00
+Granularity	string	R	The information granularity returned by the query requirements. The range is { Hour, Day, Week, Month, Season, Year } ( <b>Note: Most devices only support Hour, Day, Week.</b> )	Hour

+RuleType	string	O	Rule type, it can be: <b>NumberStat</b> , <b>ManNumDetection</b> . If omit, default is <b>NumberStat</b>	NumberStat
+MinStayTime	int	O	Valid when ruleType is <b>ManNumDetection</b> , report people stay over this minimal time.	20
+PlanID	int	O	The plan id, only valid for dome camera.	2
+PtzPresetId	int	O	The ptz preset index which starts from 1, only valid for dome camera. Note: new device should use PtzPresetId instead of PlanID.	1
+AreaID	Array<int>	O	The area id which starts from 1, max array size is 20.	[1,2]
+OtherRule	string	O	The other search rule, can be : AverageStayTime	AverageStayTime

#### Request Example

http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=startFind&channel=1&condition.StartTime=2011-01-01%2012:00:00&condition.EndTime=2011-01-10%2012:00:00&condition.Granularity=Hour&condition.RuleType=NumberStat&condition.MinStayTime=20&condition.AreaID[0]=2&condition.AreaID[1]=3

#### Response Params ( key=value format in body )

Name	Type	R/O	Description	Example
token	int	R	Token for this search, use this token to get result and stop search.	12345
totalCount	int	R	Number of find result.	56

#### Response Example

token=12345  
totalCount=56

- **Get the find result**

Get the find result of Video Stat info with channel, token, begin Number and count.

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=doFind			
<b>Method</b>	GET			
<b>Request Params ( key=value format in URL )</b>				
Name	Type	R/O	Description	Example
channel	int	O	Video channel index which starts from 1, default is 1. <b>NOTE:</b> must be the same as startFind.	1
token	int	R	Token for this search, use this token to get result and stop searching.	12345
beginNumber	int	R	The start count. It must be between 0 and totalCount -1.	0
count	int	R	The count of info for this query.	20
<b>Request Example</b>				
http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=doFind&channel=1&token=12345&beginNumber=0&count=20				

#### Response Params ( key=value format in body )

Name	Type	R/O	Description	Example
found	int	R	Number of result that return.	20
info	array<object>	R	Result information of video Stat.	
+Channel	int	R	Video channel index which starts from 0.	0
+PlanID	int	O	The plan id, only valid for dome camera.	2
+PtzPresetId	int	O	The ptz preset index which starts from 1, only valid for dome camera.	1
+AreaID	int	O	The area id which starts from 1.	3
+AverageStayTime	int	O	The average stay time, only valid when startFind with OtherRule param's value is "AverageStayTime".	20
+RuleName	string	R	Rule name, it can be: <ul style="list-style-type: none"> <li>"ManNumDetection": count people num in region, detail stat is in "InsideSubtotal"param</li> <li>"NumberStat": count people enter and leave region, detail stat is in "EnteredSubtotal"and "ExitedSubtotal"param.</li> </ul>	NumberStat
+StartTime	string	O	Find time range start	2012-01-04 00:00:00
+EndTime	string	O	Find time range end	2012-01-04 00:02:00
+EnteredSubtotal	int	O	Total enter num.	14
+ExitedSubtotal	int	O	Total leave num.	5
+InsideSubtotal	int	O	Total inside num.	65

#### Response Example

```

found=20
info[0].Channel=0
info[0].AreaID=2
info[0].RuleName=NumberStat
info[0].StartTime=2012-03-14 00:00:00
info[0].EndTime=2012-04-14 00:00:00
info[0].EnteredSubtotal=14
info[0].ExitedSubtotal=5
info[0].InsideSubtotal=65
info[1].Channel=0
info[1].AreaID=3
info[1].RuleName=NumberStat
info[1].StartTime=2012-03-14 00:00:00
info[1].EndTime=2012-04-14 00:00:00
info[1].EnteredSubtotal=14
info[1].ExitedSubtotal=5
info[1].InsideSubtotal=65
...

```

- **Stop the searching session**

Stop query video stat by channel and token.

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=stopFind			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	O	Video channel index which starts from 1, default is 1. <b>NOTE:</b> must be the same as startFind.	1
token	int	R	Token for this search, use this token to get result and stop searching.	12345
<b>Request Example</b>				
GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=stopFind&channel=1&token=12345				

<b>Response Params</b> ( OK in body )				
<b>Response Example</b>				
OK				

### 16.1.1.3 Clear the People Count Information

Clear the people count information.

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=clearSectionStat			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	O	Video channel index which starts from 1, default is 1.	1
<b>Request Example</b>				
GET http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=clearSectionStat&channel=1				

<b>Response Params</b> ( OK in body )				
<b>Response Example</b>				
OK				

### 16.1.1.4 Subscribe the People Count Information

Subscribe the people count information.

<b>Request URL</b>	http://<server>/cgi-bin/videoStatServer.cgi?action=attach			
<b>Method</b>	GET			
<b>Request Params</b> ( key=value format in URL )				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
channel	int	O	Video channel index which starts from 1, default is 1.	1
heartbeat	int	O	Send heartbeat interval, range is [1, 60],unit is	5

		second. If the URL contains this parameter, and the value is 5, it means every 5 seconds the device should send the heartbeat message to the client, the heartbeat message is an string "Heartbeat". If this parameter is not present, its default value is 60.
--	--	---

**Request Example**

http://192.168.1.108/cgi-bin/videoStatServer.cgi?action=attach&channel=1&heartbeat=5

**Response Params ( key=value format in body )**

Name	Type	R/O	Description	Example
summary	object	R	Summary information of video Stat.	
+Channel	int	R	Video channel index which starts from 0.	0
+RuleName	string	R	Rule type, it can be: <ul style="list-style-type: none"> <li><b>"ManNumDetection"</b>: Count people num in region, detail stat is in "InsideSubtotal" param.</li> <li><b>"NumberStat"</b>: Count people entering and leaving region, detail stat is in "EnteredSubtotal" and "ExitedSubtotal" param.</li> </ul>	NumberStat
+EnteredSubtotal	object	O	People enter stat.	
++Total	int	R	Total enter num.	14
++Today	int	R	Today enter num.	2
++Hour	int	R	This hour enter num.	1
++TotalInTimeSection	int	O	Today enter num after call clearSectionStat.	0
+ExitedSubtotal	object	O	People leave stat.	
++Total	int	R	Total leave num.	32
++Today	int	R	Today leave num.	3
++Hour	int	R	This hour leave num.	2
++TotalInTimeSection	int	O	Today leave num after call clearSectionStat.	0
+InsideSubtotal	object	O	People inside region stat.	
++Total	int	R	Today inside num.	65
++ManStayStat	array<object>	O	The entering and leaving stat of people that leaved.	
+++EnterTime	string	O	People enter time	2012-01-04 00:00:00
+++ExitTime	string	O	People leave time	2012-01-04 00:00:45

**Response Example**

HTTP/1.1 200 OK  
Server: Device/1.0  
Content-Type: multipart/x-mixed-replace; boundary=<boundary>  
Connection: closed

```

--<boundary>
Content-Type: text/plain
Content-Length: <length>

summary.Channel=0
summary.RuleName=NumberStat
summary.EnteredSubtotal.Today=2
summary.EnteredSubtotal.Total=14
summary.EnteredSubtotal.TotalInTimeSection=1
summary.ExitedSubtotal.Today=3
summary.ExitedSubtotal.Total=32
summary.ExitedSubtotal.TotalInTimeSection=2
summary.InsideSubtotal.Total=65
summary.InsideSubtotal.ManStayStat[0].EnterTime=2012-01-04 00:00:00
summary.InsideSubtotal.ManStayStat[0].ExitTime=2012-01-04 00:00:45
summary.InsideSubtotal.ManStayStat[1].EnterTime=2012-01-04 00:00:00
summary.InsideSubtotal.ManStayStat[1].ExitTime=2012-01-04 00:00:45
--<boundary>
Content-Type: text/plain
Content-Length: 11

Heartbeat
--<boundary>
Content-Type: text/plain
Content-Length: <length>

summary.Channel=0
summary.RuleName=NumberStat
summary.EnteredSubtotal.Today=2
summary.EnteredSubtotal.Total=14
...

```

### 16.1.1.5 [Event] NumberStat

Table 16-1

Usage	Refer to "4.9.17 Subscribe to Event Message" for how to subscribe event		
Description	When people number triggers the rule, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Number	int	R	Total number in detect region.
EnteredNumber	int	R	Total number that enter the detect region.
ExitedNumber	int	R	Total number that leave the detect region.
Type	string	R	Number overrun type, it can be: "EnterOver", "ExitOver", "InsideOver".

AreaID	int	O	The area id, begin from 1, if omit, means single area.
PresetID	int	O	The preset id, if omit, means preset is unknown.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=NumberStat;action=Start;index=0;data={   "Number": 120,   "EnteredNumber": 180,   "ExitedNumber": 60,   "Type": "EnterOver",   "Area": 2 } --&lt;boundary&gt;</pre>		

### 16.1.1.6 [Event] ManNumDetection

Table 16-2

Usage	Refer to "4.9.17 Subscribe to Event Message" for how to subscribe event		
Description	When people number triggers the rule, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
ManList	Array<object>	R	The people info list.
+BoundingBox	Array<int>	R	The detected people bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
+Stature	int	R	The people's stature, unit is cm.
AreaID	int	O	The area id, begin from 1, if omit, means single area.
PresetID	int	O	The preset id, if omit, means preset is unknown.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=ManNumDetection;action=pulse;index=0;data={   "ManList": [ { "BoundingBox": [2992,1136,4960,5192], "Stature": 170 },     { "BoundingBox": [4392,4136,6960,6512], "Stature": 175 },     {...}, ... ],   "AreaID" : 2 } --&lt;boundary&gt;</pre>		

### 16.1.1.7 [Event] CrowdDetection

Table 16-3

Usage	Refer to "4.9.17 Subscribe to Event Message" for how to subscribe event		
Description	When crowd density overrun, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
CrowdList	Array<object>	O	The global crowd density overrun list.
+Center	Array<int>	R	The center point, must be two int, means x and y value, coordinate remap to 0 — 8192.
+Radius	int	R	The radius length.
RegionList	Array<object>	O	The people num overrun region list.
+RegionID	int	R	The region index.
+PeopleNum	int	R	The people count in region.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=CrowdDetection;action=start;index=0;data={   "CrowdList": [ { "Center" : [5734,2377],"Radius" : 10}, ...{} ],   "RegionList" : [ { "RegionID" : 0, "PeopleNum" : 100 },...{} ] } --&lt;boundary&gt;</pre>		

### 16.1.2 FaceRecognitionServer

#### 16.1.2.1 Create Face Group

Table 16-4

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=createGroup		
Method	GET		
Description	Create a face group.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupName	string	R	The face group name, max string length is 127.
groupDetail	string	O	The description detail of the face group, max string length is 255.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the created face group, max string length is 63.

[ Example ]	
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=createGroup&groupName=Test1&groupDetail=ForTest1
Response	groupID=10000

### 16.1.2.2 Modify Face Group

Table 16-5

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=modifyGroup		
Method	GET		
Description	Modify a face group.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the face group, max string length is 63.
groupName	string	R	The name of the the face group, max string length is 127.
groupDetail	string	O	Description detail of the face group, max string length is 255.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=modifyGroup&groupID=10000&groupName=Test1&groupDetail=ForTest1		
Response	OK		

### 16.1.2.3 Delete Face Group

Table 16-6

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=deleteGroup		
Method	GET		
Description	Delete a face group.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the face group, max string length is 63.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=deleteGroup&groupID=10000		
Response	OK		

## 16.1.2.4 Deploy Face Group

There are two ways to deploy the group. One is based on the group (putDisposition), and the another one is based on the channel (setGroup).

- Put disposition to group

Table 16-7

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=putDisposition		
Method	GET		
Description	Deploy the face group to some video channels. If the video channel has been deployed already, it will change the similarity.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the face group, max string length is 63.
list	Array<object>	R	List of disposition info.
+channel	int	R	Video channel index which starts from 1.
+similarity	int	R	The threshold of the face similarity, 0 — 100.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
report	array<bool>	R	Result of putting disposition for each request channel.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=putDisposition&groupID=10000&list[0].channel=1&list[0].similarity=80&list[1].channel=2&list[1].similarity=70		
Response	report[0]=true report[1]=false		

- Delete some disposition from group

Table 16-8

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=deleteDisposition		
Method	GET		
Description	Remove the deployment of face group from some video channels.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the face group, max string length is 63.
channel	Array<int>	R	Video channel index which starts from 1.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
report	array<bool>	R	Result of deleting disposition for each request channel.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=deleteDisposition&groupID=10000&channel[0]=1&channel[1]=2		
Response	report[0]=true		

report[1]=false
-----------------

- set disposition group to channel

Table 16-9

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=setGroup		
Method	GET		
Description	Deploy some face groups to one video channel. If the video channel has been deployed already, it will change the similarity. <b>Note:</b> This method will do an overwrite operation.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
channel	int	R	Video channel index which starts from 1.
list	Array<object>	O	List of disposition info, if not exist, remove all disposition from channel.
+groupID	int	R	The identity of the face group, max string length is 63.
+similarity	int	R	The threshold of the face similarity, 0 — 100.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=setGroup&channel=1&list[0].groupID=10000&list[0].similarity=80&list[1].groupID=10002&list[1].similarity=75		
Response	OK		

- get disposition group from channel

Table 16-10

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=getGroup		
Method	GET		
Description	Get the Deployment about the video channel. <b>Note:</b> If the video channel does not deploy any group, then the response will be success with empty http body.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
channel	int	R	Video channel index which starts from 1.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
groupID	Array<int>	R	The identity of the face group, max string length is 63.
similarity	Array<int>	R	The threshold of the face similarity, 0 — 100.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=getGroup&channel=1		
Response	groupID[0]=10001 groupID[1]=10003 groupID[2]=10006 .... similarity[0]=80 similarity[1]=75		

similarity[2]=85 ....
--------------------------

### 16.1.2.5 Find Face Group

Table 16-11

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=findGroup		
Method	GET		
Description	Find the face group. If the groupID is not present in the URL, it will return all the groups.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	string	O	The identity of the face group, max string length is 63.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
GroupList	Array<object>	R	The face group information list.
+groupID	string	R	The identity of the face group, max string length is 63.
+groupName	string	R	Name of the face group, max string length is 127.
+groupDetail	string	O	Description detail of the face group, max string length is 255.
+groupSize	int	R	The number of face in this face group.
+channels	Array<int>	O	Video channel index which starts from 0.
+similarity	Array<int>	O	The threshold of the face similarity.
+groupType	string	O	The type of face group
+TimeSection	Array<Array<string>>	O	The time section of face group
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=findGroup		
Response	GroupList[0].groupID=00001 GroupList[0].groupName=Test1 GroupList[0].groupDetail=ForTest1 GroupList[0].groupSize=30 GroupList[0].channels[0]=1 GroupList[0].channels[1]=2 ... GroupList[0].similarity[0]=80 GroupList[0].similarity[1]=75 ... GroupList[0].groupType=BlackListDB GroupList[0].TimeSection[0][0]=1 00:00:00-23:59:59 GroupList[0].TimeSection[0][1]=0 00:00:00-23:59:59 GroupList[0].TimeSection[0][2]=0 00:00:00-23:59:59 GroupList[0].TimeSection[0][3]=0 00:00:00-23:59:59 GroupList[0].TimeSection[0][4]=0 00:00:00-23:59:59 GroupList[0].TimeSection[0][5]=0 00:00:00-23:59:59 GroupList[0].TimeSection[1][0]=1 00:00:00-23:59:59 GroupList[0].TimeSection[1][1]=0 00:00:00-23:59:59		

GroupList[0].TimeSection[1][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[1][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[1][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[1][5]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[2][0]=1 00:00:00-23:59:59  
GroupList[0].TimeSection[2][1]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[2][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[2][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[2][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[2][5]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[3][0]=1 00:00:00-23:59:59  
GroupList[0].TimeSection[3][1]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[3][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[3][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[3][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[3][5]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[4][0]=1 00:00:00-23:59:59  
GroupList[0].TimeSection[4][1]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[4][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[4][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[4][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[4][5]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[5][0]=1 00:00:00-23:59:59  
GroupList[0].TimeSection[5][1]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[5][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[5][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[5][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[5][5]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[6][0]=1 00:00:00-23:59:59  
GroupList[0].TimeSection[6][1]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[6][2]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[6][3]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[6][4]=0 00:00:00-23:59:59  
GroupList[0].TimeSection[6][5]=0 00:00:00-23:59:59

GroupList[1].groupID=00003  
GroupList[1].groupName=Test3  
GroupList[1].groupDetail=ForTest3  
GroupList[1].groupSize=50  
GroupList[1].channels[0]=1  
GroupList[1].channels[1]=2  
...  
GroupList[1].similarity[0]=70  
GroupList[1].similarity[1]=85

...

### 16.1.2.6 Re-Abstract Feature By Group

- Start ReAbstract

Table 16-12

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=groupReAbstract		
Method	GET		
Description	Abstract features for the groups. About the process of the re-extract, the device will use an event named <b>"FaceFeatureAbstract"</b> to report the process.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
groupID	Array<int>	R	The identity of the face group, max string length is 63.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
token	int	R	The identity of this operation.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=groupReAbstract&groupID[0]=10000&groupID[1]=10001		
Response	token=12345		

- Stop ReAbstract

Table 16-13

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopGroupReAbstract		
Method	GET		
Description	Stop the abstract features process.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
token	int	R	The identity of this operation.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=stopGroupReAbstract&token=12345		
Response	OK		

### 16.1.2.7 Add Person

Table 16-14

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=addPerson		
Method	POST		
Description	Add a person to a group.		

[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
groupID	string	R	The identity of the face group that this person to add. Max string length is 63.
name	string	R	The person name, max string length is 63.
birthday	string	O	The person's birthday, ex: "1980-01-01".
sex	string	O	Sex, it can be "Male", "Female", "Unknown".
country	string	O	The country name, length must be 2, and value should be according to ISO3166.
province	string	O	The province name, max string length is 63.
city	string	O	The city name, max string length is 63.
certificateType	string	O	The certificate type. It can be: "IC", "Passport", "Military", "Unknown".
id	string	O	The ID of certificate type, max string length is 31.
[ Response Params ] (key=value format )			
Name	Type	R/O	Param Description
uid	string	R	The id for this Person, max string length is 31.
[ Example ]			
Request	POST http://<server>/cgi-bin/faceRecognitionServer.cgi?action=addPerson&groupID=10000&name=ZhangSan&birthday=1980-01-05&sex=Male&country=CN&province=ZheZhang&city=Hangzhou HTTP/1.1 Content-Type: image/jpeg Content-Length: <image size>  <JPEG image data>		
Response	uid=0005		

### 16.1.2.8 Modify Person

Table 16-15

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=modifyPerson		
Method	POST		
Description	Modify a person's info. <b>Note:</b> If you do not want to change the image about the person, the request should not contain the image data. <b>Note:</b> You should provide at least one optional param to update.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
uid	string	R	The identity of the Person, max string length is 31.
groupID	string	R	The identity of the Face Group that this Person in. max string length is 63.
name	string	O	The person's name, max string length is 63.
birthday	string	O	The person's birthday, ex: "1980-01-01".
sex	string	O	Sex, it can be "Male", "Female", "Unknown".

country	string	O	The country name, length must be 2, and value should be according to ISO3166.
province	string	O	The province name, max string length is 63.
city	string	O	The city name, max string length is 63.
certificateType	string	O	The certificate type. It can be: "IC", "Passport", "Military", "Unknown".
id	string	O	The ID of certificate type, max string length is 31.
[ Response Params ] ( OK )			
[ Example ]			
Request	POST http://<server>/cgi-bin/faceRecognitionServer.cgi?action=modifyPerson&uid=0005&groupID=10000&name=ZhangSan&birthday=1980-01-05&sex=Male&country=CN&province=ZheZhang&city=Hangzhou HTTP/1.1 Content-Type: image/jpeg Content-Length: <image size>  <JPEG image data>		
Response	OK		

### 16.1.2.9 Delete Person

Table 16-16

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=deletePerson		
Method	GET		
Description	Delete a person from a group.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
uid	string	R	The identity of the person, max string length is 31.
groupID	string	R	The identity of the face group that this Person in. max string length is 63.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=deletePerson&uid=001&groupID=10000		
Response	OK		

### 16.1.2.10 Find Person

- Start to find

Table 16-17

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=startFind
Method	GET

Description	Start to find person in face groups. <b>Note:</b> the returned token will be expired after 60 seconds without any doFind call.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
condition	Object	R	Search scope condition.
+GroupID	Array<string>	R	The list of identity of the face group, max string length is 63.
person	object	O	Person condition.
+Name	string	O	Person Name, max string length is 63.
+Sex	string	O	Sex, it can be "Male", "Female", "Unknown".
+Country	string	O	Country name, length must be 2, and value should be according to ISO3166.
+Province	string	O	Province name, max string length is 63.
+City	string	O	City name, max string length is 63.
+CertificateType	string	O	Certificate Type. It can be: "IC", "Passport", "Military", "Unknown".
+ID	string	O	Person ID of CertificateType, max string length is 31.
+FeatureState	int	O	Feature State, 0:Unknown, 1:Failed, 2:OK.
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
totalCount	int	R	Result num, return -1 means still searching.
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=startFind&condition.GroupID[0]=10000&condition.GroupID[1]=10003&person.Sex=Male&person.Country=CN&person.FeatureState=1		
Response	token=123456789 totalCount=24		

- Get find result

Table 16-18

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=doFind		
Method	GET		
Description	Get one result of person's information from the search result set. <b>Note:</b> the returned token will be expired after 60 seconds without any doFind call.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
index	uint	R	The index in search result, should between 0 and totalCount -1.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
person	object	R	Person condition.
+UID	string	R	The identity of the person, max string length is 31.

+GroupID	string	R	The identity of the face group that this Person in. max string length is 63.
+Name	string	R	The person name, max string length is 63.
+Sex	string	O	Sex, it can be "Male", "Female", "Unknown".
+Birthday	string	O	The person's birthday, ex: "1980-01-01".
+Country	string	O	Country name, length must be 2, and value should be according to ISO3166.
+Province	string	O	Province name, max string length is 63.
+City	string	O	City name, max string length is 63.
+CertificateType	string	O	Certificate Type, can be: "IC", "Passport", "Military", "Unknown".
+ID	string	O	Person ID of CertificateType, max string length is 31.
+FeatureState	int	O	Feature State, 0:Unknown, 1:Failed, 2:OK.
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=doFind&token=123456789&index=0		
Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Content-Length: &lt;length&gt;  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  person.UID=0005 person.GroupID=10000 person.Name=ZhangSan person.Birthday=1980-01-01 person.Sex=Male person.Country=CN person.Province=ZheJiang person.City=HangZhou person.CertificateType=IC person.ID=1234567890 person.FeatureState=0 --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt; jpeg image data ... &gt; --&lt;boundary&gt;-- </pre>		

- Stop finding

Table 16-19

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFind		
Method	GET		
Description	Stop the search session.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	The token for this search, use this token to get result and stop search.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFind&token=123456789		
Response	OK		

### 16.1.2.11 Re-Abstract Features By Person

- Start ReAbstract

Table 16-20

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=reAbstract		
Method	GET		
Description	Abstract features for the persons. About the process of the re-extract, the device will use an event named " <b>FaceFeatureAbstract</b> " to report the process.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
UID	Array<int>	O	The list of identity of person, max string length is 31.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=reAbstract&uid[0]=001&uid[1]=002		
Response	OK		

- Stop ReAbstract

Table 16-21

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopReAbstract		
Method	GET		
Description	Stop the abstract features process.		
[ Request Params ] ( None )			
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/faceRecognitionServer.cgi?action=stopReAbstract		
Response	OK		

e	
---	--

### 16.1.2.12 [Config] Face Recognition AlarmOut Setting

- Get FaceRecognitionAlarm config

Table 16-22

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=FaceRecognitionAlarm		
Method	GET		
Description	Get the Face Recognition Alarm Out Setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	Config info table.
+FaceRecognitionAlarm	array<object>	R	Each face group has one config object in this array.
++GroupID	String	R	The face group ID, max string length is 63.
++GroupName	String	R	The face group name, max string length is 127.
++AlarmOutEnable	Bool	R	Enable AlarmOut or not.
++AlarmChannel	Array<object>	R	Each AlarmOut channel has one config object in this array.
+++AlarmRuleMask	Int	R	Alarm rule mask. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
+++AlarmOutLatCh	Int	R	Alarm out delay, unit is second, value between 1 and 300.
[ Example ]			
Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=FaceRecognitionAlarm		

Response	<pre> table.FaceRecognitionAlarm[0].GroupID=0017 table.FaceRecognitionAlarm[0].GroupName=wsd table.FaceRecognitionAlarm[0].AlarmOutEnable=true table.FaceRecognitionAlarm[0].AlarmChannel[0].AlarmRuleMask=0 table.FaceRecognitionAlarm[0].AlarmChannel[0].AlarmOutLatch=5 table.FaceRecognitionAlarm[0].AlarmChannel[1].AlarmRuleMask=0 table.FaceRecognitionAlarm[0].AlarmChannel[1].AlarmOutLatch=8 ... table.FaceRecognitionAlarm[1].GroupID=0018 table.FaceRecognitionAlarm[1].GroupName=cst table.FaceRecognitionAlarm[1].AlarmOutEnable=true table.FaceRecognitionAlarm[1].AlarmChannel[0].AlarmRuleMask=0 table.FaceRecognitionAlarm[1].AlarmChannel[0].AlarmOutLatch=10 table.FaceRecognitionAlarm[1].AlarmChannel[1].AlarmRuleMask=0 table.FaceRecognitionAlarm[1].AlarmChannel[1].AlarmOutLatch=15 ... </pre>
----------	--

- Set FaceRecognitionAlarm config

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set the Face Recognition Alarm Out Setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
FaceRecognitionAlarm	array<object>	R	Each face group has one config object in this array.
+GroupID	String	R	The face group ID, max string length is 63.
+GroupName	String	R	The face group name, max string length is 127.
+AlarmOutEnable	Bool	R	Enable AlarmOut or not.
+AlarmChannel	Array<object>	R	Each AlarmOut channel has one config object in this array.
++AlarmRuleMask	Int	R	Alarm rule mask. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
++AlarmOutLatch	Int	R	Alarm out delay, seconds, value between 1 and 300.
[ Response Params ] ( OK )			
[ Example ]			
Request	<pre> GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&amp;FaceRecognitionAlarm[0].GroupID=0017&amp;FaceRecognitionAlarm[0].GroupName=wsd&amp;FaceRecognitionAlarm[0].AlarmOutEnable=true&amp;FaceRecognitionAlarm[0].AlarmChannel[0].AlarmRuleMask=0&amp;FaceRecognitionAlarm[0].AlarmChannel[0].AlarmOutLatch=5 </pre>		
Response	OK		

### 16.1.2.13 Find Person by Picture

- Start to find

Table 16-23

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=startFindByPic		
Method	POST		
Description	<p>Start to find person in face groups by picture. The search may last for some time, so the response may push at regular intervals until 100% Progress</p> <p><b>Note:</b> If you want to find person in face groups by person info, see above "Find Person"API.</p> <p><b>Note:</b> The returned token will be expired after 60 seconds without any doFind call.</p>		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
GroupID	array<string>	R	Face groups to find, max string length is 63.
Similarity	int	R	Similarity percent, 1 — 100.
MaxCandidate	int	O	Max Candidate result number.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
token	uint	R	The token of this search, use this token to get result and stop search.
progress	uint	R	Search Progress, 100 means finished.
totalCount	int	R	Result num, return -1 means still searching.
[ Example ]			
Request	<pre>POST http://&lt;server&gt;/cgi-bin/faceRecognitionServer.cgi?action=startFindByPic&amp;GroupID[0]=0000 1&amp;GroupID[1]=00003&amp;Similarity=80&amp;MaxCandidate=500 Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;JPEG data&gt;</pre>		
Response	<pre>HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "token": 123456789,   "progress": 20   "totalCount": -1 }  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "token": 123456789,   "progress": 60</pre>		

<pre> "totalCount": -1 } --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  {"token": 123456789,  "progress": 100  "totalCount": 350 } --&lt;boundary&gt;-- </pre>
---

- Get the find result

Table 16-24

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=doFindByPic		
Method	GET		
Description	<p>Get the find result, reply using multipart format, first part is json string to describe all candidate person, then the following parts are the person's pictures, refer by UID and GroupID in part header Content-Info.</p> <p><b>Note:</b> the returned token will be expired after 60 seconds without any doFind call.</p>		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
index	uint	R	The index in search result, should between 0 and totalCount -1.
count	uint	R	Number of result person to get.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
Found	int	R	Number of result person that return.
Candidates	Array<object>	R	Candidates Person.
+Person	object	R	Person Info.
++UID	string	R	System id for this Person, max string length is 31.
++GroupID	string	R	The identity of the Face Group that this Person in. max string length is 63.
++Name	string	R	Person Name, max string length is 63.
++Birthday	string	O	Birthday ex: "1980-01-01".
++Sex	string	O	Sex, it can be "Male", "Female", "Unknown".
++Country	string	O	Country name, length must be 2, value should be according to ISO3166.
++Province	string	O	Province name, max string length is 63.
++City	string	O	City name, max string length is 63.
++CertificateType	string	O	Certificate Type. It can be "IC", "Passport", "Military" or "Unknown".
++ID	string	O	Person ID of CertificateType, max string length is 31.
++FeatureStat	int	O	Feature State, 0:Unknown, 1:Failed, 2:OK.

e			
++HomeAddress	string	O	Home Address, ex: "binanRoad NO1199"
+Similarity	int	R	Similarity.
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=doFindByPic&token=123456789&index=0&count=10		
Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "Found" : 10,   "Candidates" : [     {       "person" : {         "UID" : "0001",         "GroupID" : "001",         "Name" : "ZhangSan",         "Birthday" : "1980-01-05",         "Sex" : "Male",         ...       },       "Similarity" : 85     },     {       "person" : {         "UID" : "0002",         "GroupID" : "002",         "Name" : "LiSi",         "Birthday" : "1980-01-06",         "Sex" : "Male",         ...       },       "Similarity" : 80     },     { ... }, ...   ] }  --&lt;boundary&gt; Content-Info: UID=0001&amp;GroupID=001 Content-Type: image/jpeg </pre>		

<pre> Content-Length: &lt;length&gt;  &lt; jpeg image data ... &gt; --&lt;boundary&gt; Content-Info: UID=0002&amp;GroupID=002 Content-Type: image/jpeg Content-Length: &lt;length&gt;  &lt; jpeg image data ... &gt; --&lt;boundary&gt; ... </pre>
--

- Stop finding

Table 16-25

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFindByPic		
Method	GET		
Description	Stop the search session.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFindByPic&token=123456789		
Response	OK		

#### 16.1.2.14 Find History Person by Picture

- Start to find

Table 16-26

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=startFindHistoryByPic		
Method	POST		
Description	<p>Start to find person in capture history by picture. The search may last for some time, so the response may push at regular intervals until 100% Progress.</p> <p><b>Note:</b> If you want to find person in capture history by person info, please refer to "mediaFileFind"API.</p> <p><b>Note:</b> the returned token will be expired after 60 seconds without any doFind call.</p>		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Channel	int	R	Video channel index which starts from 0.
StartTime	string	R	Start time to search, ex: 2018-01-13T00:00:00Z.

EndTime	string	R	End time to search, ex: 2018-01-14T00:00:00Z.
Similarity	int	R	Similarity percent, 1 — 100.
MaxCandidate	int	O	Max Candidate result number.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
progress	uint	R	Search Progress, 100 means finished.
totalCount	int	R	Result num, return -1 means still searching.
[ Example ]			
Request	POST http://<server>/cgi-bin/faceRecognitionServer.cgi?action=startFindHistoryByPic&Channel=0&StartTime=2018-01-13T00:00:00Z&EndTime=2018-01-14T00:00:00Z&Type=All&Similarity=80&MaxCandidate=500 Content-Type: image/jpeg Content-Length: <image size>  <JPEG data>		

Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "token": 123456789,   "progress": 20   "totalCount": -1 }  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "token": 123456789,   "progress": 60   "totalCount": -1 }  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "token": 123456789,   "progress": 100   "totalCount": 350 }  --&lt;boundary&gt;-- </pre>
----------	--

- Get find result

Table 16-27

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=doFindHistoryByPic		
Method	GET		
Description	Get the find result, reply by multipart, first part is json string to describe all candidate person, then the following part is the person's picture, refer by UID in part header Content-Info. <b>Note:</b> the returned token will be expired after 60 seconds without any doFind call.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop search.
index	uint	R	The index in search result, should between 0 and totalCount -1.
count	uint	R	Number of result person to get, start from Index.
[ Response Params ] ( JSON format )			

Name	Type	R/O	Param Description
Found	int	R	Number of result person that return.
Candidates	Array<object>	R	Candidates Person.
+Person	object	R	Person Info.
++UID	string	R	System id for this Person, max string length is 63.
++Sex	string	O	Sex. It can be "Male", "Female" or "Unknown".
++Age	int	O	Age.
++Glasses	int	O	Glasses Status, 0: all, 1: not wear, 2: wear.
+Similarity	int	R	Similarity.
+Time	string	0	Appear time of the Person, format is "2013-09-02 00:00:00".

[ Example ]

Request	<pre>GET http://&lt;server&gt;/cgi-bin/faceRecognitionServer.cgi?action=doFindHistoryByPic&amp;token=12 3456789&amp;index=0&amp;count=12</pre>
Response	<pre>HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Connection: closed  --&lt;boundary&gt; Content-Type: application/json Content-Length: &lt;length&gt;  { "Found" : 12,   "Candidates" : [     {       "person" : {         "UID" : "0001",         "Sex" : "Male",         "Age" : 30,         "Glasses": 1       },       "Similarity" : 85       "Time": "2013-09-02 00:00:00"     },     {       "person" : {         "UID" : "0002",         "Sex" : "Male",         "Age" : 50,         "Glasses": 2       },       "Similarity" : 80       "Time": "2013-09-02 00:00:00"     },     { ... }, ...   ] }</pre>

```

}
--<boundary>
Content-Info: UID=0001
Content-Type: image/jpeg
Content-Length: <length>

< jpeg image data ... >
--<boundary>
Content-Info: UID=0002
Content-Type: image/jpeg
Content-Length: <length>

< jpeg image data ... >
--<boundary>
...

```

- Stop finding

Table 16-28

URL	http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFindHistoryByPic		
Method	GET		
Description	Stop finding.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
token	uint	R	Token for this search, use this token to get result and stop searching.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/faceRecognitionServer.cgi?action=stopFindHistoryByPic&token=123456789		
Response	OK		

### 16.1.2.15 [Event] FaceDetection

Table 16-29

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When the video channel disposition with some face group, and the video channel detect a face, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Faces	Array<object>	R	The detected faces info.

+BoundingBox	Array<int>	R	The detected face bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
+Sex	string	O	Sex, it can be "Man", "Woman".
+Age	int	O	Age.
+Feature	array<string>	O	Face feature, can be some of the following: "WearGlasses", "SunGlasses", "NoGlasses", "Smile", "Anger", "Sadness", "Disgust", "Fear", "Surprise", "Neutral", "Laugh", "Happy", "Confused", "Scream".
+Eye	int	O	Eye status, 0: not detected, 1: close eye, 2: open eye.
+Mouth	int	O	Mouth status, 0: not detected, 1: close mouth, 2: open mouth.
+Mask	int	O	Mask status, 0: not detected, 1: not wearing mask, 2: wearing mask.
+Beard	int	O	Beard status, 0: not detected, 1: no beard, 2: has beard.

[ Example ]

Event	<pre> --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=FaceDetection;action=Start;index=0;data={   "Faces": [ { "BoundingBox": [2992,136,6960,8192],     "Sex": "Man",     "Age": 40,     "Feature": [ "WearGlasses", "Smile"],     "Eye": 2,     "Mouth": 1,     "Mask": 1,     "Beard": 2   }, {...}, ... ] } --&lt;boundary&gt; </pre>
-------	---

### 16.1.2.16 [Event] FaceRecognition

Table 16-30

Usage	Refer to "4.4.3 Subscribe to Snapshot"for how to subscribe event		
Description	When the video channel disposition with some face group, and the video channel detect a face, after recognize in the face groups, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array.
+EventBaseInfo	object	R	Base info of event.
++Code	String	R	Event Code. It should be FaceRecognition.

++Action	String	R	Event Action. It can be "Start", "Stop" or "Pulse".
++Index	int	O	The channel index relate to this event, start from 0.
+UID	String	R	The identity of the Person, max string length is 31.
+Candidates	array<object>	O	The Candidates person list.
++Person	object	O	The information of candidate person.
+++UID	string	R	The identity of the Person, max string length is 31.
+++GroupID	string	R	The identity of the Face Group that this Person in. max string length is 63.
+++Name	string	O	Person Name, max string length is 63.
+++Birthday	string	O	Birthday ex: "1980-01-01".
+++Sex	string	O	Sex, it can be "Male", "Female", "Unknown".
+++Country	string	O	Country name. The length must be 2, and value should be according to ISO3166.
+++Province	string	O	Province name, max string length is 63.
+++City	string	O	City name, max string length is 63.
+++CertificateType	string	O	Certificate Type. It can be: "IC", "Passport", "Military", "Unknown".
+++ID	string	O	Person ID of CertificateType, max string length is 31.
++Similarity	int	O	Similarity of the Candidates person and the detected person, value between 1 — 100.
+Face	object	O	The attribute information of face.
++Sex	string	O	Sex, it can be "Man", "Woman".
++Age	int	O	Age.
++Feature	array<string>	O	Face feature, can be some of the following : "WearGlasses", "SunGlasses", "NoGlasses", "Smile", "Anger", "Sadness", "Disgust", "Fear", "Surprise", "Neutral", "Laugh", "Happy", "Confused", "Scream".
++Eye	int	O	Eye status, 0: not detected, 1: close eye, 2: open eye.
++Mouth	int	O	Mouth status, 0: not detected, 1: close mouth, 2: open mouth.
++Mask	int	O	Mask status, 0: not detected, 1: not wearing mask, 2: wearing mask.
++Beard	int	O	Beard status, 0: not detected, 1: no beard, 2: has beard.

[ Example ]

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo .Code=FaceRecognition Events[0].EventBaseInfo .Action=Pulse Events[0].EventBaseInfo .Index=0 Events[0].UID=00105 Events[0].Candidates[0].Person.UID=0012 Events[0].Candidates[0].Person.GroupID=10000 Events[0].Candidates[0].Person.Name=ZhangSan Events[0].Candidates[0].Person.Birthday=1980-01-02 Events[0].Candidates[0].Person.Sex=Male</pre>
-------	--

```

...
Events[0].Candidates[0].Similarity=80
Events[0].Candidates[1].Person.UID=0014
Events[0].Candidates[1].Person.GroupID=10000
Events[0].Candidates[1].Person.Name=Lisi
Events[0].Candidates[1].Person.Birthday=1980-01-05
Events[0].Candidates[1].Person.Sex=Male
...
Events[0].Candidates[1].Similarity=75
...
Events[0].Face.Sex=Man
Events[0].Face.Age=20
Events[0].Face.Feature[0]=SunGlasses
Events[0].Face.Feature[1]=Smile
Events[0].Face.Eye=2
Events[0].Face.Mouth=1
Events[0].Face.Mask=1
Events[0].Face.Beard=2
--<boundary>
Content-Type: image/jpeg
Content-Length: <image size>

<Jpeg image data>
--<boundary>

```

### 16.1.2.17 [Event] FaceFeatureAbstract

Table 16-31

Usage	Refer to "4.9.17 Subscribe to Event Message" for how to subscribe event		
Description	When Re-Abstract Feature By Group or By Person, the abstract progress detail will send in this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Infos	Array<object>	R	Abstrace detail Info, max size is 100.
+State	String	R	Abstract state, it can be : <ul style="list-style-type: none"> <li>• "Success" : Abstract success,</li> <li>• "False" : Failed to abstract;</li> <li>• "Process" : In Process;</li> </ul>
+Process	int	O	The abstract progress.
+UID	string	O	The identity of the person, max string length is 31.
+GroupID	string	O	The identity of the face group, max string length is 63.
[ Example ]			

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=FaceFeatureAbstract;action=Start;index=0;data={   "Infos": [ { "State": "Progress",               "Progress": 30,               "UID": "20005",               "GroupID": "10000"             }, {...}, ... ] } --&lt;boundary&gt;</pre>
-------	---

### 16.1.2.18 [Config] Face Recognition Event Handler Setting

- Get FaceRecognitionEventHandler config

Table 16-32

URL	http://<server>/cgi-bin/configManager.cgi?action=getConfig&name=FaceRecognitionEventHandler		
Method	GET		
Description	Get the Face Recognition Event Handler Setting.		
[ Request Params ] ( None )			
[ Response Params ] ( key=value format )			
Name	Type	R/O	Param Description
table	object	R	Config info table.
+FaceRecognitionEventHandler	array<object>	R	Each face group has one config object in this array.
++GroupID	String	R	The face group ID, max string length is 63.
++GroupName	String	R	The face group name, max string length is 127.
++EventEnableMask	int	R	Report event mask, 0 means not to report event. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
++RecordEnableMask	int	R	Record media file mask., 0 means not to record. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
++RecordLatch	Int	R	Record latch time, unit is second.
++SnapEnableMask	Int	R	Snap picture mask., 0 means not to snap picture <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
++MailEnableMask	Int	R	Send mail mask., 0 means not to send mail. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
[ Example ]			

Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=getConfig&name=FaceRecognitionEventHandler
Response	table.FaceRecognitionEventHandler[0].GroupID=0017 table.FaceRecognitionEventHandler[0].GroupName=wsd table.FaceRecognitionEventHandler[0].EventEnableMask=3 table.FaceRecognitionEventHandler[0].RecordEnableMask=0 table.FaceRecognitionEventHandler[0].RecordLatch=10 table.FaceRecognitionEventHandler[0].SnapEnableMask=3 table.FaceRecognitionEventHandler[0].MailEnableMask=0 ... table.FaceRecognitionEventHandler[1].GroupID=0018 table.FaceRecognitionEventHandler[1].GroupName=cst table.FaceRecognitionEventHandler[1].EventEnableMask=3 table.FaceRecognitionEventHandler[1].RecordEnableMask=0 table.FaceRecognitionEventHandler[1].RecordLatch=10 table.FaceRecognitionEventHandler[1].SnapEnableMask=3 table.FaceRecognitionEventHandler[1].MailEnableMask=0 ...

- Set FaceRecognitionEventHandler config

URL	http://<server>/cgi-bin/configManager.cgi?action=setConfig		
Method	GET		
Description	Set the Face Recognition Event Handler Setting.		
[ Request Params ] ( key=value format at URL )			
Name	Type	R/O	Param Description
FaceRecognitionAlarm	array<object>	R	Each face group has one config object in this array.
+GroupID	String	R	The face group ID, max string length is 63.
+GroupName	String	R	The face group name, max string length is 127.
+EventEnableMask	int	R	Report event mask, 0 means not to report event. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
+RecordEnableMask	int	R	Record media file mask., 0 means not to record. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
+RecordLatch	Int	R	Record latch time, unit is second.
+SnapEnableMask	Int	R	Snap picture mask., 0 means not to snap picture <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
+MailEnableMask	Int	R	Send mail mask., 0 means not to send mail. <ul style="list-style-type: none"> <li>• Bit 0 : recognition success</li> <li>• Bit 1 : recognition failed</li> </ul>
[ Response Params ] ( OK )			
[ Example ]			

Request	GET http://10.0.0.8/cgi-bin/configManager.cgi?action=setConfig&FaceRecognitionEventHandler[0].GroupID=0017&FaceRecognitionEventHandler[0].GroupName=wsd&FaceRecognitionEventHandler[0].EventEnableMask=3&FaceRecognitionEventHandler[0].RecordEnableMask=0&FaceRecognitionEventHandler[0].RecordLatch=10&FaceRecognitionEventHandler[0].SnapEnableMask=3&FaceRecognitionEventHandler[0].MailEnableMask=0
Response	OK

### 16.1.2.19 [Config] Face-ID Recognition Threshold

Request

<b>Template</b>	http://<server>/cgi-bin/configManager.cgi?action=setConfig&CitizenPictureCompareRule.Threshold=60			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format in URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
Threshold	uint8	Yes	Face-ID comparison threshold [1, 100]	60
<b>Example</b>				
http://192.168.1.108/cgi-bin/configManager.cgi?action=setConfig&CitizenPictureCompareRule.Threshold=60				

Response

<b>Parameter Format</b>	OK at body			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
<b>Example</b>	OK			

### 16.1.2.20 Exporting Face Database

The exported data is binary.

<b>Request URL</b>	http://<server>/cgi-bin/api/FaceLibImExport/export			
<b>Method</b>	POST			
<b>Request Params</b> (JSON format in body)				
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Description</b>	<b>Example</b>
groupID	char[64]	R	Person group ID	"10"
Password	char[64]	O	Unzip password	"abcd"
<b>Request Example</b>				
<pre>{   "groupID" : "10",   "password": "abcd" }</pre>				

```
}
```

Response Params (binary in body )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
HTTP/1.1 200 OK NTP Server Device/1.0 Content-Type: application/octet-stream Content-Length: <length>  < binary data >				

### 16.1.2.21 Importing Face Database

The imported data is binary.

<b>Request URL</b>	http://<server>/cgi-bin/api/FaceLibImExport/import			
<b>Method</b>	POST			
<b>Request Params (multipart in body)</b>				
Name	Type	R/O	Description	Example
password	int32	O	Unzip password	
<b>Request Example</b>				
POST http://<server>/cgi-bin/api/FaceLibImExport/import HTTP/1.1 User-Agent: client/1.0 Content-Type: multipart/x-mixed-replace; boundary=<boundary>  --<boundary> Content-Type: application/json Content-Length: 20  { "password": "abcd" } --<boundary> Content-Type: application/octet-stream Content-Length: 800  < binary data > --<boundary>				

Response Params ( JSON format in body )				
Name	Type	R/O	Description	Example
<b>Response Example</b>				
{}				

## 16.1.3 Video Analyse Event

### 16.1.3.1 [Event] LeftDetection

Table 16-33

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect some object left, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	R	The object that left.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=LeftDetection;action=pulse;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] } } --&lt;boundary&gt;</pre>		

### 16.1.3.2 [Event] TakenAwayDetection

Table 16-34

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect some object was taken away, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	R	The object that was taken away.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;</pre>		

	<pre>Code=TakenAwayDetection;action=pulse;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] } } --&lt;boundary&gt;</pre>
--	--

### 16.1.3.3 [Event] WanderDetection

Table 16-35

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect some object was wandering, send this event		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Objects	Array<object>	R	The objects that was wandering.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
Tracks	Array<Array<Array<int>>>	O	The object wandering tracks, array of polyline, one polyline for one object, polyline is array of points, point is array of two int, x's value and y's value. Coordinate remap to 0 — 8192.
DetectRegion	Array<Array<int>>	R	The detection region, the first array is point list, max item is 20, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=WanderDetection;action=start;index=0;data={   "Objects": [ { "BoundingBox": [2992,1136,4960,5192] },     { "BoundingBox": [4392,4136,6960,6512] },     {...}, ... ],   "Tracks": [ [ [1202, 576], [1456, 863], [1921, 1204], [2341, 1823], [3512, 2314] ],     [ [2214, 3412], [3153, 3674], [4512, 4213] ],     {...}, ... ],   "DetectRegion": [ [1292,3469], [6535,3373], ... ] } --&lt;boundary&gt;</pre>		

### 16.1.3.4 [Event] StayDetection

Table 16-36

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event
Description	When detect some object was stay, send this event

[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	O	The object that was stay.
+BoundingBox	Array<int>	O	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
Objects	array<object>	O	If detect several object, store in this array.
+BoundingBox	Array<int>	O	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
DetectRegion	Array<Array<int>>	O	The detection region, the first array is point list, max item is 20, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
AreaID	int	O	The area id, begin from 1, if omit, means single area.
PresetID	int	O	The preset id, if omit, means preset is unknown.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=StayDetection;action=start;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] },   "Objects": [ { "BoundingBox": [2992,1136,4960,5192] },     { "BoundingBox": [4392,4136,6960,6512] },     {...}, ... ],   "DetectRegion": [ [1292,3469], [6535,3373], ... ],   "AreaID" : 2 } --&lt;boundary&gt;</pre>		

### 16.1.3.5 [Event] HumanTrait

Table 16-37

Usage	Refer to "4.4.3 Subscribe to Snapshot "for how to subscribe event		
Description	When detect a human trait, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	O	Event info array.
+EventBaselInfo	object	R	Base info of event.
++Code	String	R	Event Code. It should be <b>HumanTrait</b> .
++Action	String	R	Event Action. It can be: "Start", "Stop", "Pulse".
++Index	int	O	The channel index relate to this event, start from 0.

+HumanAttributes	object	O	The human attributes.
++BoundingBox	Array<int>	O	The detected human bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
++Sex	string	O	Sex, can be "Man", "Woman", "Unknown".
++Age	int	O	Age.
++Angle	int	O	Angle, 0: unknown, 1: front, 2: side, 3: back.
++CoatColor	string	O	Coat color, can be: "White", "Orange", "Pink", "Black", "Red", "Yellow", "Gray", "Blue", "Green", "Purple", "Brown", "Sliver", "Darkviolet", "Maroon", "Dimgray", "Whitesmoke", "Darkorange", "Mistyrose", "Tomato", "Olive", "Gold", "Darkolivegreen", "Chartreuse", "Greenyellow", "Forestgreen", "Seagreen", "Chartreuse", "Deepskyblue", "Cyan", "Other".
++CoatType	int	O	Coat type, 0: unknown, 1: long sleeve, 2: short sleeve.
++TrousersColor	string	O	Trousers color, value can be that of CoatColor.
++TrousersType	int	O	Trousers type, 0: unknown, 1: long pants, 2: short pants, 3: skirt.
++HasHat	int	O	Has hat or not, 0: unknown, 1: not has hat, 2: has hat.
++HasBag	int	O	Has bag or not, 0: unknown, 1: not has bag, 2: has bag.
++HasUmbrella	int	O	Has umbrella or not, 0: unknown, 1: not has umbrella, 2: has umbrella.
++Bag	int	O	Bag type, 0: unknown, 1: handbag, 2: shoulder bag, 3: knapsack, 4: draw-bar box.
++UpperPattern	int	O	Upper clothes pattern, 0: unknown, 1: pure color, 2: stripe, 3: pattern, 4: gap, 5: grid.
++HairStyle	int	O	Hair style, 0: unknown, 1: long hair, 2: short hair, 3: ponytail, 4: updo, 5: hiddened.
++Cap	int	O	Cap style, 0: unknown, 1: normal cap, 2: helmet.
+FaceAttributes	object	O	If the human's face can be detected, find it's attributes.
++BoundingBox	Array<int>	O	The detected face bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
++Sex	string	O	Sex, it can be "Man", "Woman", "Unknown".
++Age	int	O	Age.
++Feature	array<string>	O	Face feature, can be some of the following : "WearGlasses", "SunGlasses", "NoGlasses", "Smile", "Anger", "Sadness", "Disgust", "Fear", "Surprise", "Neutral", "Laugh", "Happy", "Confused", "Scream".
++Eye	int	O	Eye status, 0: not detected, 1: close eye, 2: open eye.
++Mouth	int	O	Mouth status, 0: not detected, 1: close mouth, 2: open mouth.

++Mask	int	O	Mask status, 0: not detected, 1: not wearing mask, 2: wearing mask.
++Beard	int	O	Beard status, 0: not detected, 1: no beard, 2: has beard.
++Glass	Int	O	Glasses status, 0: unknown, 1: not wearing, 2: normal Glasses, 3: sun glasses, 4: black frame glasses.

[ Example ]

Event	<pre> --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo .Code=HumanTrait Events[0].EventBaseInfo .Action=Pulse Events[0].EventBaseInfo .Index=0 Events[0].HumanAttributes.BoundingBox[0]=1341 Events[0].HumanAttributes.BoundingBox[1]=2451 Events[0].HumanAttributes.BoundingBox[2]=4513 Events[0].HumanAttributes.BoundingBox[3]=4135 Events[0].HumanAttributes.Sex=Man Events[0].HumanAttributes.Age=30 Events[0].HumanAttributes.CoatColor=White Events[0].HumanAttributes.CoatType=1 Events[0].HumanAttributes.TrousersColor=Black Events[0].HumanAttributes.TrousersType=1 Events[0].HumanAttributes.HasHat=1 Events[0].HumanAttributes.HasBag=2 Events[0].FaceAttributes.BoundingBox[0]=1341 Events[0].FaceAttributes.BoundingBox[1]=2451 Events[0].FaceAttributes.BoundingBox[2]=4513 Events[0].FaceAttributes.BoundingBox[3]=4135 Events[0].FaceAttributes.Sex=Man Events[0].FaceAttributes.Age=30 Events[0].FaceAttributes.Feature[0]=Smile Events[0].FaceAttributes.Eye=2 Events[0].FaceAttributes.Mouth=1 Events[0].FaceAttributes.Glass=1 --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;Jpeg image data&gt; --&lt;boundary&gt; </pre>
-------	--

### 16.1.3.6 [Event] CrossLineDetection

Table 16-38

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect some object cross the line, send this event.		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	R	The object that cross the line.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
Objects	array<object>	O	If detect several object, store in this array.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
DetectLine	Array<Array<int>>	R	The detection line, the first array is point list, max item is 20, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
Direction	string	O	The crossline direction, can be : "LeftToRight", "RightToLeft", "Any".
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=CrossLineDetection;action=pulse;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] },   "DetectLine": [ [1292,3469], [6535,3373], ... ],   "Direction" : "LeftToRight" } --&lt;boundary&gt;</pre>		

### 16.1.3.7 [Event] CrossRegionDetection

Table 16-39

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect some object cross the region, send this event		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	R	The object that cross the region.
+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left-top point, x's value of right—bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
Objects	array<object>	O	If detect several object, store in this array.

+BoundingBox	Array<int>	R	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
DetectRegion	Array<Array<int>>	R	The detection region, the first array is point list, max item is 20, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
Action	string	R	The cross action, can be : "Appear", "Disappear", "Cross","Inside"
Direction	string	O	The cross direction, valid on if the "Action" is "Cross", can be : "Enter", "Leave", "Both".

[ Example ]

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=CrossRegionDetection;action=pulse;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] },   "DetectRegion": [ [1292,3469], [6535,3373], ... ],   "Action" : "Cross",   "Direction" : "Enter" } --&lt;boundary&gt;</pre>
-------	--

### 16.1.3.8 [Event] QueueStayDetection

Table 16-40

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event		
Description	When detect the queue stay time too long, send this event		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
Object	object	O	The object that was stay.
+BoundingBox	Array<int>	O	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
Objects	array<object>	O	If detect several object, store in this array.
+BoundingBox	Array<int>	O	The detected object bounding box, 4 interge, refer to x's value of left—top point, y's value of left—top point, x's value of right—bottom point, y's value of right—bottom point. Coordinate remap to 0 — 8192.
DetectRegion	Array<Array<int>>	O	The detection region, the first array is point list, max item is 20, the second array is point, must be two int, means x and y value, coordinate remap to 0 — 8192.
AreaID	int	O	The area id, begin from 1, if omit, means single area.

PresetID	int	O	The preset id, Valid ID starts from 1; 0 means meaningless, not involved. if omit, means preset is unknown.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=QueueStayDetection;action=start;index=0;data={   "Object": { "BoundingBox": [2992,1136,4960,5192] },   "Objects": [ { "BoundingBox": [2992,1136,4960,5192] },     { "BoundingBox": [4392,4136,6960,6512] },     {...}, ... ],   "DetectRegion": [ [1292,3469], [6535,3373], ... ],   "ArealD" : 2 } --&lt;boundary&gt;</pre>		

### 16.1.3.9 [Event] QueueNumDetection

Table 16-41

Usage	Refer to "4.9.17 Subscribe to Event Message" for how to subscribe event		
Description	When detect the queue people number exceed limit, send this event		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
ManList	Array<object>	R	The people info list.
+BoundingBox	Array<int>	R	The detected people bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
+Stature	int	R	The people's stature, unit is cm.
ArealD	int	O	The area id, begin from 1, if omit, means single area.
PresetID	int	O	The preset id, Valid ID starts from 1; 0 means meaningless, not involved. if omit, means preset is unknown.
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Code=QueueNumDetection;action=pulse;index=0;data={   "ManList": [ { "BoundingBox": [2992,1136,4960,5192], "Stature": 170 },     { "BoundingBox": [4392,4136,6960,6512], "Stature": 175 },     {...}, ... ],   "ArealD" : 2 } --&lt;boundary&gt;</pre>		

## 16.2 Intelligent Traffic APIs

### 16.2.1 Traffic Record

#### 16.2.1.1 Insert Traffic BlackList/RedList Record

Table 16-42

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=<RecordName>		
Method	GET		
Description	Insert traffic BlackList/Red record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name, " <b>TrafficBlackList</b> " for BlackList and " <b>TrafficRedList</b> " for RedList.
PlateNumber	string	R	The number of car plate, max string length is 31. It must be unique.
MasterOfCar	string	O	The car owner, max string length is 15
PlateColor	string	O	Plate color, max string length is 31, ex: "Yellow", "Blue", ... etc.
PlateType	string	O	Plate type, max string length is 31.
VehicleType	string	O	Vehicle type, max string length is 31.
VehicleColor	string	O	Vehicle color, max string length is 31. ex: "Yellow", "Blue", ... etc.
BeginTime	string	O	Begin time, ex: "2010-05-25 00:00:00".
CancelTime	string	O	Cancel time, ex: "2010-06-25 00:00:00".
AuthorityList	object	O	Authority list, only valid for " <b>TrafficRedList</b> " table.
+OpenGate	bool	O	Authority to open the gate.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
RecNo	int	R	The new record's id, return -1 if the device handles asynchronously.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=insert&name=TrafficBlackList&PlateNumber=AC00001&MasterOfCar=ZhangSan&PlateColor=Yellow&VehicleColor=Blue&BeginTime=2011-01-01%2012:00:00&CancelTime=2011-01-10%2012:00:00		
Response	RecNo=12345		

#### 16.2.1.2 Update Traffic BlackList/RedList Record

Table 16-43

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=update&name=<RecordName>
Method	GET

Description	Update Traffic BlackList/Red record. <b>Note:</b> Besides action, name, recno, there should be at least one more parameter to be updated.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name, " <b>TrafficBlackList</b> " for BlackList and " <b>TrafficRedList</b> "for RedList.
recno	int	R	record id
PlateNumber	string	R	The number of car plate, max string length is 31. It must be unique.
MasterOfCar	string	O	The car owner, max string length is 15
...<other param>	-	-	...<See above insert command for other params of the record. They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=update&name=TrafficBlackList&recno=12345&PlateNumber=AC00001&MasterOfCar=ZhangSan&PlateColor=Yellow&VehicleColor=Blue&BeginTime=2011-01-01%2012:00:00&CancelTime=2011-01-10%2012:00:00		
Response	OK		

### 16.2.1.3 Remove Traffic BlackList/RedList Record

Table 16-44

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=remove&name=<RecordName>		
Method	GET		
Description	Remove Traffic BlackList/Red record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name, " <b>TrafficBlackList</b> "for BlackList and " <b>TrafficRedList</b> "for RedList.
recno	int	R	The record id.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=remove&name=TrafficBlackList&recno=12345		
Response	OK		

### 16.2.1.4 Find Traffic BlackList/RedList Record

Table 16-45

URL	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=<RecordName>		
Method	GET		
Description	Find Traffic BlackList/RedList record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name, " <b>TrafficBlackList</b> "for BlackList and " <b>TrafficRedList</b> "for RedList.
count	int	O	Max result to return, default is 1024.
StartTime	string	O	The start of the record's CreateTime.
EndTime	string	O	The end of the record's CreateTime.
condition	object	O	Search condition.
+PlateNumber	string	O	Car plate number, max string length is 47.
+PlateNumberV ague	string	O	Car plate number substring, match any car plate number that contain this substring, max string length is 47.
+PlateNumberV agueGroup	array<string >	O	Car plate number substring array, max string length is 47.
+QueryCount	int	O	Query count, default is 1000.
+QueryResultBe gin	int	O	Begin number in the result set, default is 0.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
totalCount	int	O	Total record num that find.
found	int	O	Record num that returned.
records	array<object >	R	The records that returned.
+RecNo	int	R	Record id.
+CreateTime	int	R	The create time of record.
+PlateNumber	string	R	Car plate number.
+MasterOfCar	string	O	Car owner.
+...<other param>	-	-	...<see above insert command for other params of the record>
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=TrafficBlackList&condition. PlateNumber=AC00001&StartTime=123456700&EndTime=123456800&count=100		
Response	totalCount=1000 found=100 records[0].RecNo=12345 records[0].CreateTime=123456789 records[0].PlateNumber=AC00001 records[0].MasterOfCar=ZhangSan ... records[1].RecNo=13579 records[1].CreateTime=123456799 records[1].PlateNumber=AC00001 records[1].MasterOfCar=LiSi		

...
-----

### 16.2.1.5 RemoveEx Traffic BlackList/RedList Record

Table 16-46

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=removeEx&name=<RecordName>		
Method	GET		
Description	RemoveEx Traffic BlackList/Red record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name, " <b>TrafficBlackList</b> "for BlackList and " <b>TrafficRedList</b> "for RedList.
RecNo	int	O	The record id.
PlateNumber	string	O	The number of car plate, max string length is 31. It must be unique.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=removeEx&name=TrafficBlackList &PlateNumber=AC00001		
Response	OK		

## 16.2.2 Intelligent Traffic Event

### 16.2.2.1 [Event] TrafficJunction

Table 16-47

Usage	Refer to "4.4.3 Subscribe to Snapshot"for how to subscribe event		
Description	When detects vehicle passing, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array.
+EventBaseInfo	object	R	Base info of event.
++Code	String	R	Event Code. It should be <b>TrafficJunction</b> .
++Action	String	R	Event Action. It can be "Start", "Stop" or "Pulse".
++Index	int	O	The channel index relate to this event.
+GroupID	int	O	The id of event group.
+CountInGroup	int	O	Event count in the event group.
+IndexInGroup	int	O	The index of this event in the event group, start from 1.
+Lane	int	O	Lane number, start from 0.
+TriggerType	int	O	Trigger type. It can be: 0—car detector, 1—radar, 2—

			video.
+Speed	int	O	Vehicle speed, unit is km/hour.
+Vehicle	Object	O	The information of vehicle object.
++BoundingBox	Array<int>	R	The detected car bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
++Text	string	O	Vehicle logo.
++SubText	string	O	Vehicle sub logo.
++SubBrand	int	O	Vehicle sub brand index.
++BrandYear	int	O	Vehicle brand year index.
+TrafficCar	object	O	Traffic Car info.
++RecNo	int	R	The record id.
++PlateNumber	string	R	Car plate number.
++PlateType	string	O	Plate type.
++PlateColor	string	O	Plate color, ex: "Yellow", "Blue", ... etc.
++VehicleColor	string	O	Vehicle color, ex: "Yellow", "Blue", ... etc.
++BoundingBox	Array<int>	R	The detected plate bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
++Country	string	O	Country info. Max string length is 19.
++Speed	int	O	Vehicle speed, unit is km/hour.
++Event	string	O	The event info, ex: "TrafficJunction".
+CommInfo	object	O	Traffic event common info.
++Seat	array<object>	O	Vehicle front seat info.
+++Type	string	O	Front seat type. It can be: "Main""Slave".
+++Status	array<string>	O	Some driver status. It can be some of the following: "Smoking", "Calling".
+++SunShade	string	O	Sunshade status. It can be: "Unknow", "WithSunShade", "WithoutSunShade".
+++ShadePos	Array<int>	O	The detected sun shade bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
+++SafeBelt	string	O	SafeBelt status. It can be: "Unknow", "WithSafeBelt", "WithoutSafeBelt".

[ Example ]

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code=TrafficJunction Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].GroupID=123</pre>
-------	--

```

Events[0].CountInGroup=3
Events[0].IndexInGroup=1
Events[0].Lane=0
Events[0].Vehicle.BoundingBox[0]=1341
Events[0].Vehicle.BoundingBox[1]=2451
Events[0].Vehicle.BoundingBox[2]=4513
Events[0].Vehicle.BoundingBox[3]=4135
Events[0].Vehicle.Text=Audi
Events[0].Vehicle.SubText=A6L
Events[0].Vehicle.SubBrand=5
Events[0].Vehicle.BrandYear=2
Events[0].TrafficCar.RecNo=123
Events[0].TrafficCar.PlateNumber=AC00003
Events[0].TrafficCar.PlateColor=Yellow
Events[0].TrafficCar.VehicleColor=Blue
Events[0].TrafficCar.BoundingBox[0]=1341
Events[0].TrafficCar.BoundingBox[1]=2451
Events[0].TrafficCar.BoundingBox[2]=4513
Events[0].TrafficCar.BoundingBox[3]=4135
Events[0].TrafficCar.Country=China
Events[0].CommInfo.Seat[0].Type=Main
Events[0].CommInfo.Seat[0].Status[0]=Smoking
Events[0].CommInfo.Seat[0].SunShade=WithSunShade
Events[0].CommInfo.Seat[0].ShadePos[0]=2021
Events[0].CommInfo.Seat[0].ShadePos[1]=3041
Events[0].CommInfo.Seat[0].ShadePos[2]=2151
Events[0].CommInfo.Seat[0].ShadePos[3]=3661
Events[0].CommInfo.Seat[0].SafeBelt=WithoutSafeBelt
...
--<boundary>
Content-Type: image/jpeg
Content-Length: <image size>

<Jpeg image data>
--<boundary>

```

### 16.2.2.2 [Event] TrafficRetrograde

When detects vehicle retrograde, send this event.

Event params is the same as **TrafficJunction**, except for event Code is **TrafficRetrograde**.

### 16.2.2.3 [Event] TrafficJam

Table 16-48

Usage	Refer to "4.4.3 Subscribe to Snapshot" for how to subscribe event
-------	---

Description	When detects traffic jam, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array.
+EventBaseInfo	object	R	Base info of event.
++Code	String	R	Event Code. It should be <b>TrafficJam</b> .
++Action	String	R	Event Action. It can be "Start", "Stop" or "Pulse".
++Index	int	O	The channel index relate to this event.
+GroupID	int	O	The id of event group.
+CountInGroup	int	O	Event count in the event group.
+IndexInGroup	int	O	The index of this event in the event group, start from 1.
+Lane	int	O	Lane number, start from 0.
+StartJaming	int	O	Start jam time, UTC seconds.
+AlarmInterval	int	O	Alarm interval, unit is second.
+JamLenght	int	O	Jam length, percentage of the lane.
+JamRealLenght h	int	O	Jam real length, unit is metre.
[ Example ]			
Event	<pre> --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code=TrafficJam Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].GroupID=123 Events[0].CountInGroup=3 Events[0].IndexInGroup=1 Events[0].Lane=0 Events[0].StartJaming=123456789 Events[0].AlarmInterval=180 Events[0].JamLenght=70 Events[0].JamRealLength=120 ... --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;Jpeg image data&gt; --&lt;boundary&gt; </pre>		

#### 16.2.2.4 [Event] TrafficUnderSpeed

When detects vehicle under speed, send this event.

Event params is the same as **TrafficJunction**, except for event Code is **TrafficUnderSpeed**, and add following params:

Table 16-49

[ Extra Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
+SpeedLimit	array<int>	O	Speed limit, 2 integer, min speed and max speed.
+UnderSpeeding Percentage	int	O	Percentage of under speed.

### 16.2.2.5 [Event] TrafficOverSpeed

When detects vehicle over speed, send this event.

Event params is the same as **TrafficJunction**, except for event Code is **TrafficOverSpeed**, and add following params:

Table 16-50

[ Extra Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
+SpeedLimit	array<int>	O	Speed limit, 2 integer, min speed and max speed.
+SpeedingPercentage	int	O	Percentage of over speed.

### 16.2.2.6 [Event] TrafficPedestrian

Table 16-51

Usage	Refer to "4.4.3 Subscribe to Snapshot"for how to subscribe event		
Description	When detects human in the lane, send this event		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array.
+EventBaseInfo	object	R	Base info of event.
++Code	String	R	Event Code. It should be <b>TrafficPedestrian</b> .
++Action	String	R	Event Action. It can be "Start", "Stop" or "Pulse".
++Index	int	O	The channel index relate to this event.
+GroupID	int	O	The id of event group.
+CountInGroup	int	O	Event count in the event group.
+IndexInGroup	int	O	The index of this event in the event group, start from 1.
+Lane	int	O	Lane number, start from 0.
+Vehicle	object	O	Actually this is the human that detected, not vehicle.
++Category	string	R	Must be <b>"Passerby"</b> , means this actually is a human.
++BoundingBox	Array<int>	R	The detected car bounding box, 4 interge, refer to x's value of left-top point, y's value of left-top point, x's value of right-bottom point, y's value of right-bottom point. Coordinate remap to 0 — 8192.
[ Example ]			

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code=TrafficPedestrian Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].GroupID=123 Events[0].CountInGroup=3 Events[0].IndexInGroup=1 Events[0].Lane=0 Events[0].Vehicle.Category=Passerby Events[0].Vehicle.BoundingBox[0]=1341 Events[0].Vehicle.BoundingBox[1]=2451 Events[0].Vehicle.BoundingBox[2]=4513 Events[0].Vehicle.BoundingBox[3]=4135 ... --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;Jpeg image data&gt; --&lt;boundary&gt;</pre>
-------	--

### 16.2.2.7 [Event] TrafficParking

When detects vehicle illegal parking, send this event.

Event params is the same as **TrafficJunction**, except for event Code is **TrafficParking**, and remove param "**speed**", add following params:

Table 16-52

[ Extra Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
+StartParking	int	O	The start time of parking, UTC seconds.
+AlarmInterval	int	O	The alarm interval.
+ParkingAllowed Time	int	O	The allowed time of parking.

## 16.2.3 Traffic Flow

### 16.2.3.1 [Event] TrafficFlowStat

Table 16-53

Usage	Refer to "4.9.17 Subscribe to Event Message"for how to subscribe event
-------	--

Description	When traffic flow trigger the rule, send this event		
[ Event Params ] ( JSON format )			
Name	Type	R/O	Param Description
FlowStates	array<object>	R	Traffic flow info, each object in list is traffic flow info about one lane.
+Lane	int	R	Lane number, start from 0.
+Flow	int	R	Traffic flow number.
+Period	int	R	Traffic stat time, unit is minute.
+PeriodByMili	int	O	Traffic stat time, unit is millisecond, value should between 0 and 59999.
+DrivingDirection	array<string>	O	Driving direction, should be an array of three strings: <ul style="list-style-type: none"> <li>• 1st string: direction, can be: "<b>Approach</b>", "<b>Leave</b>"</li> <li>• 2nd string: Approach position name</li> <li>• 3rd string: Leave position name</li> </ul>
[ Example ]			
Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: xxxx  Code=TrafficFlowStat;action=Pulse;index=0;data={   "FlowStates": [ { "Lane": 0,     "Flow": 50,     "Period": 5,     "DrivingDirection": [ "Approach", "Hangzhou", "Shanghai" ]   }, {...}, ... ] } --&lt;boundary&gt;</pre>		

### 16.2.3.2 Find Traffic Flow History

Table 16-54

URL	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=TrafficFlow		
Method	GET		
Description	Find traffic flow history record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The name of record table. It should be " <b>TrafficFlow</b> ".
count	int	O	Max result to return, default is 1024.
StartTime	string	O	The start of the record's CreateTime.
EndTime	string	O	The End of the record's CreateTime.
condition	object	O	Search condition.
+Channel	int	O	Video channel index which starts from 0.
+Lane	int	O	Lane index, starts from 0.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description

totalCount	int	O	Total record num that find.
found	int	O	Record num that returned.
records	array<object>	R	The records that returned.
+RecNo	int	R	Record id.
+CreateTime	int	R	The create time of record.
+StatisticsTime	int	R	Traffic flow statistics time, UTC seconds.
+Period	string	R	Statistics period, unit is second.
+Channel	int	R	Video channel index which starts from 0.
+Lane	int	R	Lane index, starts from 0.
+Vehicles	int	R	Total vehicle num.
+AverageSpeed	float	O	Average speed of the vehicle, -1 means no vehicle, 0 means congestion.

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=TrafficFlow&condition.Channel=0&condition.Lane=0&StartTime=123456700&EndTime=123456800&count=100
Response	totalCount=1000 found=100 records[0].RecNo=12345 records[0].CreateTime=123456789 records[0].StatisticsTime=123456789 records[0].Period=300 records[0].Channel=0 records[0].Lane=0 records[0].Vehicles=250 records[0].AverageSpeed=25.4 ... records[1].RecNo=13579 records[1].CreateTime=123456799 records[1].StatisticsTime=123456799 records[1].Period=300 records[1].Channel=0 records[1].Lane=0 records[1].Vehicles=220 records[1].AverageSpeed=21.8 ...

### 16.2.3.3 Start Traffic Statistics Search

Start Traffic Statistics Search

Request

<b>Syntax</b>	http://<server>/cgi-bin/trafficFlowStat.cgi?action=startFind			
<b>Method</b>	POST			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
StartTime	string	Yes	Start time, accurate to	"2010-05-25 00:00:00"

			hour	
EndTime	string	Yes	End time, accurate to hour	"2010-05-25 23:59:59"
Channel	int	No	Channel number	0
Lane	int	No	Lane number	0
Lanes	int[]	No	Lane number array, supports data search by multiple lanes	[0]
ClassType	int	No	As video metadata and road monitoring use the same database table, they need to be distinguished according to their types (0 represents video metadata and 1 represents road monitoring).	0
Granularity	enumchar[8]	No	The granularity of statistical information that needs to be returned in the searching process. This needs to be supported by getCaps. Enumchar[8]{ "Minute": By 5 minutes (hourly report) "Hour": By hour (daily report) "Day": By day (monthly report) "Month": By month (yearly report) } The default value is Hour.	Hour
PresetID	int[]	No	Preset	[1]
Direction	enumchar[16]	No	Search by statistical direction enumchar[16]{ "Both" "LeftToRight" "RightToLeft" }	Both
<b>Example</b>				
POST /cgi-bin/trafficFlowStat.cgi?action=startFind HTTP/1.1				
<pre>{   "StartTime" : "2010-05-25 00:00:00",   "EndTime" : "2010-05-25 23:59:59",</pre>				

```

    "Channel" : 0,
    "Lane" : 0,
    "Lanes" : [0],
    "ClassType": 1,
    "Granularity" : "Hour"
    "PresetID":[1],
    "Direction" : "Both"
}

```

Response

Parameter Format	key=value format at URL			
Parameter	Type	Required	Description	Example
token	int	Yes	Fetches search token	2342343
totalCount	int	Yes	Total count of results that meet the search criteria	100
<b>Example</b>				
token=2342343 totalCount=100				

### 16.2.3.4 Get Traffic Statistics

Get Traffic Statistics

Request

Syntax	http://<server>/cgi-bin/trafficFlowStat.cgi?action=doFind			
Method	GET			
Parameter Format	key=value format at URL			
Parameter	Type	Required	Description	Example
token	int	Yes	Search token.	46878
beginNumber	int	Yes	Search begin number. That is to start from the beginNumber records, fetch the count records, and return. 0<=beginNumber<=totalCount-1	0
count	int	Yes	Number of traffic statistic entries per search	24
<b>Example</b>				
http://192.168.1.108/cgi-bin/trafficFlowStat.cgi?action=doFind&token=46878&beginNumber=0&count=24				

Response

Parameter Format	key=value format at URL			
Parameter	Type	Required	Description	Example
found	int	Yes	Number of entries that are searched	100
info	int	Yes	Traffic statistics (For details, see traffic-flow statistics status information structure in wiki)	[, ... ]

<b>Example</b>
found=100 info=<info>

### 16.2.3.5 End Traffic Statistics Search

End Traffic Statistics Search

Request

<b>Syntax</b>	http://<server>/cgi-bin/trafficFlowStat.cgi?action=stopFind			
<b>Method</b>	GET			
<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
token	uint	Yes	Search token.	46878
<b>Example</b>	http://192.168.1.108/cgi-bin/trafficFlowStat.cgi?action=stopFind&token=46878			

Response

<b>Parameter Format</b>	key=value format at URL			
<b>Parameter</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>	<b>Example</b>
	int	Yes		
<b>Example</b>	ok			

## 16.3 Access Control APIs

### 16.3.1 Access User

#### 16.3.1.1 Add Access User Face

Table 16-55

<b>URL</b>	http://<server>/cgi-bin/FaceInfoManager.cgi?action=add		
<b>Method</b>	POST		
<b>Description</b>	Add an access user face.		
<b>[ Request Params ] ( JSON format in body )</b>			
<b>Name</b>	<b>Type</b>	<b>R/O</b>	<b>Param Description</b>
UserID	string	R	The user's id.
Info	object	R	The user's info.
+UserName	string	O	The user's name.
+RoomNo	array<string>	O	The user's room num list, max array size is 32, max string size is 12. ( This param is not supported by access control. )
+FaceData	array<string>	O	The user's face feature, encode with base64, max array size is 20, and max string size is 2k.

			<b>Note:</b> There must be at least one between FaceData and PhotoData.
+PhotoData	array<string>	O	The user's face photo, encode with base64, max array size is 5, and max string size is 200k. <b>Note:</b> There must be at least one between FaceData and PhotoData.
[ Response Params ] ( OK )			
[ Example ]			
Request	POST http://<server>/cgi-bin/FacInfoManager.cgi?action=add Content-Type: application/json Content-Length: <length>  <pre>{ "UserID": "102",   "Info": {     "UserName": "ZhangSan",     "RoomNo": [ "301", "303", ... ],     "FaceData": [ "xxxx", "xxxx", ... ],     "PhotoData": [ "yyyy", "yyyy", ... ]   } }</pre>		
Response	OK		

### 16.3.1.2 Modify Access User Face

Table 16-56

URL	http://<server>/cgi-bin/FacInfoManager.cgi?action=update		
Method	POST		
Description	Update an access user face.		
[ Request Params ] ( JSON format in body )			
Name	Type	R/O	Param Description
UserID	string	R	The user's id.
Info	object	R	The user's info.
+UserName	string	O	The user's name.
+RoomNo	array<string>	O	The user's room num list, max array size is 32, max string size is 12. ( This param is not supported by access control. )
+FaceData	array<string>	O	The user's face feature, encode with base64, max array size is 20, and max string size is 2k. <b>Note:</b> There must be at least one between FaceData and PhotoData.
+PhotoData	array<string>	O	The user's face photo, encode with base64, max array size is 5, max string size is 200k. <b>Note:</b> There must be at least one between FaceData and PhotoData.
[ Response Params ] ( OK )			
[ Example ]			

Request	<pre> POST http://&lt;server&gt;/cgi-bin/FaceInfoManager.cgi?action=update Content-Type: application/json Content-Length: &lt;length&gt;  { "UserID": "102",   "Info": {     "UserName": "ZhangSan",     "RoomNo": ["301", "303", ...],     "FaceData": ["xxxx", "xxxx", ...],     "PhotoData": ["yyyy", "yyyy", ...]   } } </pre>
Response	OK

### 16.3.1.3 Delete Access User Face

Table 16-57

URL	http://<server>/cgi-bin/FaceInfoManager.cgi?action=remove		
Method	GET		
Description	Delete an access user face.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
UserID	string	R	The user's id.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://<server>/cgi-bin/FaceInfoManager.cgi?action=remove&UserID=102		
Response	OK		

### 16.3.1.4 Find Access User Face

- Start to find access user face

Table 16-58

URL	http://<server>/cgi-bin/FaceInfoManager.cgi?action=startFind		
Method	GET		
Description	Start to find access user face.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Condition	object	O	Search condition.
+UserID	string	O	The user's id.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
Token	int	O	The token of this search, use this token to get result and stop

			search.
Total	int	O	Result num, return 0 if not found.
[ Example ]			
Request	GET http://<server>/cgi-bin/FaceInfoManager.cgi?action=startFind		
Response	<pre>{ "Token": 1234,   "Total": 20 }</pre>		

- Get the find result

Table 16-59

URL	http://<server>/cgi-bin/FaceInfoManager.cgi?action=doFind		
Method	GET		
Description	Get the find result.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Token	int	R	The token of this search, use this token to get result and stop search.
Offset	int	R	Offset in the result record set, range is [0, Total – 1].
Count	int	R	Count of result to get.
[ Response Params ] ( JSON format )			
Name	Type	R/O	Param Description
Info	array<object>	R	The face information of user.
+UserID	string	R	The user's id.
+MD5	array<string>	O	The user's face photo's MD5 hash string, max array size is 5, max string size is 33.
[ Example ]			
Request	GET http://<server>/cgi-bin/FaceInfoManager.cgi?action=doFind&Token=1234&Offset=0&Count=20		
Response	<pre>{ "Info": [ {   "UserID": "102",   "MD5": [ "xxxx", "xxxx", ... ] }, { ... }, ... ] }</pre>		

- Stop the find session

Table 16-60

URL	http://<server>/cgi-bin/FaceInfoManager.cgi?action=stopFind		
Method	GET		
Description	Stop the find session.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
Token	int	R	Token for this search, use this token to get result and stop search.

[ Response Params ] ( OK )	
[ Example ]	
Request	GET http://<server>/cgi-bin/FaceInfoManager.cgi?action=stopFind&Token=1234
Response	OK

### 16.3.1.5 Add Access User Card and Fingerprint

- Add Access user card only

Table 16-61

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=AccessControlCard		
Method	GET		
Description	Insert access user card info.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be "AccessControlCard".
CardName	string	R	Access user card name, max string length is 32.
CardNo	string	R	Access user card number.
UserID	string	R	The user's id.
CardStatus	int	R	Card status, 0 means normal, and abnormal status can be sum of following: <ul style="list-style-type: none"> <li>• 1 &lt;&lt; 0: report the loss of the card</li> <li>• 1 &lt;&lt; 1: the card is withdrawn</li> <li>• 1 &lt;&lt; 2: the card is frozen</li> <li>• 1 &lt;&lt; 3: the card is arrearage</li> <li>• 1 &lt;&lt; 4: the card exceeds the time limit</li> <li>• 1 &lt;&lt; 5: the card is arrearage but still can open the door, and there will be voice prompts.</li> </ul>
CardType	int	O	The card type: 0 : Normal Card, 1: VIP Card, 2: Visitor Card, 3: Patrol Card, 4: Blacklist Card, 5: Stress Card, 0xff: Mother Card
Password	string	O	The Access card's password. ( This param is not supported by video talk device )
Doors	array<int>	O	The index of the doors that this card can open. ( This param is not supported by video talk device )
TimeSections	array<int>	O	The index of the time sections of each door that this card can open. ( This param is not supported by video talk device )
VTOPosition	string	O	VTO position number. ( This param is not supported by access control device )
ValidDateStart	string	O	The start of valid date, format is "yyyyMMdd hhmmss".
ValidDateEnd	string	O	The end of valid date, format is "yyyyMMdd hhmmss".
IsValid	bool	O	Is the card still valid. ( This param is not supported by video talk device )
[ Response Params ] (key=value format)			

Name	Type	R/O	Param Description
RecNo	int	R	The record id.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=insert&name=AccessControlCard&CardName=ZhangSan&CardNo=12345&UserID=102&CardStatus=0&CardType=0&Password=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&VTOPosition=01018001&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811		
Response	RecNo=12345		

- Add access user card with fingerprint

仅下发卡片请使用

http://<server>/cgi-bin/recordUpdater.cgi?action=insert&name=AccessControlCard

Table 16-62

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=insertEx&name=AccessControlCard		
Method	POST		
Description	Insert access user card and fingerprint. ( This api is not supported by video talk device )		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be <b>"AccessControlCard"</b> .
CardName	string	O	Access user card name, max string length is 32.
CardNo	string	O	Access user card number.
UserID	string	R	The user's id.
CardStatus	int	O	Card status, 0 means normal, and abnormal status can be sum of following: <ul style="list-style-type: none"> <li>• 1 &lt;&lt; 0: report the loss of the card</li> <li>• 1 &lt;&lt; 1: the card is withdrawn</li> <li>• 1 &lt;&lt; 2: the card is frozen</li> <li>• 1 &lt;&lt; 3: the card is arrearage</li> <li>• 1 &lt;&lt; 4: the card exceeds the time limit</li> <li>• 1 &lt;&lt; 5: the card is arrearage but still can open the door, and there will be voice prompts.</li> </ul>
CardType	int	O	The card type: 0: Normal Card, 1: VIP Card, 2: Visitor Card, 3: Patrol Card, 4: Blacklist Card, 5: Stress Card, 0xff: Mother Card
Password	string	O	The Access card's password.
Doors	array<int>	O	The index of the doors that this card can open.
TimeSections	array<int>	O	The index of the time sections of each door that this card can open.
VTOPosition	string	O	VTO position number. ( This param is not supported by access control device )
ValidDateStart	string	O	The start of valid date, format is <b>"yyyyMMdd hhmmss"</b> .
ValidDateEnd	string	O	The end of valid date, format is <b>"yyyyMMdd hhmmss"</b> .
IsValid	bool	O	Is the card still valid.

FingerprintPacket	object	O	The fingerprint packet info.
+Length	int	O	The length of each fingerprint packet.
+Count	int	O	The count of fingerprint packet.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
RecNo	int	R	The record id.
[ Example ]			
Request	POST http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=insertEx&name=AccessControlCard&CardName=ZhangSan&CardNo=12345&UserID=102&CardStatus=0&Password=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&VTOPosition=01018001&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811&FingerprintPacket.Length=500&FingerprintPacket.Count=3 Content-Type: application/octet-stream Content-Length: <length>  <fingerprint packet binary data>		
Response	RecNo=12345		

### 16.3.1.6 Modify Access User Card and Fingerprint

- Update Access user card only

Table 16-63

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=update&name=AccessControlCard		
Method	GET		
Description	Update access user card info. <b>Note:</b> You should provide at least one optional param to update.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be " <b>AccessControlCard</b> ".
recno	int	R	The record id.
CardName	string	O	Access user card name, max string length is 31.
CardNo	string	O	Access user card number. Primary key, modification is not allowed in principle
UserID	string	O	The user's id. Primary key, modification is not allowed in principle
...<other param>	—	—	...<See above insert command for other params of the record, They are all optional.>
[ Response Params ] ( OK )			
[ Example ]			

Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=update&name=AccessControlCard&recno=12345&CardName=ZhangSan&CardNo=12345&UserID=102&CardStatus=0&CardType=0&Password=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811
Response	OK

- Update access user card and fingerprint

Table 16-64

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=updateEx&name=AccessControlCard		
Method	POST		
Description	Update access user card info and fingerprint. ( This api is not supported by video talk device ) <b>Note:</b> You should provide at least one optional param to update.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be " <b>AccessControlCard</b> ".
recno	int	R	The record id.
CardName	string	O	Access user card name, max string length is 32.
CardNo	string	O	Access user card number. Primary key, modification is not allowed in principle
UserID	string	O	The user's id. Primary key, modification is not allowed in principle
...<other param>			...<See above insert command for other params of the record. They are all optional.>
FingerprintPacket	object	O	The fingerprint packet info.
+Length	int	O	The length of each fingerprint packet.
+Count	int	O	The count of fingerprint packet.
[ Response Params ] ( OK )			
[ Example ]			
Request	POST http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=updateEx&name=AccessControlCard&recno=12345&CardName=ZhangSan&CardNo=12345&UserID=102&CardStatus=0&CardType=0&Password=123456&Doors[0]=1&Doors[1]=3&Doors[2]=5&ValidDateStart=20151022%20093811&ValidDateEnd=20151222%20093811&FingerprintPacket.Length=500&FingerprintPacket.Count=3 Content-Type: application/octet-stream Content-Length: <length>  <fingerprint packet binary data>		
Response	OK		

### 16.3.1.7 Delete Access User Card and Fingerprint

- Delete Access user card and fingerprint record by recno

Table 16-65

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=remove&name=AccessControlCard		
Method	GET		
Description	Remove the access user card and fingerprint record. ( The video talk device can not delete fingerprint data.)		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be "AccessControlCard".
recno	int	R	The record id.
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=remove&name=AccessControlCard&recno=12345		
Response	OK		

- Delete all Access user card and fingerprint records

Table 16-66

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=clear&name=AccessControlCard		
Method	GET		
Description	Remove all the access user card and fingerprint records. ( The video talk device can not delete fingerprint data.)		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Access user card and fingerprint record name, should be "AccessControlCard".
[ Response Params ] ( OK )			
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=clear&name=AccessControlCard		
Response	OK		

### 16.3.1.8 Find Access User Card and Fingerprint

- Find Access user card by condition

Table 16-67

URL	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=AccessControlCard		
Method	GET		
Description	Find Access user card record by condition.		

[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Record Name, should be " <b>AccessControlCard</b> ".
count	int	O	Max result to return, default is 1024.
StartTime	string	O	The start of the record's CreateTime.
EndTime	string	O	The End of the record's CreateTime.
condition	object	O	Search condition.
+CardNo	string	O	Access user card number.
+UserID	string	O	The user's id.
+IsValid	bool	O	The access card valid or not.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
totalCount	int	O	The total record num that find.
found	int	O	The record num that returned.
records	array<object>	R	The records that returned.
+RecNo	int	R	The record id.
+CreateTime	int	O	The create time of record.
+CardName	string	R	The access user card name, max string length is 32.
+CardNo	string	R	The access user card number.
+UserID	string	R	The user's id.
+CardStatus	int	R	Card status, 0 means normal, and abnormal status can be sum of following: <ul style="list-style-type: none"> <li>1 &lt;&lt; 0: report the loss of the card</li> <li>1 &lt;&lt; 1: the card is withdrawn</li> <li>1 &lt;&lt; 2: the card is frozen</li> <li>1 &lt;&lt; 3: the card is arrearage</li> <li>1 &lt;&lt; 4: the card exceeds the time limit</li> <li>1 &lt;&lt; 5: the card is arrearage but still can open the door, and there will be voice prompts.</li> </ul>
+CardType	int	O	The card type: 0: Normal Card, 1: VIP Card, 2: Visitor Card, 3: Patrol Card, 4: Blacklist Card, 5: Stress Card, 0xff: Mother Card
+Password	string	O	The Access card's password. ( This param is not supported by video talk device )
+Doors	array<int>	O	The index of the doors that this card can open. ( This param is not supported by video talk device )
+TimeSections	array<int>	O	The index of the time sections of each door that this card can open. ( This param is not supported by video talk device )
+VTOPosition	string	O	VTO position number. ( This param is not supported by access control device )
+ValidDateStart	string	O	The start of valid date, format is " <b>yyyyMMdd hhmss</b> ".
+ValidDateEnd	string	O	The end of valid date, format is " <b>yyyyMMdd hhmss</b> ".
+IsValid	bool	O	Is the card still valid. ( This param is not supported by video talk device )
+Handicap	bool	O	is the cad of extend time

+CitizenIDNo	string	O	citizen card id
+RepeatEnterRouteTimeout	integer	O	timeout of repeater enter route, unit: s
+UserType	integer	O	user type, it can be { 1: normal user 2: user in blacklist }
+FirstEnter	bool	O	is with first enter authority
+DynamicCheckCode	string	O	dynamic check code
+UseTime	integer	O	count of use

[ Example ]

Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=AccessControlCard&condition.UserID=103&StartTime=123456700&EndTime=123456800&count=100
Response	<pre> totalCount=1000 found=100 records[0].RecNo=12345 records[0].CreateTime=123456789 records[0].CardName=ZhangSan records[0].CardNo=300 records[0].UserID=103 records[0].CardStatus=0 records[0].CardType=0 records[0].Doors[0]=1 records[0].Doors[1]=3 records[0].Doors[2]=5 records[0].VTOPosition=01018001 records[0].ValidStart=20151022 093811 records[0].ValidEnd=20151222 093811 records[0].IsValid=true ... records[1].RecNo=13579 records[1].CreateTime=123456799 records[1].StaticsticsTime=123456799 records[1].CardName=ZhangSan records[1].CardNo=302 records[1].UserID=103 records[1].CardStatus=0 records[1].CardType=0 records[1].Doors[0]=2 records[1].Doors[1]=4 records[1].Doors[2]=6 records[1].VTOPosition=01018002 records[1].ValidStart=20151022 093811 records[1].ValidEnd=20151222 093811 records[1].IsValid=true ... </pre>

- Find Access user card by recno

Table 16-68

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=get&name=AccessControlCard		
Method	GET		
Description	Find Access user card record by recno.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	Record Name, should be " <b>AccessControlCard</b> ".
recno	int	R	The record id.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
record	object	R	The record that returned.
+RecNo	int	R	The record id.
+CreateTime	int	O	The create time of record.
+CardName	string	R	The access user card name, max string length is 32.
+CardNo	string	R	The access user card number.
+UserID	string	R	The user's id.
...<other param>	—		...<See above find command for other params of the record.>
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=get&name=AccessControlCard&recno=4		
Response	<pre> record.RecNo=4 record.CreateTime=123456789 record.CardName=ZhangSan record.CardNo=300 record.UserID=103 record.CardStatus=0 record.CardType=0 record.Doors[0]=1 record.Doors[1]=3 record.Doors[2]=5 record.VTOPosition=01018001 record.ValidStart=20151022 093811 record.ValidEnd=20151222 093811 record.IsValid=true ... </pre>		

- Find Access user card and fingerprint by recno

Table 16-69

URL	http://<server>/cgi-bin/recordUpdater.cgi?action=getEx&name=AccessControlCard		
Method	GET		
Description	Find Access user card and fingerprint record by recno. ( This api is not supported by video talk device )		
[ Request Params ] ( key=value format in URL )			

Name	Type	R/O	Param Description
name	string	R	Record Name, should be " <b>AccessControlCard</b> ".
recno	int	R	The record id.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
record	object	R	The record that returned.
+RecNo	int	R	The record id.
+CreateTime	int	O	The create time of record.
+CardName	string	R	The access user card name, max string length is 32.
+CardNo	string	R	The access user card number.
+UserID	string	R	The user's id.
...<other param>	—	—	...<See above find command for other params of the record.>
+FingerprintPacket	object	O	The fingerprint packet info.
++Length	int	O	The length of each fingerprint packet.
++Count	int	O	The count of fingerprint packet.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordUpdater.cgi?action=getEx&name=AccessControlCard&recno=4		

Response	<pre> HTTP/1.1 200 OK Server: Device/1.0 Content-Type: multipart/x-mixed-replace; boundary=&lt;boundary&gt; Content-Length: &lt;length&gt;  --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  record.RecNo=4 record.CreateTime=123456789 record.CardName=ZhangSan record.CardNo=300 record.UserID=103 record.CardStatus=0 record.CardType=0 record.Doors[0]=1 record.Doors[1]=3 record.Doors[2]=5 record.VTOPosition=01018001 record.ValidStart=20151022 093811 record.ValidEnd=20151222 093811 record.IsValid=true ... record.FingerprintPacket.Length=500 record.FingerprintPacket.Count=3 --&lt;boundary&gt; Content-Type: application/octet-stream Content-Length: &lt;length&gt;  &lt;fingerprint packet binary data&gt; --&lt;boundary&gt;-- </pre>
----------	---

### 16.3.1.9 Get the Total Number of Records of Access User Card and Fingerprint

Table 16-70

URL	http://<server>/cgi-bin/recordFinder.cgi?action=getQuerySize&name=AccessControlCard		
Method	GET		
Description	Get the total number of records of the access user card record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description

name	string	R	Access user card and fingerprint record name, should be " <b>AccessControlCard</b> ".
[ Response Params ] ( key=value )			
count	int	R	The total number of records.
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=getQuerySize&name=AccessControlCard		
Response	count=150		

## 16.3.2 Access control

### 16.3.2.1 [Event] AccessControl

Table 16-71

Usage	Refer to "4.4.3 Subscribe to Snapshot" for how to subscribe event		
Description	When user trying to open the door, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array
+EventBaseInfo	object	R	Base info of event
++Code	String	R	Event Code. It should be <b>AccessControl</b>
++Action	String	R	Event Action. It can be: "Start", "Stop", "Pulse"
++Index	int	O	The channel index relate to this event.
+RecNo	int	R	The record id
+Name	string	O	The name of the door
+Type	string	O	The event type. It can be: " <b>Entry</b> ", " <b>Exit</b> "
+Status	int	O	Open door result, can be: 0—failed, 1—success If this param does not exist, that means success.

+Method	int	R	<p>Open door method, can be:</p> <ul style="list-style-type: none"> <li>0: by password</li> <li>1: by access card</li> <li>2: by access card and then password</li> <li>3: by password and then access card</li> <li>6: by fingerprint</li> <li>7: by password and access card and fingerprint together</li> <li>8 : by password and fingerprint together</li> <li>9 : by access card and fingerprint together</li> <li>10: reserved</li> <li>11: by multiple access user</li> <li>12: by key</li> <li>13: by duress password</li> <li>14: by QR code, local</li> <li>15: by face recognition, local</li> <li>16: reserved</li> <li>17: by ID card</li> <li>18: by face and ID card</li> <li>19: by Bluetooth</li> <li>20: by custom password</li> <li>21: by UserId and password</li> <li>22: by face and password</li> <li>23: by fingerprint and password</li> <li>24: by fingerprint and face</li> <li>25: by access card and face</li> <li>26: by face or password</li> <li>27: by fingerprint or password</li> <li>28: by fingerprint or face</li> <li>29: by access card or face</li> <li>30: by access card or fingerprint</li> <li>31: by fingerprint and face and password</li> <li>32: by access card and face and password</li> <li>33: by access card and fingerprint and password</li> <li>34: by access card and fingerprint and face</li> <li>35: by fingerprint or face or password</li> <li>36: by access card or face or password</li> <li>37: by access card or fingerprint or face</li> <li>38: by access card and fingerprint and face and password</li> <li>39: by access card or fingerprint or face or password</li> <li>40: by ID card and face, or access card or face</li> <li>41: by ID card or QR code or face</li> <li>42: by DTMF(SIPINFO, RFC2833, INBAND)</li> <li>43: by QR code, remote</li> <li>44: by face recognition, remote</li> <li>45: by ID card ( match with fingerprint in ID card )</li> <li>46: by temporary password</li> <li>47: by health code</li> </ul>
---------	-----	---	--

+CardNo	string	O	Card number if the door is opened by card
+UserID	string	R	The user id
+ErrorCode	integer	O	error code
+ObjectProperties	object	O	Dynamic structure info
+Mask	integer	O	is with mask, it can be { 0: unknow 1: without mask 2: with mask }
+ManTemperatureInfo	object	O	people's temperature info
++CurrentTemperature	float	O	people's current temperature
++TemperatureUnit	integer	O	temperature unit:{ 0: centigrade 1: Fahrenheit 2: Kelvin }
++IsOverTemperature	bool	O	is over temperature

[ Example ]

Event	<pre> --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code=TrafficJam Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].RecNo=123 Events[0].Name=Door1 Events[0].Type=Entry Events[0].Status=1 Events[0].Method=1 Events[0].CardNo=09DDAABB Events[0].UserID=101 ... --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;Jpeg image data&gt; --&lt;boundary&gt; </pre>
-------	---

### 16.3.2.2 [Event] CitizenPictureCompare

Table 16-72

Usage	Refer to "4.4.3 Subscribe to Snapshot"for how to subscribe event		
Description	When user trying to open the door, send this event.		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event info array
+EventBaseInfo	object	R	Base info of event
++Code	String	R	Event Code. It should be <b>CitizenPictureCompare</b>
++Action	String	R	Event Action. It can be: "Pulse"
++Index	int	O	The channel index relate to this event.
+UTC	uint32	O	UTC 时间
+CompareResult	bool	O	compare result. If the similarity is greater than or equal to the threshold, the comparison is considered successful
+Similarity	integer	O	Similarity of two pictures, Unit: percentage range[1, 100]
+Threshold	integer	O	check threshold range[1, 100]
+Citizen	String	O	citizen name
+Sex	integer	O	sex, it can be: { 0 unknow 1 male 2 female 9 not stated }
+Ethnicity	integer	O	ethnicity (refer to CitizenIDCard)
+Birth	String	O	birthday, such as "1980-01-01"
+Address	String	O	address
+Number	String	O	citizen ID
+Authority	String	O	signing and issuing organization
+Start	String	O	Effective date
+End	String	O	Expiration date, and "Endless" mens Long term effectiveness
+ImageInfo	array<object>	O	Picture information, the first for the face cutout, the second for the ID card photo
++Offset	integer	O	the offset in binary data
++Length	integer	O	length of picture, unit: Byte
++Width	integer	O	Picture width, pixels
++Height	integer	O	Picture height, pixels
+ImageInfoEx	array<object>	O	extension of picture info, (The total number of images uploaded is determined by imageinfo and imageinfoex) 6 at most
++Type	integer	O	picture type 0 Local face database

			1 Shooting scene map
++Offset	integer	O	the offset in binary data
++Length	integer	O	length of picture, unit: Byte
++Width	integer	O	Picture width, pixels
++Height	integer	O	Picture height, pixels
+CardNo	String	O	IC card number (for building products)
+CellPhone	String	O	phone number (input number before comparison) (for building products)
+BuildingNumber	String	O	building number (for building products)
+BuildingUnitNumber	String	O	building entrance number (for building products)
+BuildingRoomNumber	String	O	room number (for building products)
+PersonnelRelationship	String	O	relationship (for building products)
+Method	integer	O	method (refer to OpenDoorMethod)
+EventGroupID	integer	O	event group id, used to associate different events are with same action. (Used with AccessControlevents)
+CallNumber	String	O	phone number to be called.(called after compare succeed) (for building products)
+EventType	integer	O	event type: 0: compare result 1: Face acquisition 2: visitor registration 3: Face permission distribution 4: inquirement of ID card face group
+UserID	String	O	user ID
+FaceIndex	integer	O	face index, range [0,4]
+Mask	integer	O	is with mask, it can be { 0: unknow 1: without mask 2: with mask } default 0.

[ Example ]

Event	<pre>--&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code= CitizenPictureCompare Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].UTC=1999999999 Events[0].CompareResult=true Events[0].Similarity=80</pre>
-------	---

	<pre> Events[0].Threshold=75 ... --&lt;boundary&gt; Content-Type: image/jpeg Content-Length: &lt;image size&gt;  &lt;Jpeg image data&gt; --&lt;boundary&gt; </pre>
--	--

### 16. 3. 2. 3 [Event] Door Status Event

Usage	For the method of subscribing to events, refer to "错误!未找到引用源。 错误!未找到引用源。".		
Description	Door status event		
[ Event Params ] ( key=value format )			
Name	Type	R/O	Param Description
Events	array<object>	R	Event array
+EventBaseInfo	object	R	Basic information of event
++Code	String	R	Event code ("DoorStatus")
++Action	String	R	Event action ("Pulse")
++Index	int	O	Door channel
+UTC	uint32	-	Standard UTC time (without DST deviation of time zone). Required for access control products, and optional for intercom products.
+Status	enumchar[32]	-	Door status Enumchar[32]{ "Open" "Close" "CloseAlways" "OpenAlways" "Normal" }
<b>Example</b>			
Event	<pre> --&lt;boundary&gt; Content-Type: text/plain Content-Length: &lt;length&gt;  Events[0].EventBaseInfo.Code= DoorStatus Events[0].EventBaseInfo.Action=Pulse Events[0].EventBaseInfo.Index=0 Events[0].UTC=1999999999 Events[0].Status= CloseAlways --&lt;boundary&gt; </pre>		

### 16.3.2.4 Find AccessControl Record

Table 16-73

URL	http://<server>/cgi-bin/recordFinder.cgi?action=find&name=<RecordName>		
Method	GET		
Description	Find AccessControl record.		
[ Request Params ] ( key=value format in URL )			
Name	Type	R/O	Param Description
name	string	R	The record table name. It should be <b>"AccessControlCardRec"</b> .
count	int	O	Max result to return, default is 1024.
StartTime	string	O	The start of the record's CreateTime.
EndTime	string	O	The end of the record's CreateTime.
condition	object	O	Search condition.
+CardNo	string	O	Access user card number.
[ Response Params ] (key=value format)			
Name	Type	R/O	Param Description
totalCount	int	O	Total record num that find.
found	int	O	Record num that returned.
records	array<object >	R	The records that returned.
+RecNo	int	R	Record id.
+CreateTime	int	O	Record create time, UTC time.
+CardNo	int	R	Access user card number.
+CardName	string	O	Access user card name, max string length is 31.
+CardType	int	O	The card type: 0: Normal Card, 1: VIP Card, 2: Visitor Card, 3: Patrol Card, 4: Blacklist Card, 5: Stress Card, 0xff: Mother Card
+Password	string	O	The Access card's password.
+UserID	string	R	The user's id.
+Type	string	O	The event type. It can be: <b>"Entry"</b> , <b>"Exit"</b> .
+Status	int	O	Open door result. It can be: 0—failed, 1—success. If this param does not exist, that means success.
+Method	int	R	Open door method. It can be: <ul style="list-style-type: none"> <li>0: by password</li> <li>1: by access card</li> <li>2: by access card and then password</li> <li>3: by password and then access card</li> <li>6: by fingerprint</li> <li>15: by face recognition</li> </ul>
+Door	int	O	The index of the door. ( This param is not supported by video talk device )
+ReaderID	string	O	The access user card ID of reader. ( This param is not supported by video talk device )
+ErrorCode	int	O	The error code, valid only when Status is 0.

+URL	string	O	The picture's URL, max string length is 127. ( This param is not supported by access control device )
+RecordURL	string	O	The record video's URL, max string length is 127. ( This param is not supported by access control device )
+IsOverTemperature	bool	O	is over temperature
+TemperatureUnit	integer	O	temperature unit:{ 0: centigrade 1: Fahrenheit 2: Kelvin }
+CurrentTemperature	float	O	current temperature
+CitizenIDResult	bool	O	citizen ID compare result
+CitizenIDName	string	O	citizen name
+CitizenIDNo	string	O	citizen ID number
+CitizenIDSex	integer	O	sex:{ 0: unknow 1: male 2: female 9: not stated }
+CitizenIDEthnicity	integer	O	citizen ID ethnicity
+CitizenIDBirth	string	O	birthday, format as "1980-01-01"
+CitizenIDAddress	string	O	address
+CitizenIDAuthority	string	O	signing and issuing organization
+CitizenIDStart	string	O	Effective date
+CitizenIDEnd	string	O	Expiration date, and "Endless" mens Long term effectiveness
[ Example ]			
Request	GET http://192.168.1.108/cgi-bin/recordFinder.cgi?action=find&name=AccessControlCardRec&StartTime=123456700&EndTime=123456800&condition.CardNo=12001&count=100		
Response	totalCount=1000 found=100 records[0].RecNo=12345 records[0].CreateTime=123456789 records[0].CardNo=12001 records[0].CardName=ZhangSan records[0].UserID=ZhangSan records[0].Type=Entry records[0].Method=1 ... records[1].RecNo=13579		

```
records[1].CreateTime=123456799
records[1].CardNo=12001
records[1].CardName=ZhangSan
records[1].UserID=ZhangSan
records[1].Type=Exit
records[1].Method=1
...
```

DAHUA HTTP API V3.05 for Whisper House Kft.