HTTP API 说明书
HTTP API Specifications

VER. 1.5

修订记录

修订日期	修订版本	描述	修订人
2023-11-03	**V1.0**	首次发布	李光明
2023-12-05	**V1.1**	增加**IO**状态控制	李光明
2024-01-03	**V1.2**	增加对补光灯亮度模式的配置支持	李光明
2024-01-15	**V1.3**	**1)**事件通知增加**Picture**可选项可携带图片 **2)**事件通知增加**OccurFlag**属性，**true**表示报警发生，**false**表示报警结束	李光明
2024-01-16	**V1.4**	**1)**事件通知增加本地时间字符串 **2)**补充设置目标检测**HAPI**方法章节	李光明
2024-03-18	**V1.5**	增加视频遮挡、人脸检测、越界检测、区域（周界）侦测、烟火检测配置**GET/SET**	李光明

目录

HTTP API 说明书 1
HTTP API Specifications 1
VER. 1.5 1
1 引言 7
1.1 HAPI 概述 7
1.2 HAPI 采用的 HTTP 方法 7
1.3 HAPI 设备回应消息 7
1.4 HAPI URL 规则 8
1.5 HAPI 认证 8
1.5.1 用户名/密码认证方式 9
1.5.2 Session ID 认证方式 9
1.6 HAPI 接口示例代码 9
1.6.1 GET 方法明文密码调用 getuid 9
1.6.2 PUT 方法调用 getuid 9
1.6.3 GET 方法密文密码调用 getuid 10
1.6.4 GET 方法调用 keep_alive 10
1.6.5 PUT 方法调用 keep_alive 10
2 接口介绍 11
2.1 Session ID 11
2.1.1 获取 uid /HAPI/V1.0/uid/getuid 11
2.1.2 保活 uid /HAPI/V1.0/uid/keep_alive 11
2.2 系统信息 12
2.2.1 获取设备信息 /HAPI/V1.0/sysinfo/device_info 12
2.2.2 获取 API 接口列表 /HAPI/V1.0/sysinfo/functionlist 13
2.2.3 获取设备能力集 /HAPI/V1.0/sysinfo/capability 16
2.2.4 获取 RTSP URL /HAPI/V1.0/sysinfo/rtspurl 18
2.3 系统控制 19
2.3.1 重启设备 /HAPI/V1.0/sysman/reboot 19
2.3.2 恢复出厂 /HAPI/V1.0/sysman/factory 19
2.4 IO 状态控制 20
2.4.1 获取 IO 输入状态 /HAPI/V1.0/io/input/get 20
2.4.2 获取 IO 输出状态 /HAPI/V1.0/io/output/get 20
2.4.3 设置 IO 输出状态 /HAPI/V1.0/io/output/set 21

2.5 手动抓图 22
2.5.1 手动抓图 /HAPI/V1.0/snapshot.cgi......22
2.6 云台控制 23
2.6.1 云台停止 /HAPI/V1.0/ptz_ctrl/stop......23
2.6.2 云台转动 /HAPI/V1.0/ptz_ctrl/move......23
2.6.3 云台预置点操作 /HAPI/V1.0/ptz_ctrl/preset......24
2.6.4 云台变倍操作 /HAPI/V1.0/ptz_ctrl/zoom......25
2.6.5 云台聚焦操作 /HAPI/V1.0/ptz_ctrl/focus......25
2.6.6 云台光圈操作 /HAPI/V1.0/ptz_ctrl/iris......26
2.6.7 云台高级功能执行操作 /HAPI/V1.0/ptz_ctrl/advfunction/exec......27
2.6.8 云台高级功能列表 /HAPI/V1.0/ptz_ctrl/advfunction/get......27
2.7 系统时间 28
2.7.1 获取设备时间和配置 /HAPI/V1.0/systime/gettime......28
2.7.2 设置设备时间和时区 /HAPI/V1.0/systime/settime......29
2.7.3 设置设备 NTP 配置 /HAPI/V1.0/systime/setntp......29
2.8 灯光配置 30
2.8.1 获取支持的灯光开关控制模式能力集/HAPI/V1.0/ system/light/ctrlmode/capability......30
2.8.2 获取支持的灯光工作模式能力集/HAPI/V1.0/ system/light/workmode/capability......31
2.8.3 获取灯光配置/HAPI/V1.0/system/light/get......32
2.8.4 设置灯光配置/HAPI/V1.0/system/light/set......33
2.9 图像配置 34
2.9.1 获取图像配置/HAPI/V1.0/system/image/get......34
2.9.2 设置图像配置/HAPI/V1.0/system/image/set......35
2.10 视频配置 35
2.10.1 获取视频编码能力集/HAPI/V1.0/system/video/capability......35
2.10.2 获取视频编码配置/HAPI/V1.0/system/video/get......37
2.10.3 设置视频编码配置/HAPI/V1.0/system/video/set......39
2.11 音频配置 40
2.11.1 获取音频配置编码能力集/HAPI/V1.0/system/audio/capability......40
2.11.2 获取音频配置/HAPI/V1.0/system/audio/get......41
2.11.3 设置音频配置/HAPI/V1.0/system/audio/set......42
2.12 OSD 配置......43
2.12.1 获取 OSD 配置/HAPI/V1.0/system/osd/get......43

2.12.2 设置 OSD 配置/HAPI/V1.0/system/osd/set 45
2.13 智能检测配置....47
2.13.1 获取智能检测能力集/HAPI/V1.0/smart/capability 47
2.13.2 获取目标检测（人车非机动车）算法支持的目标类型/HAPI/V1.0/smart/objectdetect/capability 48
2.13.3 获取目标检测（人车非机动车）算法支持联动能力/HAPI/V1.0/smart/linkage/capability 49
2.13.4 获取警戒音列表/HAPI/V1.0/smart/audiofiles/get 50
2.13.5 获取运动侦测配置/HAPI/V1.0/motiondetect/get....51
2.13.6 设置运动侦测配置/HAPI/V1.0/motiondetect/set 55
2.13.7 获取目标检测配置/HAPI/V1.0/smart/objectdetect/get 57
2.13.8 设置目标检测配置/HAPI/V1.0/smart/objectdetect/set....61
2.13.9 获取视频遮挡配置/HAPI/V1.0/smart/videocover/get 63
2.13.10 设置视频遮挡配置/HAPI/V1.0/smart/videocover/set 64
2.13.11 获取人脸检测配置/HAPI/V1.0/smart/facedetect/get....65
2.13.12 设置人脸检测配置/HAPI/V1.0/smart/facedetect/set 67
2.13.13 获取越界检测配置/HAPI/V1.0/smart/videogate/get 69
2.13.14 设置越界检测配置/HAPI/V1.0/smart/videogate/set 74
2.13.15 获取区域(周界)侦测配置/HAPI/V1.0/smart/regionai/get 75
2.13.16 设置区域(周界)侦测配置/HAPI/V1.0/smart/regionai/set....79
2.13.17 获取车牌识别配置/HAPI/V1.0/smart/lpr/get 80
2.13.18 设置车牌识别配置/HAPI/V1.0/smart/lpr/set....82
2.13.19 获取烟火检测配置/HAPI/V1.0/smart/flameflumes/get....84
2.13.20 设置烟火检测配置/HAPI/V1.0/smart/flameflumes/set 87
3 事件 89
3.1 事件工作流程....89
3.2 事件订阅 90
3.2.1 注册订阅/HAPI/V1.0/Event/subscription/regist....90
3.2.2 刷新订阅/HAPI/V1.0/Event/subscription/refresh 91
3.2.3 删除订阅/HAPI/V1.0/Event/subscription/delete....93
3.3 事件通知 94
3.3.1 事件通知/HAPI/V1.0/Event/Notification 94
4 JSON 结构化数据说明....97

4.1 布防方式（ArmingMode） 97
4.2 报警检测布防配置....97
4.3 运动侦测区域配置....98
4.4 报警联动布防配置....99
4.5 警戒音联动配置....100
4.6 IO 输出联动配置 100
4.7 目标检测类型....101
4.8 目标检测区域配置....101
4.9 矩形检测区域配置....102
4.10 越界检测规则配置....102
4.11 区域侦测规则配置....104
4.12 事件通知 Json Block....105
4.12.1 MotionDetectInfo 105
4.12.1 ObjectDetectInfo 107

1 引言

1.1 HAPI 概述

HTTP API 简称为 HAPI，采用 HTTP 短连接机制，一次交互过程中，客户端通过 HTTP URL 通知服务端需要请求的资源，服务端收到消息后，将 URL 指定的资源信息和请求结果等按照 Json 格式返回。

HAPI 支持：

1） 获取设备能力，以及支持的 HAPI 方法集合
2） 系统控制，例如重启、恢复出厂等
3） 云台控制，例如转动、预置点设置、预置点调用等
4） 配置的获取和设置，例如网络、视频、音频、存储、报警、智能分析、云台等配置
5） 事件订阅机制
6） 事件报警上报

1.2 HAPI 采用的 HTTP 方法

HAPI 接口使用的 http 方法包括 GET/PUT/POST 等。
GET 请求的消息体为空，需要的参数在 URL 中携带。URL 以？开头携带参数，参数之间用&分割。
PUT/POST 请求参数在消息体中以 json 格式携带。
POST 请求需要指定 Content-Type：application/x-www-form-urlencoded。

为了方便浏览器直接调试 HAPI 接口，大部分获取信息、系统控制、系统控制等 HAPI 接口支持 GET 方法，在 URL 中填写必要参数。支持 GET 的接口同时支持 PUT 方法，在消息体中用 json 格式携带多项参数。
事件订阅、刷新、取消订阅、通知都采用 POST 方法

1.3 HAPI 设备回应消息

当方法是 GET 时，HTTP 请求在 URL 中携带需要的参数，消息体为空。
当方法是 PUT
POST 时，HTTP 消息体通常需要包含 JSON 数据。请求报文必须发往设备的 HTTP/HTTPS 端口。
设备以 JSON 数据格式进行回应，通常包含如下字段，各字段含义如下：
ResponseURL：表示 HAPI 请求报文 URL 请求的方法
SessionID：getuid 返回的接口 Session ID。如果是用户名/密码认证方式，SessionID 为空值。
ResponseCode：表示系统处理结果，0 表示成功
ResponseString：Succeed 表示成功，其他文本为为错误提示
Data：表示请求报文（GET）的响应结果，多数情况下是 JSON 格式。若 URL 相同，PUT 请求报文

数据格式与 GET 响应报文中的数据格式相同。下表中的 Data 项列出了每个 API 的数据。如果服务器处理失败，Data 为“null”。如果 HTTP 报文方法是 PUT/POST，响应报文中的 Data 为“null”。

```
{
  "Response": {
    "ResponseURL": "/HAPI/V1.0/uid/getuid",
    " SessionID": "15E25D",
    "ResponseCode": 0,
    "ResponseString": "Succeed",
    "Data": "null",
  }
}
```

1.4 HAPI URL 规则

/HAPI/V1.0[/Channels/<ID>]/<service-name>/<resource-name>[/<child-resource-name>][/<ID>]][?<key1>=<value1>[&<key2>=<value2>]]

参数说明：

组成	说明
[ ]:	表示非必填参数
<>	表示必填参数
[/Channels/<ID>]	通道号，多通道设备 **ID** 为 **0** 表示通道号无关，**ID** 为从 **1** 开始表示通道号（适用于 **NVR** 和多通道摄像机）。单目摄像机该参数可省略。
<service-name>	业务类型，比如系统、媒体、网络、存储、云台控制等
<resource-name>	资源名称
[/<child-resource-name>]	子资源名称，非必填
[?<key1>=<value1>[&<key2>=<value2>]]	以？开头，携带需要的参数。多个参数之间用&分割

1.5 HAPI 认证

设备收到 **HAPI** 请求时，首先会对请求的 **API URL** 进行合法性认证。合法性认证不通过的请求，回应失败信息给请求端。

HTTP 请求携带 **Session ID** 或者用户名/密码作为认证信息，**GET** 请求在 **URL** 中携带，**PUT/POST** 请求在消息体中携带。

1.5.1 用户名/密码认证方式

直接携带用户名和密码信息，密码支持明文和 32 位 MD5 加密字符串：
[username=<username>&password=<password>]
用户名/密码字符串的大小写不影响认证结果。

1.5.2 Session ID 认证方式

首先通过**/HAPI/V1.0/uid/getuid** 接口获取 **Session ID**，携带上节所述的用户名和密码信息。后续调用 **HAPI** 接口都需要在携带该 **Session ID**。
Session ID 在最后一次成功请求后的 **60** 秒超时，失效之后需要重新请求 **Session ID**。如果长时间没有 **HAPI** 请求但又希望 **Session ID** 不失效，可以使用**/HAPI/V1.0/uid/ keep_alive** 保持心跳。
[uid=<uid>]

1.6 HAPI 接口示例代码

1.6.1 GET 方法明文密码调用 getuid

```
GET /HAPI/V1.0/uid/getuid?username=admin&password=123456
```

返回结果:

```
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/uid/getuid",
        "SessionID": "371C1A3",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": "null"
    }
}
```

1.6.2 PUT 方法调用 getuid

```
PUT /HAPI/V1.0/uid/getuid HTTP/1.1

User-Agent: PostmanRuntime-ApipostRuntime/1.1.0

Cache-Control: no-cache
```

```
content-type: application/json

Accept: */*

Accept-Encoding: gzip, deflate, br

Connection: keep-alive

Host: 192.168.1.202

Content-Length: 82


{

    "username":"admin",

    "password":"e10adc3949ba59abbe56e057f20f883e"

}
```

返回结果：

```
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/uid/getuid",
        "SessionID": "3729116",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": "null"
    }
}
```

1.6.3 GET 方法密文密码调用 getuid

```
GET /HAPI/V1.0/uid/getuid?username=admin&password=e10adc3949ba59abbe56e057f20f883e
```

1.6.4 GET 方法调用 keep_alive

```
GET /HAPI/V1.0/uid/keep_alive?uid=15E25D
```

1.6.5 PUT 方法调用 keep_alive

```
PUT /HAPI/V1.0/uid/keep_alive HTTP/1.1

User-Agent: PostmanRuntime-ApipostRuntime/1.1.0

Cache-Control: no-cache

content-type: application/json

Accept: */*

Accept-Encoding: gzip, deflate, br

Connection: keep-alive
```

```
Host: 192.168.1.202

Content-Length: 25


{

    "uid":"3CFABD6"

}
```

2 接口介绍

2.1 Session ID

2.1.1 获取 uid /HAPI/V1.0/uid/getuid

URL	**/HAPI/V1.0/uid/getuid**
Description	**获取 uid，用于后续 API 会话**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/uid/getuid?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password**

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/uid/getuid",
	        "SessionID": "3CC2457",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": "null"
	    }
	}

Note	
Status	

2.1.2 保活 uid /HAPI/V1.0/uid/keep_alive

URL	**/HAPI/V1.0/uid/keep_alive**
Description	**定期（60 秒以内）发送心跳，使改 uid 在设备上一直有效**

Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/uid/keep_alive?uid=3CFABD6
URL params(简单参数) **Input Data(PUT)**	**uid**

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/uid/keep_alive",
	        "SessionID": "3CC2457",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": "null"
	    }
	}

Note	
Status	

2.2 系统信息

2.2.1 获取设备信息 /HAPI/V1.0/sysinfo/device_info

URL	**/HAPI/V1.0/sysinfo/device_info**
Description	获取设备信息
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/sysinfo/device_info?username=admin&password=123456
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/sysinfo/device_info",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": {
	            "SN": "EF000000000000A3",
	            "device_type": "YM800S",
	            "model": "MC800S",
	            "ether": "F0:00:00:00:00:A3",
	            "kernelversion": "Linux 4.9.84 #73 SMP PREEMPT Mon Feb 28 12:36:32 CST 2022 armv7l",
	            "fsversion": "YM800S_AF_V0_Y_EN-RTMP-H5_L V3.2.5.3 build 2023-10-31 15:04:32 "

	} } }
Note	
Status	

2.2.2 获取 API 接口列表 /HAPI/V1.0/sysinfo/functionlist

URL	**/HAPI/V1.0/sysinfo/functionlist**
Description	**获取设备支持的 API 接口列表**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/sysinfo/functionlist?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/sysinfo/functionlist",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": [
	{
	"api": "/HAPI/V1.0[/Channels/ID]/uid/getuid"
	},
	{
	"api": "/HAPI/V1.0[/Channels/ID]/uid/keep_alive"
	},
	{
	"api": "/HAPI/V1.0[/Channels/ID]/sysinfo/device_info"
	},
	{
	"api": "/HAPI/V1.0[/Channels/ID]/sysinfo/functionlist"
	},
	{
	"api": "/HAPI/V1.0[/Channels/ID]/sysinfo/capability"
	},
	{
	"api": "/HAPI/V1.0[/Channels/ID]/sysinfo/rtspurl"
	},
	{

```
            "api": "/HAPI/V1.0[/Channels/ID]/sysman/reboot"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/sysman/factory"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/io/input/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/io/output/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/io/output/set"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/systime/gettime"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/systime/settime"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/systime/setntp"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/stop"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/move"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/preset"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/zoom"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/focus"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/iris"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/advfunction/exec"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/ptz_ctrl/advfunction/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/light/ctrlmode/capability"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/light/workmode/capability"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/light/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/light/set"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/image/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/image/set"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/video/capability"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/video/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/video/set"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/audio/capability"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/audio/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/audio/set"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/osd/get"
        },
        {
            "api": "/HAPI/V1.0[/Channels/ID]/system/osd/set"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/capability"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/audiofiles/get"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/objectdetect/capability"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/linkage/capability"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/motiondetect/get"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/motiondetect/set"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/objectdetect/get"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Smart/objectdetect/set"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Event/subscription/regist"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Event/subscription/refresh"
            },
            {
                "api": "/HAPI/V1.0[/Channels/ID]/Event/subscription/delete"
            }
        ]
    }
}
```

Note	**API 接口列表通过 json 数组返回**
Status	

2.2.3 获取设备能力集 /HAPI/V1.0/sysinfo/capability

URL	**/HAPI/V1.0/sysinfo/capability**
Description	**获取设备能力集**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/sysinfo/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/sysinfo/capability",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": [
	{
	"caps": "gpio_input"
	},
	…………
	{
	"caps": "dzoomsetting"
	}
	]
	}
	}

Note	
Status	

灯光、智能、编码等需要关注的能力集字符串列表

ptz_control	具备云台控制能力
audio_action	支持报警联动声音输出
ledtype_set	支持设置补光灯模式：智能双光/纯红外/纯白光
ir_led_panel	纯红外灯板，补光灯模式不支持切换到白光/双光模式
white_led_panel	纯白光灯板，补光灯模式不支持切换到白光/双光模式
double_led_panel	智能双光灯板
ircut_setting	支持设置 IRCUT 模式（软光敏自动、外部、定时）
ircut_ledmanualswitch	IRCUT 模式增加 2 种：手动灯光常开、手动灯光常关(具体什么灯由补光灯模式决定)
ircut_leddelay	支持设置软光敏的开关灯灵敏度

AlarmVG	拌线检测
AlarmRegionAI	区域侦测
VideoPD	人车检测
face_detect	人脸检测
face_recognize	人脸识别
VehicleCar	车形检测-汽车
VehicleMoto	车形检测-摩托
VehicleElectricbicycle	车型检测-电单车
VehicleBicycle	车型检测-自行车
LPRecognition	车牌识别
Fire	火焰检测
three_video	支持视频第三码流
twolens_stitch	双目拼接摄像机

2.2.4 获取 RTSP URL /HAPI/V1.0/sysinfo/rtspurl

URL	**/HAPI/V1.0/sysinfo/rtspurl**
Description	**获取 RTSP URL**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/sysinfo/rtspurl?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	`{`
	`    "Response": {`
	`        "ResponseURL": "/HAPI/V1.0/sysinfo/rtspurl",`
	`        "SessionID": "",`
	`        "ResponseCode": 0,`
	`        "ResponseString": "Succeed",`
	`        "Data": {`
	`            "ch0_main": "rtsp://192.168.1.202:554/stream0",`
	`            "ch0_sub": "rtsp://192.168.1.202:554/stream1"`
	`        }`
	`    }`
	`}`

Note	
Status	

2.3 系统控制

2.3.1 重启设备 /HAPI/V1.0/sysman/reboot

URL	**/HAPI/V1.0/sysman/reboot**
Description	重启设备
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/sysman/reboot?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/sysman/reboot",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": "null"
	    }
	}

Note	
Status	

2.3.2 恢复出厂 /HAPI/V1.0/sysman/factory

URL	**/HAPI/V1.0/sysman/ factory**
Description	恢复出厂
Sample	http://192.168.1.202/HAPI/V1.0/sysman/factory?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**
Method	**GET/PUT**
Input Data	无

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/sysman/factory",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": "null"
	    }

	}
Note	
Status	

2.4 IO 状态控制

2.4.1 获取 IO 输入状态 /HAPI/V1.0/io/input/get

URL	**/HAPI/V1.0/io/input/get**
Description	获取 **IO** 输入状态
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/io/input/get?chn=1&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**chn：IO 通道号，从 1 开始，根据 IPC 实际配置** **username/password 或者 uid**

Success Return Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/io/input/get",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": 0
	    }
	}

Note	**IO 通道号不存在时，返回错误：**
	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/io/input/get",
	        "SessionID": "",
	        "ResponseCode": -1,
	        "ResponseString": "params error",
	        "Data": "null"
	    }
	}

Status	

2.4.2 获取 IO 输出状态 /HAPI/V1.0/io/output/get

URL	**/HAPI/V1.0/io/output/get**
Description	获取 **IO** 输出状态

Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/io/output/get?chn=1&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**chn：IO 通道号，从 1 开始，根据 IPC 实际配置** **username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/io/output/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": 1
	}
	}

Note	**IO 通道号不存在时，返回错误：**
	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/io/output/get",
	"SessionID": "",
	"ResponseCode": -1,
	"ResponseString": "params error",
	"Data": "null"
	}
	}

Status	

2.4.3 设置 IO 输出状态 /HAPI/V1.0/io/output/set

URL	**/HAPI/V1.0/io/output/set**
Description	**获取 IO 输出状态**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/io/output/set?chn=1&status=0&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**chn：IO 通道号，从 1 开始，根据 IPC 实际配置** **username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/io/output/set",
	"SessionID": "",
	"ResponseCode": 0,

	"ResponseString": "Succeed", "Data": 0 } }

Note	**IO 通道号不存在时，返回错误：**
	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/io/output/set",
	"SessionID": "",
	"ResponseCode": -1,
	"ResponseString": "params error",
	"Data": "null"
	}
	}

Status	

2.5 手动抓图

2.5.1 手动抓图 /HAPI/V1.0/snapshot.cgi

URL	**/HAPI/V1.0/snapshot.cgi**
Description	获取图片
Sample	http://192.168.1.202/HAPI/V1.0/snapshot.cgi?stream=1&username=admin&password=e10adc3949ba59abbe56e057f20f883e http://192.168.1.202/HAPI/V1.0/snapshot.cgi?stream=0&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**Stream：抓图码流序号，stream=0 表示抓子码流，stream=1 表示抓主码流；** **username/password 或者 uid**
Method	**GET/PUT**
Input Data	无
Success Return Data	**Jpg** 图片
Note	
Status	

2.6 云台控制

2.6.1 云台停止 /HAPI/V1.0/ptz_ctrl/stop

URL	**/HAPI/V1.0/ptz_ctrl/stop**
Description	停止控制云台
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/ptz_ctrl/stop?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/stop",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": "null"
	}
	}

Note	
Status	

2.6.2 云台转动 /HAPI/V1.0/ptz_ctrl/move

URL	**/HAPI/V1.0/ptz_ctrl/move**
Description	控制云台转动方向，支持 8 向控制
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/ptz_ctrl/move?direction=left&autostop=500&speed=5&username=admin&password=e10adc3949ba59abbe56e057f20f883e

URL params(简单参数)
Input Data(PUT)
	direction：云台移动方向，取值范围："left", "right", "up", "down", "left_up", "right_up", "left_down", "right_down";
	speed：云台速度，取值范围 1-10;
	autostop：控制后达到指定毫秒时自动停止，阻塞式调用，大于 1000 毫秒的数值将在 1000 毫秒是自动停止。不填或者为 0 时表示不自动停止
	username/password 或者 uid

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/move",
	"SessionID": "",

	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"direction": "left",
	"speed": 5,
	"autostop": 500
	}
	}
	}

Note	
Status	

2.6.3 云台预置点操作 /HAPI/V1.0/ptz_ctrl/preset

URL	**/HAPI/V1.0/ptz_ctrl/preset**
Description	**控制云台转动方向，支持 8 向控制**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/ptz_ctrl/preset?method=set&presetno=1&username=admin&password=e10adc3949ba59abbe56e057f20f883e http://192.168.1.202/HAPI/V1.0/ptz_ctrl/preset?method=call&presetno=2&uid=3CFABD6 http://192.168.1.202/HAPI/V1.0/ptz_ctrl/preset?method=delete&presetno=3&uid=3CFABD6
URL params(简单参数) **Input Data(PUT)**	**method：预置点操作方法，取值范围："set", "call", "delete"；** **presetno：预置点序号，**取值范围 1-255； **username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/move",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	" Data": {
	"direction": "left",
	"speed": 5
	}
	}
	}

Note	
Status	

2.6.4 云台变倍操作 /HAPI/V1.0/ptz_ctrl/zoom

URL	**/HAPI/V1.0/ptz_ctrl/zoom**
Description	控制云台变倍
Method	**GET/PUT**
Sample	放大： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/zoom?direction=in&autostop=500&username=admin&password=e10adc3949ba59abbe56e057f20f883e 缩小： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/zoom?direction=out&autostop=100&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**Direction：变倍方向，取值范围："in", "out"；** **autostop：控制后达到指定毫秒时自动停止，阻塞式调用，大于 1000 毫秒的数值将在 1000 毫秒是自动停止。不填或者为 0 时表示不自动停止** **username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/zoom",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"direction": "in",
	"autostop": 500
	}
	}
	}

Note	Response 中携带控制方向。
Status	

2.6.5 云台聚焦操作 /HAPI/V1.0/ptz_ctrl/focus

URL	**/HAPI/V1.0/ptz_ctrl/focus**
Description	控制云台聚焦
Method	**GET/PUT**
Sample	向近处聚焦： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/focus?direction=near&autostop=100&username=admin&password=e10adc3949ba59abbe56e057f20f883e 向远处聚焦： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/focus?direction=far&autostop=100&username=admin&password=e

	10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**Direction：聚焦方向，取值范围："near", "far"；** **autostop：控制后达到指定毫秒时自动停止，阻塞式调用，大于 1000 毫秒的数值将在 1000 毫秒是自动停止。不填或者为 0 时表示不自动停止；** **username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/ptz_ctrl/focus",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": {
	            "direction": "near",
	            "autostop": 100
	        }
	    }
	}

Note	**Response 中携带控制方向。**
Status	

2.6.6 云台光圈操作 /HAPI/V1.0/ptz_ctrl/iris

URL	**/HAPI/V1.0/ptz_ctrl/iris**
Description	控制云台光圈
Method	**GET/PUT**
Sample	光圈加大： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/iris?direction=open&autostop=100&username=admin&password=e10adc3949ba59abbe56e057f20f883e 光圈缩小： http://192.168.1.202/HAPI/V1.0/ptz_ctrl/iris?direction=close&autostop=100&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**direction：光圈控制方向，取值范围："open", "close"；** **autostop：控制后达到指定毫秒时自动停止，阻塞式调用，大于 1000 毫秒的数值将在 1000 毫秒是自动停止。不填或者为 0 时表示不自动停止；** **username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/ptz_ctrl/iris",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",

	"Data": {
	"direction": "open",
	"autostop": 100
	}
	}
	}

Note	Response 中携带控制方向。
Status	

2.6.7 云台高级功能执行操作 /HAPI/V1.0/ptz_ctrl/advfunction/exec

URL	**/HAPI/V1.0/ptz_ctrl/advfunction/exec**
Description	**执行云台高级功能**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/ptz_ctrl/advfunction/exec?functionname=PTZReset&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**functionname：高级功能名称，取值范围需要根据设备配置来确定；** **username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/advfunction/exec",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	" Data": {
	"functionname": "PTZReset"
	}
	}
	}

Note	设备可以配置一系列云台高级功能，每个高级功能执行某一个或多一个预置点的设置或调用。
Status	

2.6.8 云台高级功能列表 /HAPI/V1.0/ptz_ctrl/advfunction/get

URL	**/HAPI/V1.0/ptz_ctrl/advfunction/get**
Description	**获取设备云台高级功能列表**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/ptz_ctrl/advfunction/get?username=admin&password=e10

	adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/ptz_ctrl/advfunction/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": [
	{
	"functionname": "GuardPositionOff"
	},
	…………
	{
	"functionname": "TrackAutoOff"
	},
	{
	"functionname": "TrackAutoOn"
	}
	]
	}
	}

Note	设备可以配置一系列云台高级功能，每个高级功能执行某一个或多一个预置点的设置或调用。本 API 用于获取已经配置的更高级功能列表。
Status	

2.7 系统时间

2.7.1 获取设备时间和配置 /HAPI/V1.0/systime/gettime

URL	**/HAPI/V1.0/systime/gettime**
Description	**获取设备当前时间和时间配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/systime/gettime?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**username/password 或者 uid**

Success Return Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/systime/gettime",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"timeMode": "MANUAL",
	"timeZone": 1200,
	"nowtime": "2023-11-06 20:13:46"
	}
	}
	}

Note	timeMode：校时方式，Manual，P2P，NTP timeZone：时区，整数，（时区+12）*60，例如东 8 区的 timeZone 值为 1200 nowtime：设备本地时间字符串，按照年月日时分秒顺序格式化为"%04d-%02d-%02d %02d:%02d:%02d"
Status	

2.7.2 设置设备时间和时区 /HAPI/V1.0/systime/settime

URL	**/HAPI/V1.0/systime/settime**
Description	设置设备时间
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/systime/settime?localtime=20231106202840&timeZone=1260&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**timeZone：时区，整数，（时区+12）*60，例如东 8 区的 timeZone 值为 1200，可缺省，不填的时候不改变设备时区；** **localtime：设置本地时间字符串，按照年月日时分秒顺序格式化为"%04d%02d%02d%02d%02d%02d"，可以缺省，不填的时候不改变设备时间；** **username/password 或者 uid**
Success Return Data	参见 2.7.1GET 返回数据。
Note	"
Status	

2.7.3 设置设备 NTP 配置 /HAPI/V1.0/systime/setntp

URL	**/HAPI/V1.0/systime/**setntp
Description	**设置设备 NTP 配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/systime/setntp?serverIP=ipvs.icamra.com&username=admi

	n&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**NTP serverIP/serverPort/refreshInterval，username/password 或者 uid**
Success Return Data	参见 2.7.1GET 返回数据。
Note	serverPort 可缺省，默认为 123. refreshInterval 可缺省，默认为 60（秒）
Status	

2.8 灯光配置

2.8.1 获取支持的灯光开关控制模式能力集/HAPI/V1.0/system/light/ctrlmode/capability

URL	**/HAPI/V1.0/system/light/ctrlmode/capability**
Description	获取支持的灯光开关控制模式列表
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/light/ctrlmode/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/system/light/ctrlmode/capability ",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": [
	            {
	                "mode": 0
	            },
	            {
	                "mode": 1
	            },
	            {
	                "mode": 2
	            },
	            {
	                "mode": 3
	            }

	]
	    }
	}

Note	灯光开关控制模式（也叫 IRCUT 控制模式、日夜切换模式），用于配置开灯、切换 IRCUT、以及图像彩转灰的控制模式，取值范围为： 0：主动模式/软光敏自动控制模式，ISP 自动判断 SENSOR 增益来控制 1：日夜模式，根据时间段来控制 2：被动模式/硬光敏外部控制模式，灯板根据自身光敏电阻给的硬件信号来开灯，并通过高低电平通知摄像机，摄像机根据该信号来切换 IRCUT 和图像彩转灰 3：手动模式，由调用者来手动切换，主要用于界面上测试 IRCUT 功能 4：反向被动模式，已废弃 5： 硬光敏自动控制模式，模组根据光敏电阻硬件的 adc 数据来控制 6：手动灯光常开 7：手动灯光常关
Status	

2.8.2 获取支持的灯光工作模式能力集/HAPI/V1.0/system/light/workmode/capability

URL	**/HAPI/V1.0/system/light/workmode/capability**
Description	获取支持的灯光开关控制模式列表
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/light/workmode/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/system/light/workmode/capability",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": [
	            {
	                "mode": 0
	            },
	            {
	                "mode": 1
	            },

	{
	    "mode": 2
	}
	]
	}
	}

Note	灯光工作模式，用于配置开灯时补光灯的工作模式，取值范围： 0：纯红外，需要开灯的条件下只开红外灯 1：纯白光，需要开灯的条件下只开白光灯 2：智能双光，需要开灯的条件下，根据图像智能检测结果，有检测事件时开白光灯并保持彩色图像，没有检测事件时开红外灯
Status	

2.8.3 获取灯光配置/HAPI/V1.0/system/light/get

URL	**/HAPI/V1.0/system/light/get**
Description	**获取灯光配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/light/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/system/light/get",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": {
	            "led_ctrl_mode": 2,
	            "led_work_mode": 0,
	            "light_open_brightness": 3,
	            "light_off_sensitivity": 50,
	            " led_brightness_mode": 0,
	            "led_brightness_value": 100,
	            "night_starttime": "18:00:00",
	            "night_endtime": "08:00:00"
	        }
	    }
	}

Note	led_ctrl_mode：灯光开关控制模式，取值范围参见 2.8.1 led_work_mode：灯光工作模式，取值范围参见 2.8.2； light_open_brightness：开启环境光亮度，取值范围 0-17，值越小说明环境亮度越低才开灯； light_off_sensitivity:关灯灵敏度，取值范围 0-100，用于控制开灯后亮度变化后的关灯灵敏度； led_brightness_mode：补光灯亮度模式：取值范围 0-2（0：自动；1：手动；2：低功耗） led_brightness_value：补光亮度，取值范围 0-100； night_starttime:　　　夜晚开始时间，用于按照配置的日夜时间来定时开灯，按照时分秒顺序格式化为 "%02d:%02d:%02d"； night_endtime:　　　夜晚结束时间，用于按照配置的日夜时间来定时关灯，按照时分秒顺序格式化为 "%02d:%02d:%02d" ；
Status	

2.8.4 设置灯光配置/HAPI/V1.0/system/light/set

URL	**/HAPI/V1.0/system/light/set**
Description	**设置灯光配置**
Method	**GET/PUT**
Sample	设置灯光控制模式为日夜模式： http://192.168.1.202/HAPI/V1.0/system/light/set?led_ctrl_mode=1&night_endtime=093000&night_starttime=153500&username=admin&password=e10adc3949ba59abbe56e057f20f883e 设置补光灯工作模式为智能双光： http://192.168.1.202/HAPI/V1.0/system/light/set?led_work_mode=2&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	填充以下参数中的一个或者多个参数，缺少参数或者范围不正确将返回错误： led_ctrl_mode：灯光开关控制模式，取值范围参见 2.8.1 led_work_mode：灯光工作模式，取值范围参见 2.8.2； light_open_brightness：开启环境光亮度，取值范围 0-17，值越小说明环境亮度越低才开灯； light_off_sensitivity:关灯灵敏度，取值范围 0-100，用于控制开灯后亮度变化后的关灯灵敏度； led_brightness_mode：补光灯亮度模式：取值范围 0-2（0：自动；1：手动；2：低功耗） led_brightness_value：补光亮度，取值范围 0-100； night_starttime:　　　夜晚开始时间，用于按照配置的日夜时间来定时开灯，按照时分秒顺序格式化为 "%02d:%02d%:02d"； night_endtime:　　　夜晚结束时间，用于按照配置的日夜时间来定时关灯，按照时分秒顺序格式化为 "%02d:%02d:%02d" ； username/password 或者 uid
Success Response Data	参见 2.8.3GET 返回数据。
Note	返回数据中携带当前灯光配置数据
Status	

2.9 图像配置

2.9.1 获取图像配置/HAPI/V1.0/system/image/get

URL	**/HAPI/V1.0/system/image/get**
Description	获取图像当前配置
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/image/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/system/image/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"brightness": 128,
	"contrast": 128,
	"saturation": 128,
	"sharpness": 128,
	"tvsystem": 1,
	"hflip": 0,
	"vflip": 0,
	"rotate": 0
	}
	}
	}

Note	brightness：亮度，0-255 contrast：对比度，0-255 saturation：饱和度，0-255 sharpness：锐度：0-255 tvsystem：视频制式　0：NTSC（60HZ），1：PAL（50HZ） hflip：水平翻转 vflip：垂直翻转 rotate：走廊模式（旋转 90 度，部分摄像机支持）
Status	

2.9.2 设置图像配置/HAPI/V1.0/system/image/set

URL	**/HAPI/V1.0/system/image/set**
Description	设置图像配置
Method	**GET/PUT**
Sample	设置亮度： http://192.168.1.202/HAPI/V1.0/system/image/set?brightness=129&username=admin&password=e10adc3949ba59abbe56e057f20f883e 设置视频制式： http://192.168.1.202/HAPI/V1.0/system/image/set?tvsystem=1&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	填充以下参数中的一个或者多个参数，缺少参数或者范围不正确将返回错误： brightness：亮度，0-255 contrast：对比度，0-255 saturation：饱和度，0-255 sharpness：锐度：0-255 tvsystem：视频制式 0：NTSC（60HZ），1：PAL（50HZ） hflip：水平翻转 vflip：垂直翻转 rotate：走廊模式（旋转 90 度，部分摄像机支持）； username/password 或者 uid
Success Response Data	参见 2.9.1 GET 返回数据。
Note	返回数据中携带当前图像配置数据
Status	

2.10 视频配置

2.10.1 获取视频编码能力集/HAPI/V1.0/system/video/capability

URL	**/HAPI/V1.0/system/video/capability**
Description	获取支持的视频编码能力集
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/video/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {

```
        "ResponseURL": "/HAPI/V1.0/system/video/capability",
        "SessionID": "",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": [
            {
                "codec_name": "H265+",
                "res_name": "3840X2160",
                "stream_type": 0,
                "def_bitrate": 6000,
                "min_bitrate": 512,
                "max_bitrate": 12288,
                "def_framerate": 20,
                "min_framerate": 5,
                "max_framerate": 20,
                "def_config": 0
            },
       …………
            {
                "codec_name": "H264",
                "res_name": "CIF",
                "stream_type": 2,
                "def_bitrate": 150,
                "min_bitrate": 64,
                "max_bitrate": 2048,
                "def_framerate": 10,
                "min_framerate": 5,
                "max_framerate": 10,
                "def_config": 0
            }
        ]
    }
}
```

Note	支持的视频编码列表以 json 数组返回。数组成员含义: codec_name:编码名称，取值为 H264、H265、H265+、MJPEG 等 res_name: 分辨率名称，常规情况下以分辨率宽 X 高表示，例如 2560X1440，部分常规分辨率用 1080P/CIF 等字符串直接表示 stream_type:码流序号，0 为主码流，1 为子码流，2 为三码流， def_bitrate:选择该分辨率时的默认码率 min_bitrate:该分辨率支持的最小码率 max_bitrate:该分辨率支持的最大码率 def_framerate:选择该分辨率时的默认帧率 min_framerate:该分辨率支持的最小帧率

	max_framerate：该分辨率支持的最大帧率 def_config：该分辨率是否为摄像机的默认配置
Status	

2.10.2 获取视频编码配置/HAPI/V1.0/system/video/get

URL	**/HAPI/V1.0/system/video/get**
Description	**获取视频编码配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/video/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/system/video/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"stream": [
	{
	"enable": 1,
	"streamID": 1,
	"encodeFormat": "H265",
	"resolution": "3840X2160",
	"bitRateControl": "VBR",
	"gop": 80,
	"frameRate": 20,
	"bitRate": 6000,
	"bitRateQuality": 0,
	"qp_enable": 0,
	"qp_min": 30,
	"qp_max": 51
	},
	{
	"enable": 1,
	"streamID": 2,
	"encodeFormat": "H265",
	"resolution": "640X360",
	"bitRateControl": "VBR",
	"gop": 80,
	"frameRate": 20,

	"bitRate": 700,
	"bitRateQuality": 0,
	"qp_enable": 0,
	"qp_min": 30,
	"qp_max": 51
	},
	{
	"enable": 0,
	"streamID": 3,
	"encodeFormat": "H265",
	"resolution": "720P",
	"bitRateControl": "VBR",
	"gop": 40,
	"frameRate": 10,
	"bitRate": 1000,
	"bitRateQuality": 0,
	"qp_enable": 0,
	"qp_min": 28,
	"qp_max": 51
	}
	]
	}
	}

Note	视频编码配置列表以 json 数组返回。每个数组成员分别表示主码流、子码流、三码流，数组成员中各元素含义： Enable：使能标记 streamID：码流 ID，1 为主码流，2 为子码流，3 为三码流。注意，由于历史原因，这里的数字定义跟/HAPI/V1.0/system/video/capability 接口列出的 stream_type 取值定义不一样。 encodeFormat：编码名称，取值为 H264、H265、H265+、MJPEG 等 resolution：分辨率名称，常规情况下以分辨率宽 X 高表示，例如 2560X1440，部分常规分辨率用 1080P/CIF 等字符串直接表示 bitRateControl：码率控制方式，取值为 VBR、CBR 等 gop：I 帧间隔，即每隔多少帧产生一个 I 帧 frameRate：帧率 bitRate：码率 bitRateQuality：视频质量， 取值范围 0-5。0 表示自定义，1 表示设置为允许的最低码率，2 表示设置为最低码率有默认码率的中间值，3 表示设置为默认码率，4 表示设置为默认码率与最高码率的中间值，5 表示设置为允许的最高码率 qp_enable：使能自定义 QP qp_min：自定义 QP 的最小 QP 值，1-51 qp_max：自定义 QP 的最大 QP 值，1-51
Status	

2.10.3 设置视频编码配置/HAPI/V1.0/system/video/set

URL	/HAPI/V1.0/system/video/set
Description	设置视频编码
Method	GET/PUT

Sample	设置主码流分辨率：
	http://192.168.1.202/HAPI/V1.0/system/video/set?streamId=1&resolution=3840X2160&username=admin&password=e10adc3949ba59abbe56e057f20f883e
	PUT 方法：
	PUT /HAPI/V1.0/system/video/set HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	content-type: application/json
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Host: 192.168.1.202
	Content-Length: 116
	{
	"streamId":"1",
	"resolution":"3840X2160",
	"username":"admin",
	"password":"e10adc3949ba59abbe56e057f20f883e"
	}
	设置子码流编码、分辨率、帧率：
	http://192.168.1.202/HAPI/V1.0/system/video/set?streamId=2&encodeFormat=H264&resolution=VGA&frameRate=25&username=admin&password=e10adc3949ba59abbe56e057f20f883e
	设置主码流码率、帧率：
	设置播放音量：
	http://192.168.1.202/HAPI/V1.0/system/audio/set?volume_play=80&username=admin&password=e10adc3949ba59abbe56e057f20f883e

URL params(简单参数) Input Data(PUT)	填充以下参数中的一个或者多个参数，缺少参数或者范围不正确将返回错误： streamID：码流 ID，1 为主码流，2 为子码流，3 为三码流。注意，由于历史原因，这里的数字定义跟 /HAPI/V1.0/system/video/capability 接口列出的 stream_type 取值定义不一样。 enable：使能标记 encodeFormat：编码名称，取值为 H264、H265、H265+、MJPEG 等 resolution：分辨率名称，常规情况下以分辨率宽 X 高表示，例如 2560X1440，部分常规分辨率用 1080P/CIF 等字符串直接表示

	bitRateControl：码率控制方式，取值为 VBR、CBR 等 gop：I 帧间隔，即每隔多少帧产生一个 I 帧 frameRate：帧率 bitRate：码率 bitRateQuality：视频质量， 取值范围 0-5。0 表示自定义，1 表示设置为允许的最低码率，2 表示设置为最低码率有默认码率的中间值，3 表示设置为默认码率，4 表示设置为默认码率与最高码率的中间值，5 表示设置为允许的最高码率 qp_enable：使能自定义 QP qp_min：自定义 QP 的最小 QP 值，1-51 qp_max：自定义 QP 的最大 QP 值，1-51 username/password 或者 uid
Success Response Data	参见 2.10.2GET 返回数据。
Note	设置音频参数成功的情况下，返回数据中携带当前音频配置数据
Status	

2.11 音频配置

2.11.1 获取音频配置编码能力集

/HAPI/V1.0/system/audio/capability

URL	**/HAPI/V1.0/system/audio/capability**
Description	获取支持的音频编码能力集
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/audio/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/system/audio/capability",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": [
	            {
	                "codec_name": "G.711",
	                "channels": 1,
	                "bitspersample": 16,
	                "samplerate": 8,
	                "bitrate": 64,

	"def_config": 1
	},
	…………
	{
	"codec_name": "AAC",
	"channels": 2,
	"bitspersample": 16,
	"samplerate": 16,
	"bitrate": 16,
	"def_config": 0
	}
	]
	}
	}

Note	支持的视频编码列表以 json 数组返回。数组成员含义： codec_name：编码名称，取值为 G.711、G.711A、AAC 等 channels：音频通道数 bitspersample ：采样位数，即每个采样占用的比特数 samplerate：采样率 bitrate：比特率 def_config：该编码类型是否为摄像机的默认配置
Status	

2.11.2 获取音频配置/HAPI/V1.0/system/audio/get

URL	**/HAPI/V1.0/system/audio/get**
Description	**获取音频配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/system/audio/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/system/audio/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"enable": 1,

	"sampleRate": 8000, "encodeFormat": "G.711", "bitRate": 64000, "volume_capture": 90, "volume_play": 95, "amplify": 1, "aec_enable": 1, "mute_ptz_turn": 0 } } }
Note	Enable：使能标记 sampleRate：音频编码采样率 encodeFormat：编码名称，编码名称，取值为 G.711、G.711A、AAC 等 bitRate：音频编码比特率 volume_capture：音频输入音量 volume_play：音频输出音量 amplify：使能设备内部软件功放 aec_enable：使能回声消除（部分设备支持） mute_ptz_turn：使能云台转动时静音功能
Status	

2.11.3 设置音频配置/HAPI/V1.0/system/audio/set

URL	**/HAPI/V1.0/system/audio/set**
Description	**获取音频配置**
Method	**GET/PUT**
Sample	设置编码： http://192.168.1.202/HAPI/V1.0/system/audio/set?encodeFormat=G.711&username=admin&password=e10adc3949ba59abbe56e057f20f883e 设置播放音量： http://192.168.1.202/HAPI/V1.0/system/audio/set?volume_play=80&username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	填充以下参数中的一个或者多个参数，缺少参数或者范围不正确将返回错误： enable：使能音频编码 sampleRate：音频编码采样率 encodeFormat：编码名称，编码名称，取值为 G.711、G.711A、AAC 等 bitRate：音频编码比特率 volume_capture：音频输入音量 volume_play：音频输出音量 amplify：使能设备内部软件功放

	aec_enable：使能回声消除（部分设备支持） mute_ptz_turn：使能云台转动时静音功能 username/password 或者 uid
Success Response Data	参见 2.11.2GET 返回数据。
Note	设置参数成功的情况下，返回当前配置
Status	

2.12 OSD 配置

2.12.1 获取 OSD 配置/HAPI/V1.0/system/osd/get

<table>
<tr><td><b>URL</b></td><td><b>/HAPI/V1.0/system/osd/get</b></td></tr>
<tr><td><b>Description</b></td><td><b>获取视频编码配置</b></td></tr>
<tr><td><b>Method</b></td><td><b>GET/PUT</b></td></tr>
<tr><td><b>Sample</b></td><td>http://192.168.1.202/HAPI/V1.0/system/osd/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e</td></tr>
<tr><td><b>URL params(简单参数)</b>
<b>Input Data(PUT)</b></td><td><b>username/password 或者 uid</b></td></tr>
<tr><td><b>Success Response Data</b></td><td><pre>
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/system/osd/get",
        "SessionID": "",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": {
            "enable": 1,
            "add_overlay": 3,
            "style": 3,
            "DsplayWeek": 1,
            "OverlayFps": 0,
            "fontsize": 0,
            "time24or12": 0,
            "timeOverlay": {
                "posType": 0,
                "posX": 1,
                "posY": 1,
                "timeFormat": "yyyy-mm-dd hh:mm:ss"
            },
            "titleOverlay": {
                "posType": 0,
</pre></td></tr>
</table>

```
                "posX": 0,

                "posY": 0,

                "titleType": 0,

                "title_utf8": "e6b58be8af9543616d657261"

            }

        }

    }

}
```

Note

enable：OSD 使能

add_overlay：叠加附加信息。0：无，1：叠加分辨率，2：叠加码率，3：叠加分辨率和码率

style：叠加风格，取值范围参考枚举值定义：

```
          typedef enum

          {

            AJ_OVERLAY_STYLE_BLACK_WHITE = 0,        //黑字白底

            AJ_OVERLAY_STYLE_WHITE_BLACK = 1,        //白字黑底

            AJ_OVERLAY_STYLE_TRANSPARENT_BLACKWHITE = 2,   //透明背景，黑字白框

            AJ_OVERLAY_STYLE_TRANSPARENT_WHITEBLACK = 3,   //透明背景，白字黑框

            AJ_OVERLAY_STYLE_TRANSPARENT_BLACK = 4,  //透明背景，黑字

            AJ_OVERLAY_STYLE_TRANSPARENT_WHITE = 5, //透明背景，白字

            AJ_OVERLAY_STYLE_INVERSE_COLOR = 6, //反色

          }AjOsdOverlayStyle;
```

DsplayWeek:显示星期几

OverlayFps：叠加帧率

fontsize:叠加文字大小，0：标准 1：大字体 2：超大字体

time24or12：0:12 小时格式，1：24 小时格式

timeOverlay：时间 OSD 配置

titleOverlay：标题 OSD 配置

posType：坐标类型，0 表示 posX、posY 配置的是 4 角位置，1 表示 posX、posY 配置的是画面百分比位置

posX:坐标类型为四角位置时，0 画面左边，1 表示画面右边，2 表示不显示。坐标类型为百分比时，取值范围 0-100，表示位置为在画面水平尺寸的百分比。

posY:坐标类型为四角位置时，0 画面上边，1 表示画面下边，2 表示不显示。坐标类型为百分比时，取值范围 0-100，表示位置为在画面垂直尺寸的百分比。

timeFormat：时间格式，取值范围：

```
            "yyyy-mm-dd hh:mm:ss",

            "yyyy/mm/dd hh:mm:ss",

            "yy-mm-dd hh:mm:ss",

            "yy/mm/dd hh:mm:ss",

            "hh:mm:ss dd/mm/yyyy",

            "hh:mm:ss dd-mm-yyyy",

            "hh:mm:ss mm/dd/yyyy",

            "hh:mm:ss mm-dd-yyyy",

            "mm/dd/yyyy hh:mm:ss",

            "mm-dd-yyyy hh:mm:ss"；
```

titleType：标题类型，0 表示叠加文字，1 表示叠加图片

title_utf8：当 titleType 为 0 时，内容为 utf8 格式的 16 进制格式字符串（utf8 格式无法直接用 ascii 码字符串存放和输出，所以将每个字节按 16 进制格式化成 2 个字节的 ascii 字符）；当 titleType 为 1 时，内容为摄像机上 bmp 图片绝对路径。

utf8 字符串转换成 16 进制格式字符串的 API 参考：

```
int hexdataTohexStr(const char* buf, unsigned int len, char* out, unsigned int outbuflen)
{
    if( outbuflen == 0 )
        return -1;

    if( outbuflen <= 2 )
    {
        out[0] = 0;
    }

    if( len >= (outbuflen-1)/2 )
        len = (outbuflen-1)/2;

    unsigned int iIndex;
    for(iIndex = 0; iIndex < len; iIndex++)
    {
        sprintf(&out[2*iIndex], "%02x", (unsigned char)buf[iIndex]);
    }

    return 0;
}
```

Status	

2.12.2 设置 OSD 配置/HAPI/V1.0/system/osd/set

URL	**/HAPI/V1.0/system/osd/set**
Description	获取视频编码配置
Method	**PUT**
Sample	PUT /HAPI/V1.0/system/osd/set HTTP/1.1 User-Agent: PostmanRuntime-ApipostRuntime/1.1.0 Cache-Control: no-cache content-type: application/json Accept: */* Accept-Encoding: gzip, deflate, br Connection: keep-alive Host: 192.168.1.202 Content-Length: 404

	{
	"username": "admin",
	"password": "123456",
	"enable": 1,
	"add_overlay": 2,
	"style": 3,
	"DsplayWeek": 1,
	"OverlayFps": 0,
	"fontsize": 0,
	"time24or12": 0,
	"timeOverlay": {
	"posType": 1,
	"posX": 50,
	"posY": 50,
	"timeFormat": "hh:mm:ss mm-dd-yyyy"
	},
	"titleOverlay": {
	"posType": 0,
	"posX": 0,
	"posY": 0,
	"titleType": 0,
	"title_utf8": "43616d657261"
	}
	}

URL params(简单参数)
Input Data(PUT)
	填充以下参数中的一个或者多个参数，缺少参数或者范围不正确将返回错误。参数取值范围参见 2.12.1
	
	{
	"username": "admin",
	"password": "123456",
	"enable": 1,
	"add_overlay": 2,
	"style": 3,
	"DsplayWeek": 1,
	"OverlayFps": 0,
	"fontsize": 0,
	"time24or12": 0,
	"timeOverlay": {
	"posType": 1,
	"posX": 50,
	"posY": 50,
	"timeFormat": "hh:mm:ss mm-dd-yyyy"
	},
	"titleOverlay": {
	"posType": 0,

	"posX": 0, "posY": 0, "titleType": 0, "title_utf8": "43616d657261" } }
Success Response Data	参见 2.12.1GET 返回数据。
Note	设置参数成功的情况下，返回当前配置
Status	

2.13 智能检测配置

2.13.1 获取智能检测能力集/HAPI/V1.0/smart/capability

URL	**/HAPI/V1.0/smart/capability**
Description	获取智能检测能力集
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/Smart/capability",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"SupportTargetDetect": 1,
	"SupportFaceFd": 0,
	"SupportFaceFr": 0,
	"SupportLPR": 0,
	"SupportVG": 0,
	"SupportRegionAI": 0,
	"SupportAudioDetect": 1,
	"SupportCoverDetect": 1
	}
	}
	}

Note	SupportTargetDetect:具备人车非目标检测能力

	SupportFaceFd:具备人脸检测能力 SupportFaceFr:具备人脸识别能力 SupportLPR:具备车牌识别能力 SupportVG:具备拌线检测能力 SupportRegionAI:具备区域智能检测（逗留、进入、离开区域等）能力 SupportAudioDetect:支持异常声音检测能力 SupportCoverDetect:支持视频遮挡检测能力
Status	

2.13.2 获取目标检测（人车非机动车）算法支持的目标类型 /HAPI/V1.0/smart/objectdetect /capability

URL	**/HAPI/V1.0/smart/objectdetect/capability**
Description	获取智能检测能力集
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/objectdetect/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/Smart/objectdetect/capability",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": [
	{
	"type": 4
	}
	]
	}
	}

Note	支持的目标类型由 json 数组返回，type 定义：
	typedef enum
	{
	AI_TYPE_BIT_CAR = 0, //汽车
	AI_TYPE_BIT_MOTO = 1, //摩托车
	AI_TYPE_BIT_ELECTRICBICYCLE = 2,//电单车
	AI_TYPE_BIT_BICYCLE = 3,//自行车

	AI_TYPE_BIT_HUMAN = 4, //人形
	AI_TYPE_BIT_FACE = 5, //人脸
	AT_TYPE_BIT_NONMOTO_VEHICLE = 6,//非机动车
	AT_TYPE_BIT_FIRE=7,//火焰
	AT_TYPE_BIT_FALLINGOBJECT=8,//高空抛物
	AI_TYPE_BIT_MAX,
	}AjAiBits;

Status	

2.13.3 获取目标检测（人车非机动车）算法支持联动能力 /HAPI/V1.0/smart/linkage/capability

URL	**/HAPI/V1.0/smart/linkage/capability**
Description	获取智能检测联动能力集
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/linkage/capability?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/Smart/linkage/capability",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": [
	{
	"SupportLightAction": 0
	},
	{
	"SupportBRAlarmAction": 0
	},
	{
	"SupportAudioAction": 1
	},
	{
	"SupportIOOutAction": 1
	}
	]
	}
	}

Note	SupportLightAction：支持补光灯联动闪烁（有检测到目标时闪烁补光灯） SupportBRAlarmAction：支持红蓝警灯联动 SupportAudioAction：支持警戒音联动 SupportIOOutAction：支持 IO 输出联动
Status	

2.13.4 获取警戒音列表/HAPI/V1.0/smart/audiofiles/get

URL	**/HAPI/V1.0/smart/audiofiles/get**
Description	获取设备警戒音列表
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/audiofiles/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/Smart/audiofiles/get",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": [
	            {
	                "filename": "/opt/ch/attention_en.mp3",
	                "UnicodeDesc":
	"fffe5700610072006e002000720065006d0069006e006400650072002c00200059006f007500200068006100760065002000
	65006e0074006500720065006400200074006800650020006d006f006e00690074006f00720069006e0067002000610072006
	5006100"
	            },
	            {
	                "filename": "/opt/ch/attention03.mp3",
	                "UnicodeDesc": "fffec179ba4e8698df57f78bff526097d18f"
	            },
	            {
	                "filename": "/opt/ch/attention02.mp3",
	                "UnicodeDesc": "fffea860f25ddb8f6551dd8f81793a53df57f78b3d5ceb5fbb79005f"
	            },
	            {
	                "filename": "/opt/ch/attention.mp3",
	                "UnicodeDesc": "fffe296ea899d0633a79a860f25ddb8f6551d176a7633a53df57"

	},
	 {
	 "filename": "/opt/ch/Audio.mp3"
	 }
	]
	 }
	 }

Note	Filename 为警戒音绝对路径。 UnicodeDesc 为相应文件的描述信息，unicode 字符串的 16 进制格式化字符串。
Status	

2.13.5 获取运动侦测配置/HAPI/V1.0/motiondetect/get

URL	**/HAPI/V1.0/smart/motiondetect/get**
Description	获取运动侦测配置
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/motiondetect/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	 "Response": {
	 "ResponseURL": "/HAPI/V1.0/Smart/motiondetect/get",
	 "SessionID": "",
	 "ResponseCode": 0,
	 "ResponseString": "Succeed",
	 "Data": {
	 "enable": 1,
	 "ArmingFlag": {
	 "ArmingMode": 1,
	 "WeekPlan": {
	 "day0": 16777215,
	 "day1": 16777215,
	 "day2": 16777215,
	 "day3": 16777215,
	 "day4": 16777215,
	 "day5": 16777215,
	 "day6": 16777215
	 }
	 },

```
            "AreaConfig": {
                "block_col": 22,
                "block_row": 18,
                "blockconfig": [
                    {
                        "block_row": "0001111111111111111111"
                    },
                    {
                        "block_row": "0001111111111111111111"
                    },
                    {
                        "block_row": "0001111111111111111111"
                    },
                    {
                        "block_row": "0001111111111111111111"
                    },
                    {
                        "block_row": "1111111111111111111111"
                    },
                    {
                        "block_row": "1111111111111111111111"
                    },
                    {
                        "block_row": "1111111111111111111111"
                    },
                    {
                        "block_row": "1111111111111111111111"
                    },
                    {
                        "block_row": "1111111111110000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
```

```
                "block_row": "0000000000000000000000"
            },
            {
                "block_row": "0000000000000000000000"
            },
            {
                "block_row": "0000000000000000000000"
            },
            {
                "block_row": "0000000000000000000000"
            },
            {
                "block_row": "0000000000000000000000"
            }
        ]
    },
    "Sensitivity": 80,
    "AlarmThreshold": 20,
    "DayNightSwitch": 1,
    "NightSensitivity": 80,
    "NightAlarmThreshold": 20,
    "NightStartTime": "00:00:00",
    "NightEndTime": "00:00:00",
    "AlarmAction": {
        "LightAction": {
            "ArmingMode": 0,
            "TimeSpanNum": 0,
            "DayPlan": []
        },
        "BRAlarmAction": {
            "ArmingMode": 0,
            "TimeSpanNum": 0,
            "DayPlan": []
        },
        "AlarmServer": {
            "ArmingMode": 0,
            "TimeSpanNum": 0,
            "DayPlan": []
        },
        "AlarmPush": {
            "ArmingMode": 4,
            "TimeSpanNum": 1,
            "DayPlan": [
                {
                        "startTime": "00:00:00",
                        "endTime": "23:59:59"
                    }
                ]
            },
            "AudioAction": {
                "times": 1,
                "intervalsecnods": 10,
                "filename": "/opt/ch/attention_en.mp3",
                "ArmingSetting": {
                    "ArmingMode": 0,
                    "TimeSpanNum": 1,
                    "DayPlan": [
                        {
                            "startTime": "00:00:00",
                            "endTime": "23:59:59"
                        }
                    ]
                }
            },
            "IOOutputAction": {
                "channelCnt": 4,
                "channels": [
                    {
                        "portIndex": 1,
                        "enable": 0
                    },
                    {
                        "portIndex": 2,
                        "enable": 0
                    },
                    {
                        "portIndex": 3,
                        "enable": 0
                    },
                    {
                        "portIndex": 4,
                        "enable": 0
                    }
                ]
            }
        }
    }
}
```

	}
Note	enable：使能算法检测 ArmingFlag：报警检测布防配置，作为一个总的开关控制所有的联动输出，具体含义参见 4.1 AreaConfig：运动侦测区域配置，具体含义参见 4.3 Sensitivity:运动侦测灵敏度，0-100。数值越大越容易触发。 AlarmThreshold: 阈值，0-100，配置每个子区域中检测到变化像素的占比阈值，数值越小越容易触发。 DayNightSwitch: 日夜参数开关，启用时在夜晚时使用配置的夜晚灵敏度、阈值 NightSensitivity: 夜晚运动侦测灵敏度 NightAlarmThreshold: 夜晚阈值 NightStartTime: 夜晚开始时间 NightEndTime: 夜晚结束时间 AlarmAction：报警联动配置 LightAction：补光灯联动配置，具体含义参见 4.4 BRAlarmAction：红蓝警灯联动配置：具体含义参见 4.4 BRAlarmAction：红蓝警灯联动配置：具体含义参见 4.4 AlarmServer：联动上报服务器配置：具体含义参见 4.4 AlarmPush：报警推送联动配置：具体含义参见 4.4 AudioAction：警戒音联动配置：具体含义参见 4.5 IOOutputAction：IO 输出联动配置：具体含义参见 4.6
Status	

2.13.6 设置运动侦测配置/HAPI/V1.0/motiondetect/set

URL	**/HAPI/V1.0/smart/motiondetect/set**
Description	获取运动侦测配置
Method	**PUT**

Sample	PUT /HAPI/V1.0/Smart/motiondetect/set HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 1175
	
	{
	"username": "admin",
	"password": "e10adc3949ba59abbe56e057f20f883e",
	"AreaConfig": {
	"block_col": 22,
	"block_row": 18,

```
    "blockconfig": [
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111110000000000"
        },
        {
            "block_row": "0000000000000000000000"
        },
        {
            "block_row": "0000000000000000000000"
        },
        {
            "block_row": "0000000000000000000000"
        },
        {
            "block_row": "0000000000000000000000"
        },
        {
            "block_row": "0000000000000000000000"
        },
        {
```

	"block_row": "0000000000000000000000"
	},
	{
	"block_row": "0000000000000000000000"
	},
	{
	"block_row": "0000000000000000000000"
	},
	{
	"block_row": "0000000000000000000000"
	}
	]
	}
	}

URL params(简单参数) **Input Data(PUT)**	参考 2.13.5 获取的 JSON 数据，填充以下参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据（ArmingFlag/AreaConfi/以及 AlarmAction 节点下的 LightAction/BRAlarmAction/AlarmServer/AlarmPush/AudioAction/IOOutputAction 等）时，需要完整填充所有必须成员参数。 username password enable ArmingFlag AreaConfig Sensitivity AlarmThreshold DayNightSwitch NightSensitivity NightAlarmThreshold NightStartTime NightEndTime AlarmAction
Success Response Data	参见 2.13.5GET 返回数据。
Note	设置参数成功的情况下，返回当前配置。
Status	

2.13.7 获取目标检测配置/HAPI/V1.0/smart/objectdetect/get

URL	**/HAPI/V1.0/smart/objectdetect/get**
Description	**获取目标检测配置**
Method	**GET/PUT**

Sample	http://192.168.1.202/HAPI/V1.0/Smart/objectdetect/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**
Success Response Data	

```
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/Smart/objectdetect/get",
        "SessionID": "",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": {
            "enable": 1,
            "type": {
                "CAR": 0,
                "MOTO": 0,
                "ELECTRICBICYCLE": 0,
                "BICYCLE": 0,
                "HUMAN": 1,
                "FACE": 0,
                "NONMOTO_VEHICLE": 0,
                "FIRE": 0,
                "FALLINGOBJECT": 0
            },
            "ArmingFlag": {
                "ArmingMode": 1,
                "WeekPlan": {
                    "day0": 16777215,
                    "day1": 16777215,
                    "day2": 16777215,
                    "day3": 16777215,
                    "day4": 16777215,
                    "day5": 16777215,
                    "day6": 16777215
                }
            },
            "Polygon": {
                "PointCnt": 7,
                "Points": [
                    {
                        "x": 17,
                        "y": 11
                    },
                    {
```

```
            "x": 86,
            "y": 13
        },
        {
            "x": 95,
            "y": 83
        },
        {
            "x": 62,
            "y": 94
        },
        {
            "x": 16,
            "y": 90
        },
        {
            "x": 5,
            "y": 37
        },
        {
            "x": 8,
            "y": 14
        }
    ]
},
"Sensitivity": 5,
"AlarmThreshold": 75,
"minTargetRate": 0,
"nonMotionFilter": 1,
"AlarmAction": {
    "LightAction": {
        "ArmingMode": 0,
        "TimeSpanNum": 0,
        "DayPlan": []
    },
    "BRAlarmAction": {
        "ArmingMode": 1,
        "TimeSpanNum": 0,
        "DayPlan": []
    },
    "AlarmServer": {
        "ArmingMode": 0,
        "TimeSpanNum": 0,
        "DayPlan": []
    },
    "AlarmPush": {
        "ArmingMode": 1,
        "TimeSpanNum": 0,
        "DayPlan": []
    },
    "AudioAction": {
        "times": 1,
        "intervalsecnods": 10,
        "filename": "/opt/ch/attention.mp3",
        "ArmingSetting": {
            "ArmingMode": 4,
            "TimeSpanNum": 3,
            "DayPlan": [
                {
                    "startTime": "01:00:00",
                    "endTime": "13:59:00"
                },
                {
                    "startTime": "15:00:00",
                    "endTime": "17:00:00"
                },
                {
                    "startTime": "21:00:00",
                    "endTime": "22:00:00"
                }
            ]
        }
    },
    "IOOutputAction": {
        "channelCnt": 4,
        "channels": [
            {
                "portIndex": 1,
                "enable": 0
            },
            {
                "portIndex": 2,
                "enable": 0
            },
            {
                "portIndex": 3,
                "enable": 0
            },
                    {
                        "portIndex": 4,
                        "enable": 0
                    }
                ]
            },
            "draw_rect_enable": 1,
            "draw_human_enable": 1,
            "track_human_enable": 0,
            "rect_twinkle_enable": 0,
            "auto_zoom_enable": 0,
            "gunball_track_mode": 0
        }
    }
}
}
```

Note	"enable": 算法启用，控制算法的检测，用于实现摄像机的软光敏开关灯等逻辑，以及将报警通知给其他联动模块 "type": 目标检测类型，参见 4.7 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Polygon": 检测区域 "Sensitivity": 灵敏度，取值范围 0-10 "AlarmThreshold": 报警阈值（暂未使用） "minTargetRate": 检测最小目标的画面比例，0-100 "nonMotionFilter": 不动不检，0-1 "LightAction": 补光灯联动，有报警时闪烁补光灯 "BRAlarmAction": 红蓝警灯联动，有报警时闪烁专用接口的红蓝警灯 "AlarmServer": 上报到报警中心 "AlarmPush": 推送给 APP "AudioAction": 联动警戒声音 "IOOutputAction": 联动 IO 输出 "draw_rect_enable": 显示检测区域 "draw_human_enable": 显示检测目标 "track_human_enable": 开启人形跟踪 "rect_twinkle_enable": 闪烁检测框 "auto_zoom_enable": 自动变倍跟踪 "gunball_track_mode": 开启枪球跟踪
Status	

2.13.8 设置目标检测配置/HAPI/V1.0/smart/objectdetect/set

URL	**/HAPI/V1.0/smart/objectdetect/set**
Description	**设置目标检测配置**

Method	**PUT**

Sample	PUT /HAPI/V1.0/Smart/objectdetect/set HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 294
	
	{
	"username": "admin",
	"password": "e10adc3949ba59abbe56e057f20f883e",
	"ArmingFlag": {
	"ArmingMode": 2,
	"WeekPlan": {
	"day0": 16777211,
	"day1": 16777212,
	"day2": 16777213,
	"day3": 16777214,
	"day4": 16777215,
	"day5": 16777216,
	"day6": 16777217
	}
	}
	}

URL params(简单参数) **Input Data(PUT)**	参考 2.13.7 获取的 JSON 数据，填充以下参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据（ArmingFlag/ Polygon/以及 AlarmAction 节点下的 LightAction/BRAlarmAction/AlarmServer/AlarmPush/AudioAction/IOOutputAction 等）时，需要完整填充所有必须成员参数。 username password enable type ArmingFlag Polygon Sensitivity AlarmThreshold minTargetRate nonMotionFilter AlarmAction
Success Response Data	参见 2.13.7 GET 返回数据。
Note	设置参数成功的情况下，返回当前配置。

Status	

2.13.9 获取视频遮挡配置/HAPI/V1.0/smart/videocover/get

URL	**/HAPI/V1.0/smart/videocover/get**
Description	获取视频遮挡配置
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/videocover/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/Smart/videocover/get",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": {
	            "enable": 1,
	            "Sensitivity": 20,
	            "Threadhold_second": 5,
	            "BackgroundUpdateSecond": 40,
	            "ArmingFlag": {
	                "ArmingMode": 1,
	                "WeekPlan": {
	                    "day0": 16777215,
	                    "day1": 16777215,
	                    "day2": 16777215,
	                    "day3": 16777215,
	                    "day4": 16777215,
	                    "day5": 16777215,
	                    "day6": 16777215
	                }
	            },
	            "AlarmAction": {
	                "AudioAction": {
	                    "times": 1,
	                    "intervalsecnods": 10,
	                    "filename": "/opt/ch/attention_en.mp3",
	                    "ArmingSetting": {
	                        "ArmingMode": 1,
	                        "TimeSpanNum": 1,

```
                        "DayPlan": [
                            {
                                "startTime": "00:00:00",
                                "endTime": "23:59:59"
                            }
                        ]
                    }
                },
                "IOOutputAction": {
                    "channelCnt": 4,
                    "channels": [
                        {
                            "portIndex": 1,
                            "enable": 1
                        },
                        {
                            "portIndex": 2,
                            "enable": 0
                        },
                        {
                            "portIndex": 3,
                            "enable": 0
                        },
                        {
                            "portIndex": 4,
                            "enable": 0
                        }
                    ]
                }
            }
        }
    }
}
```

Note	设置参数成功的情况下，返回当前配置
Status	

2.13.10 设置视频遮挡配置/HAPI/V1.0/smart/videocover/set

URL	**/HAPI/V1.0/smart/videocover/set**
Description	**设置视频遮挡配置**
Method	**PUT**

Sample	`PUT /HAPI/V1.0/Smart/videocover/set HTTP/1.1`
	`User-Agent: PostmanRuntime-ApipostRuntime/1.1.0`

	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 91
	{
	"username": "admin",
	"password": "e10adc3949ba59abbe56e057f20f883e",
	"enable": 1
	}

URL params(简单参数) Input Data(PUT)	参考 2.13.9 获取的 JSON 数据，填充以下参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。 username password enable Sensitivity Threadhold_second BackgroundUpdateSecond ArmingFlag Sensitivity AlarmAction
Success Response Data	参见 2.13.9 GET 返回的数据
Note	设置参数成功的情况下，返回当前配置
Status	

2.13.11 获取人脸检测配置/HAPI/V1.0/smart/facedetect/get

URL	**/HAPI/V1.0/smart/facedetect/get**
Description	**获取人脸检测配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/facedetect/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**username/password 或者 uid**

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/Smart/facedetect/get",
	        "SessionID": "",
	        "ResponseCode": 0,

```
    "ResponseString": "Succeed",
    "Data": {
        "enable": 1,
        "ArmingFlag": {
            "ArmingMode": 1,
            "WeekPlan": {
                "day0": 16777215,
                "day1": 16777215,
                "day2": 16777215,
                "day3": 16777215,
                "day4": 16777215,
                "day5": 16777215,
                "day6": 16777215
            }
        },
        "Area": {
            "x": 0,
            "y": 0,
            "w": 50,
            "h": 90
        },
        "Sensitivity": 0,
        "AlarmThreshold": 75,
        "AlarmAction": {
            "draw_rect_enable": 1,
            "AudioAction": {
                "times": 1,
                "intervalsecnods": 0,
                "filename": "/opt/ch/attention.mp3",
                "ArmingSetting": {
                    "ArmingMode": 1,
                    "TimeSpanNum": 1,
                    "DayPlan": [
                        {
                            "startTime": "00:00:00",
                            "endTime": "23:59:59"
                        }
                    ]
                }
            },
            "IOOutputAction": {
                "channelCnt": 4,
                "channels": [
                    {
```

	"portIndex": 1,
	"enable": 0
	},
	{
	"portIndex": 2,
	"enable": 0
	},
	{
	"portIndex": 3,
	"enable": 0
	},
	{
	"portIndex": 4,
	"enable": 0
	}
	]
	}
	}
	}
	}
	}

Note	"enable": 算法启用 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Area": 检测矩形区域 "Sensitivity": 灵敏度，取值范围 0-100 "AlarmThreshold": 检测阈值 "AudioAction": 联动警戒声音 "IOOutputAction": 联动 IO 输出 "draw_rect_enable": 显示检测区域
Status	

2.13.12 设置人脸检测配置/HAPI/V1.0/smart/facedetect/set

URL	**/HAPI/V1.0/smart/facedetect/set**
Description	**设置人脸检测配置**
Method	**PUT**
Sample	PUT /HAPI/V1.0/Smart/facedetect/set HTTP/1.1 User-Agent: PostmanRuntime-ApipostRuntime/1.1.0 Cache-Control: no-cache Accept: */* Accept-Encoding: gzip, deflate, br Connection: keep-alive

```
Content-Type: application/x-www-form-urlencoded
Host: 192.168.1.202
Content-Length: 1043

{
    "username": "admin",
    "password": "e10adc3949ba59abbe56e057f20f883e",
    "enable": 1,
    "ArmingFlag": {
        "ArmingMode": 1,
        "WeekPlan": {
            "day0": 16777215,
            "day1": 16777215,
            "day2": 16777215,
            "day3": 16777215,
            "day4": 16777215,
            "day5": 16777215,
            "day6": 16777215
        }
    },
    "Area": {
        "x": 0,
        "y": 0,
        "w": 50,
        "h": 90
    },
    "Sensitivity": 0,
    "AlarmThreshold": 75,
    "AlarmAction": {
        "draw_rect_enable": 1,
        "AudioAction": {
            "times": 1,
            "intervalsecnods": 0,
            "filename": "/opt/ch/attention.mp3",
            "ArmingSetting": {
                "ArmingMode": 1,
                "TimeSpanNum": 1,
                "DayPlan": [
                    {
                        "startTime": "00:00:00",
                        "endTime": "23:59:59"
                    }
                ]
            }
```

	},
	 "IOOutputAction": {
	 "channelCnt": 4,
	 "channels": [
	 {
	 "portIndex": 1,
	 "enable": 0
	 },
	 {
	 "portIndex": 2,
	 "enable": 0
	 },
	 {
	 "portIndex": 3,
	 "enable": 0
	 },
	 {
	 "portIndex": 4,
	 "enable": 0
	 }
	]
	 }
	 }
	 }

URL params(简单参数) Input Data(PUT)	参考 2.13.11 获取的 JSON 数据，填充参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。
Success Response Data	参考 2.13.11 获取的 JSON 数据
Note	设置参数成功的情况下，返回当前配置
Status	

2.13.13 获取越界检测配置/HAPI/V1.0/smart/videogate/get

URL	**/HAPI/V1.0/smart/videogate/get**
Description	**获取越界检测配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/videogate/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) Input Data(PUT)	**username/password 或者 uid**

Success Response Data	{
	 "Response": {

```
    "ResponseURL": "/HAPI/V1.0/Smart/videogate/get",
    "SessionID": "",
    "ResponseCode": 0,
    "ResponseString": "Succeed",
    "Data": {
        "enable": 1,
        "ArmingFlag": {
            "ArmingMode": 1,
            "WeekPlan": {
                "day0": 16777215,
                "day1": 16777215,
                "day2": 16777215,
                "day3": 16777215,
                "day4": 16777215,
                "day5": 16777215,
                "day6": 16777215
            }
        },
        "Rules": [
            {
                "enable": 1,
                "Sensitivity": 80,
                "type": {
                    "CAR": 1,
                    "MOTO": 0,
                    "ELECTRICBICYCLE": 0,
                    "BICYCLE": 0,
                    "HUMAN": 0,
                    "FACE": 0,
                    "NONMOTO_VEHICLE": 0,
                    "FIRE": 0,
                    "FALLINGOBJECT": 0
                },
                "direction": 2,
                "x0": 0,
                "y0": 0,
                "x1": 0,
                "y1": 100
            },
            {
                "enable": 1,
                "Sensitivity": 0,
                "type": {
                    "CAR": 0,
                "MOTO": 0,
                "ELECTRICBICYCLE": 0,
                "BICYCLE": 0,
                "HUMAN": 0,
                "FACE": 0,
                "NONMOTO_VEHICLE": 0,
                "FIRE": 0,
                "FALLINGOBJECT": 0
            },
            "direction": 0,
            "x0": 0,
            "y0": 0,
            "x1": 0,
            "y1": 0
        },
        {
            "enable": 0,
            "Sensitivity": 0,
            "type": {
                "CAR": 0,
                "MOTO": 0,
                "ELECTRICBICYCLE": 0,
                "BICYCLE": 0,
                "HUMAN": 0,
                "FACE": 0,
                "NONMOTO_VEHICLE": 0,
                "FIRE": 0,
                "FALLINGOBJECT": 0
            },
            "direction": 0,
            "x0": 0,
            "y0": 0,
            "x1": 0,
            "y1": 0
        },
        {
            "enable": 0,
            "Sensitivity": 0,
            "type": {
                "CAR": 0,
                "MOTO": 0,
                "ELECTRICBICYCLE": 0,
                "BICYCLE": 0,
                "HUMAN": 0,
                    "FACE": 0,
                    "NONMOTO_VEHICLE": 0,
                    "FIRE": 0,
                    "FALLINGOBJECT": 0
                },
                "direction": 0,
                "x0": 0,
                "y0": 0,
                "x1": 0,
                "y1": 0
            }
        ],
        "AlarmAction": {
            "LightAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "BRAlarmAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmServer": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmPush": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AudioAction": {
                "times": 0,
                "intervalsecnods": 0,
                "filename": "",
                "ArmingSetting": {
                    "ArmingMode": 0,
                    "TimeSpanNum": 1,
                    "DayPlan": [
                        {
                            "startTime": "00:00:00",
                            "endTime": "23:59:59"
                            }
                        ]
                    }
                },
                "IOOutputAction": {
                    "channelCnt": 4,
                    "channels": [
                        {
                            "portIndex": 1,
                            "enable": 0
                        },
                        {
                            "portIndex": 2,
                            "enable": 0
                        },
                        {
                            "portIndex": 3,
                            "enable": 0
                        },
                        {
                            "portIndex": 4,
                            "enable": 0
                        }
                    ]
                },
                "draw_rect_enable": 0,
                "draw_target_enable": 0
            }
        }
    }
}
```

Note	"enable": 算法启用 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Rules": 配置检测规则，最多可以配置 4 条，参见 4.10 "type": 目标检测类型，参见 4.7 "LightAction": 补光灯联动，有报警时闪烁补光灯 "BRAlarmAction": 红蓝警灯联动，有报警时闪烁专用接口的红蓝警灯 "AlarmServer": 上报到报警中心 "AlarmPush": 推送给 APP "AudioAction": 联动警戒声音 "IOOutputAction": 联动 IO 输出 "draw_rect_enable": 显示检测区域 "draw_target_enable": 显示检测目标

Status	

2.13.14 设置越界检测配置/HAPI/V1.0/smart/videogate/set

URL	**/HAPI/V1.0/smart/videogate/set**
Description	**设置越界检测配置**
Method	**PUT**
Sample	

```
PUT /HAPI/V1.0/Smart/videogate/set HTTP/1.1
User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
Cache-Control: no-cache
Accept: */*
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Host: 192.168.1.202
Content-Length: 476

{
    "username": "admin",
    "password": "e10adc3949ba59abbe56e057f20f883e",
    "enable": 1,
    "ArmingFlag": {
        "ArmingMode": 1
    },
    "Rules": [
        {
            "enable": 1,
            "Sensitivity": 80,
            "type": {
                "CAR": 1,
                "HUMAN": 1
            },
            "direction": 2,
            "x0": 0,
            "y0": 0,
            "x1": 100,
            "y1": 100
        }
    ]
}
```

URL params(简单参数) **Input Data(PUT)**	参考 2.13.13 获取的 JSON 数据，填充参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。
Success Response Data	参考 2.13.13 获取的 JSON 数据

Note	设置参数成功的情况下，返回当前配置
Status	

2.13.15 获取区域(周界)侦测配置/HAPI/V1.0/smart/regionai/get

URL	**/HAPI/V1.0/smart/videogate/get**
Description	**获取区域(周界)侦测配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/regionai/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**
Success Response Data	

```
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/Smart/regionai/get",
        "SessionID": "",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": {
            "enable": 1,
            "ArmingFlag": {
                "ArmingMode": 1,
                "WeekPlan": {
                    "day0": 16777215,
                    "day1": 16777215,
                    "day2": 16777215,
                    "day3": 16777215,
                    "day4": 16777215,
                    "day5": 16777215,
                    "day6": 16777215
                }
            },
            "Polygon": {
                "PointCnt": 4,
                "Points": [
                    {
                        "x": 0,
                        "y": 0
                    },
                    {
                        "x": 100,
                        "y": 0
                    },
                    {
                        "x": 100,
                        "y": 100
                    },
                    {
                        "x": 0,
                        "y": 100
                    }
                ]
            },
            "Rules": [
                {
                    "enable": 1,
                    "Sensitivity": 80,
                    "type": {
                        "CAR": 1,
                        "MOTO": 0,
                        "ELECTRICBICYCLE": 0,
                        "BICYCLE": 0,
                        "HUMAN": 1,
                        "FACE": 0,
                        "NONMOTO_VEHICLE": 0,
                        "FIRE": 0,
                        "FALLINGOBJECT": 0
                    },
                    "mode": "LEAVE",
                    "stayseconds": 5
                },
                {
                    "enable": 0,
                    "Sensitivity": 0,
                    "type": {
                        "CAR": 0,
                        "MOTO": 0,
                        "ELECTRICBICYCLE": 0,
                        "BICYCLE": 0,
                        "HUMAN": 0,
                        "FACE": 0,
                        "NONMOTO_VEHICLE": 0,
                        "FIRE": 0,
                        "FALLINGOBJECT": 0
                    },
                    "mode": "STAY",
                    "stayseconds": 0
            },
            {
                "enable": 0,
                "Sensitivity": 0,
                "type": {
                    "CAR": 0,
                    "MOTO": 0,
                    "ELECTRICBICYCLE": 0,
                    "BICYCLE": 0,
                    "HUMAN": 0,
                    "FACE": 0,
                    "NONMOTO_VEHICLE": 0,
                    "FIRE": 0,
                    "FALLINGOBJECT": 0
                },
                "mode": "STAY",
                "stayseconds": 0
            }
        ],
        "AlarmAction": {
            "LightAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "BRAlarmAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmServer": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmPush": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AudioAction": {
                "times": 0,
                "intervalsecnods": 0,
                "filename": "",
                    "ArmingSetting": {
                        "ArmingMode": 0,
                        "TimeSpanNum": 1,
                        "DayPlan": [
                            {
                                "startTime": "00:00:00",
                                "endTime": "23:59:59"
                            }
                        ]
                    }
                },
                "IOOutputAction": {
                    "channelCnt": 4,
                    "channels": [
                        {
                            "portIndex": 1,
                            "enable": 0
                        },
                        {
                            "portIndex": 2,
                            "enable": 0
                        },
                        {
                            "portIndex": 3,
                            "enable": 0
                        },
                        {
                            "portIndex": 4,
                            "enable": 0
                        }
                    ]
                },
                "draw_rect_enable": 0,
                "draw_target_enable": 0
            }
        }
    }
}
```

Note	"enable": 算法启用 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Polygon": 检测区域 "Rules": 配置检测规则，最多可以配置 3 条，参见 4.11 "type": 目标检测类型，参见 4.7 "LightAction": 补光灯联动，有报警时闪烁补光灯

	"BRAlarmAction": 红蓝警灯联动，有报警时闪烁专用接口的红蓝警灯 "AlarmServer": 上报到报警中心 "AlarmPush": 推送给 APP "AudioAction": 联动警戒声音 "IOOutputAction": 联动 IO 输出 "draw_rect_enable": 显示检测区域 "draw_target_enable": 显示检测目标
Status	

2.13.16 设置区域(周界)侦测配置/HAPI/V1.0/smart/regionai/set

URL	**/HAPI/V1.0/smart/regionai/set**
Description	**设置区域(周界)侦测配置**
Method	**PUT**

Sample

```
PUT /HAPI/V1.0/Smart/regionai/set HTTP/1.1

User-Agent: PostmanRuntime-ApipostRuntime/1.1.0

Cache-Control: no-cache

Accept: */*

Accept-Encoding: gzip, deflate, br

Connection: keep-alive

Content-Type: application/x-www-form-urlencoded

Host: 192.168.1.202

Content-Length: 659


{

      "username": "admin",

      "password": "e10adc3949ba59abbe56e057f20f883e",

      "enable": 1,

      "ArmingFlag": {

            "ArmingMode": 1

            },

      "Polygon": {

            "PointCnt": 4,

            "Points": [

                  {

                        "x": 0,

                        "y": 0

                  },

                  {

                        "x": 100,

                        "y": 0

                  },

                  {
```

```
                "x": 100,
                "y": 100
            },
            {
                "x": 0,
                "y": 100
            }
        ]
    },
    "Rules": [
        {
            "enable": 1,
            "Sensitivity": 80,
            "type": {
                "CAR": 1,
                "MOTO": 0,
                "ELECTRICBICYCLE": 0,
                "BICYCLE": 0,
                "HUMAN": 1,
                "FACE": 0,
                "NONMOTO_VEHICLE": 0,
                "FIRE": 0,
                "FALLINGOBJECT": 0
            },
            "mode": "LEAVE",
            "stayseconds": 5
        }
    ]
}
```

URL params(简单参数) Input Data(PUT)	参考 2.13.15 获取的 JSON 数据，填充参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。
Success Response Data	参考 2.13.15 获取的 JSON 数据
Note	设置参数成功的情况下，返回当前配置
Status	

2.13.17 获取车牌识别配置/HAPI/V1.0/smart/lpr/get

URL	**/HAPI/V1.0/smart/lpr/get**
Description	**获取车牌识别配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/lpr/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e

URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**
Success Response Data	

```
{
    "Response": {
        "ResponseURL": "/HAPI/V1.0/Smart/lpr/get",
        "SessionID": "",
        "ResponseCode": 0,
        "ResponseString": "Succeed",
        "Data": {
            "enable": 1,
            "Sensitivity": 80,
            "detectionmode": 0,
            "actionInterval": 5,
            "snapQuality": "Default",
            "ArmingFlag": {
                "ArmingMode": 1,
                "WeekPlan": {
                    "day0": 16777215,
                    "day1": 16777215,
                    "day2": 16777215,
                    "day3": 16777215,
                    "day4": 16777215,
                    "day5": 16777215,
                    "day6": 16777215
                }
            },
            "Polygon": {
                "PointCnt": 4,
                "Points": [
                    {
                        "x": 0,
                        "y": 0
                    },
                    {
                        "x": 100,
                        "y": 0
                    },
                    {
                        "x": 100,
                        "y": 100
                    },
                    {
                        "x": 0,
                        "y": 100
                    }
                ]
            },
            "AlarmAction": {
                "AlarmServer": {
                    "ArmingMode": 1,
                    "TimeSpanNum": 0,
                    "DayPlan": []
                },
                "AlarmPush": {
                    "ArmingMode": 1,
                    "TimeSpanNum": 0,
                    "DayPlan": []
                },
                "draw_rect_enable": 1,
                "draw_target_enable": 1,
                "draw_osd_enable": 1,
                "play_voice_enable": 1
            }
        }
    }
}
```

Note	"enable": 算法启用，控制算法的检测，用于实现摄像机的软光敏开关灯等逻辑，以及将报警通知给其他联动模块 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Polygon": 检测区域 "Sensitivity": 灵敏度，取值范围 0-100 "detectionmode": 推图模式，0:最优推图 1:间隔推图 "actionInterval":相同车牌的情况下两次上报时间间隔，单位秒 "snapQuality ": 抓图质量(分辨率)，Default,1280X720,后续根据需要增加其他分辨率支持 "draw_rect_enable": 绘制检测区域 "draw_target_enable": 绘制检测目标 "draw_osd_enable": 绘制识别到的车牌 "play_voice_enable": 语音播报车牌
Status	

2.13.18 设置车牌识别配置/HAPI/V1.0/smart/lpr/set

URL	**/HAPI/V1.0/smart/lpr/set**
Description	**设置车牌识别配置**
Method	**PUT**
Sample	PUT /HAPI/V1.0/Smart/lpr/set HTTP/1.1 User-Agent: PostmanRuntime-ApipostRuntime/1.1.0

```
Cache-Control: no-cache
Accept: */*
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Host: 192.168.1.202
Content-Length: 663

{
    "username": "admin",
    "password": "e10adc3949ba59abbe56e057f20f883e",
    "enable": 1,
    "Sensitivity": 80,
    "detectionmode": 0,
    "actionInterval": 5,
    "snapQuality": "Default",
    "ArmingFlag": {
        "ArmingMode": 1
    },
    "Polygon": {
        "PointCnt": 4,
        "Points": [
            {
                "x": 0,
                "y": 0
            },
            {
                "x": 100,
                "y": 0
            },
            {
                "x": 100,
                "y": 100
            },
            {
                "x": 0,
                "y": 100
            }
        ]
    },
    "AlarmAction": {
        "AlarmServer": {
            "ArmingMode": 1
        },
```

	"AlarmPush": {
	"ArmingMode": 1
	},
	"draw_rect_enable": 1,
	"draw_target_enable": 1,
	"draw_osd_enable": 1,
	"play_voice_enable": 1
	}
	}

URL params(简单参数) **Input Data(PUT)**	参考 2.13.17 获取的 JSON 数据，填充参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。
Success Response Data	参考 2.13.17 获取的 JSON 数据
Note	设置参数成功的情况下，返回当前配置。
Status	

2.13.19 获取烟火检测配置/HAPI/V1.0/smart/flameflumes/get

URL	**/HAPI/V1.0/smart/flameflumes/get**
Description	**获取烟火配置**
Method	**GET/PUT**
Sample	http://192.168.1.202/HAPI/V1.0/Smart/flameflumes/get?username=admin&password=e10adc3949ba59abbe56e057f20f883e
URL params(简单参数) **Input Data(PUT)**	**username/password 或者 uid**

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/Smart/flameflumes/get",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"enable": 1,
	"Sensitivity": 60,
	"actionInterval": 5,
	"snapQuality": "1280X720",
	"ArmingFlag": {
	"ArmingMode": 0,
	"WeekPlan": {
	"day0": 16777215,
	"day1": 16777215,
	"day2": 16777215,

```
                "day3": 16777215,
                "day4": 16777215,
                "day5": 16777215,
                "day6": 16777215
            }
        },
        "Polygon": {
            "PointCnt": 4,
            "Points": [
                {
                    "x": 0,
                    "y": 0
                },
                {
                    "x": 100,
                    "y": 0
                },
                {
                    "x": 100,
                    "y": 100
                },
                {
                    "x": 0,
                    "y": 100
                }
            ]
        },
        "AlarmAction": {
            "LightAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "BRAlarmAction": {
                "ArmingMode": 0,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmServer": {
                "ArmingMode": 1,
                "TimeSpanNum": 0,
                "DayPlan": []
            },
            "AlarmPush": {
```

```
            "ArmingMode": 0,
            "TimeSpanNum": 0,
            "DayPlan": []
        },
        "AudioAction": {
            "times": 1,
            "intervalsecnods": 0,
            "filename": "/opt/ch/Welcome.mp3",
            "ArmingSetting": {
                "ArmingMode": 0,
                "TimeSpanNum": 1,
                "DayPlan": [
                    {
                        "startTime": "00:00:00",
                        "endTime": "23:59:59"
                    }
                ]
            }
        },
        "IOOutputAction": {
            "channelCnt": 4,
            "channels": [
                {
                    "portIndex": 1,
                    "enable": 0
                },
                {
                    "portIndex": 2,
                    "enable": 0
                },
                {
                    "portIndex": 3,
                    "enable": 0
                },
                {
                    "portIndex": 4,
                    "enable": 0
                }
            ]
        },
        "draw_rect_enable": 1,
        "draw_target_enable": 1
    }
}
```

	} }
Note	"enable": 算法启用，控制算法的检测，用于实现摄像机的软光敏开关灯等逻辑，以及将报警通知给其他联动模块 "ArmingFlag": 算法布防总开关，可配置 7X24 周计划 "Polygon": 检测区域 "Sensitivity": 灵敏度，取值范围 0-100 "actionInterval":相同车牌的情况下两次上报时间间隔，单位秒 "snapQuality ": 抓图质量(分辨率)，Default,1280X720,后续根据需要增加其他分辨率支持 "draw_rect_enable": 绘制检测区域 "draw_target_enable": 绘制检测目标 "LightAction": 补光灯联动，有报警时闪烁补光灯 "BRAlarmAction": 红蓝警灯联动，有报警时闪烁专用接口的红蓝警灯 "AlarmServer": 上报到报警中心 "AlarmPush": 推送给 APP "AudioAction": 联动警戒声音 "IOOutputAction": 联动 IO 输出
Status	

2.13.20 设置烟火检测配置/HAPI/V1.0/smart/flameflumes/set

URL	**/HAPI/V1.0/smart/flameflumes/set**
Description	**设置烟火检测配置**
Method	**PUT**

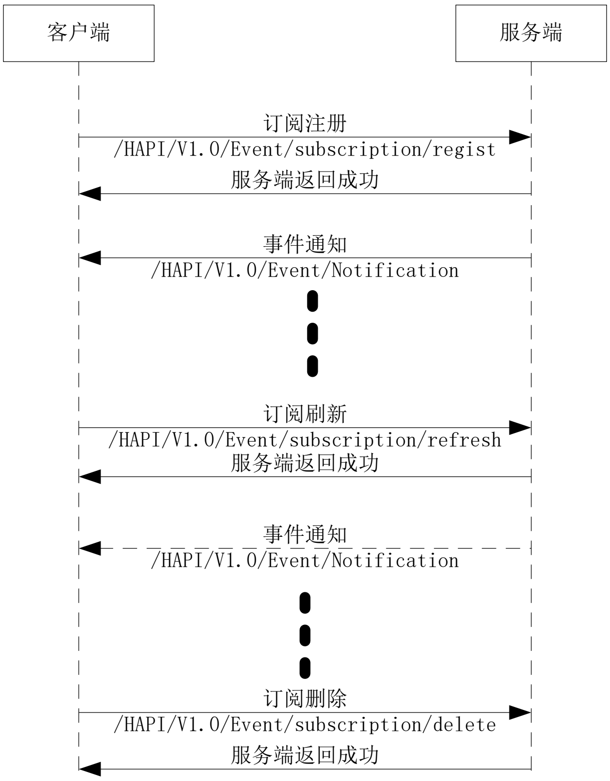
Sample	PUT /HAPI/V1.0/Smart/flameflumes/set HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 452
	
	{
	"username": "admin",
	"password": "e10adc3949ba59abbe56e057f20f883e",
	"enable": 1,
	"Sensitivity": 60,
	"actionInterval": 5,
	"snapQuality": "1280X720",
	"ArmingFlag": {
	"ArmingMode": 0

	},
	"AlarmAction": {
	"AlarmServer": {
	"ArmingMode": 1
	},
	"AlarmPush": {
	"ArmingMode": 0
	},
	"AudioAction": {
	"ArmingSetting": {
	"ArmingMode": 0
	}
	},
	"draw_rect_enable": 1,
	"draw_target_enable": 1
	}
	}

URL params(简单参数) Input Data(PUT)	参考 2.13.19 获取的 JSON 数据，填充参数中的一个或者多个参数，缺少必要参数或者范围不正确将返回错误。当参数是某项单独配置的 JSON 结构化数据时，需要完整填充所有必须成员参数。
Success Response Data	参考 2.13.19 获取的 JSON 数据
Note	设置参数成功的情况下，返回当前配置。
Status	

3 事件

3.1 事件工作流程


客户端
服务端
订阅注册
/HAPI/V1.0/Event/subscription/regist
服务端返回成功
事件通知
/HAPI/V1.0/Event/Notification
订阅刷新
/HAPI/V1.0/Event/subscription/refresh
服务端返回成功
事件通知
/HAPI/V1.0/Event/Notification
订阅删除
/HAPI/V1.0/Event/subscription/delete
服务端返回成功

流程说明

告警上报

步骤 1：客户端开启 TCP 服务用于接受服务端事件推送

步骤 2：客户端向设备侧（服务端）订阅告警，接口详情参见注册订阅 /HAPI/V1.0/Event/subscription/regist

步骤 3：设备侧（服务端）返回的参数分别代表订阅标识（ID）、订阅开始时间（CurrentTime）及结束时间（TerminationTime）等订阅信息，订阅标识（ID）用于后续告警的刷新或取消。

步骤 4：客户端需在订阅结束时间（TerminationTime）之前定期发送刷新接口进行刷新，刷新订阅。

步骤 5：设备侧（服务端）返回刷新的订阅时间。

步骤 6：如果设备侧（服务端）发生告警，设备侧通过 TCP 短连接向客户端指定的地址推送事件，详见事件通知。事件通知遵循 HTTP 协议，通过 JSON 携带事件信息。

告警取消

步骤 1：客户端取消告警，接口详情参见删除订阅/HAPI/V1.0/Event/subscription/delete

步骤 2：服务端取消成功，不再推送告警。

3.2 事件订阅

3.2.1 注册订阅/HAPI/V1.0/Event/subscription/regist

URL	/HAPI/V1.0/Event/subscription/regist
Description	注册订阅
Method	**POST**

Sample	POST
	/HAPI/V1.0/Event/subscription/regist?username=admin&password=e10adc3949ba59abbe56e057f20f883e
	HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 107
	
	{
	"ServerType":0,
	"ServerName": "192.168.1.253",
	"Port": 9998,
	"Duration": 3600,
	"EventType": "all"
	}

URL params(简单参数)
Input Data(POST)
	{
	    "ServerType": 0,
	    "ServerName": "192.168.1.253",
	    "Port": 9998,
	    "Duration": 3600,
	    "PostURLPrefix":"",
	    "EventType": "all"
	}

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/Event/subscription/regist",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",

```
        "Data": {
            "ID": 2,
            "ServerType": 9998,
            "ServerName": "192.168.1.253",
            "ServerPort": 9998,
            "EventType": "all",
            "CurrentTime": 1699930618,
            "TerminationTime": 1699934218
        }
    }
}
```

Note	设备上对相同服务器和端口的注册订阅请求，不会创建的新的重复订阅。
Status	

Param	Requirment	Type	Description	Example
ServerType	M	Enum	IP 地址类型： 0: IPv4； 1:域名	0
ServerName	M	string	IPv4 地址或者域名。 长度范围为[0,64]。	
Port	M	unsigned long	端口，范围为[1, 65535]。	
Duration	M	unsigned long	订阅周期，单位为 s，范围为[30, 3600]。	
PostURLPrefix	C	string	设备上报 event 时需要添加的 URL 前缀，用以报警事件服务器指定路径。	0
EventType	C	string	订阅事件类型，不填或者填”all”时意味着需要上报所有事件，不同事件用逗号分割。	"MotionDetect,ObjectDetect”
ID	M	unsigned long	设备返回的订阅标识，用以识别哪次订阅，刷新订阅、删除订阅操作需要携带此 ID，以指明需操作哪次订阅。	0
CurrentTime	M	unsigned long	UTC 时间，从 1970 年 1 月 1 日 0 点开始的秒数。	1477104900
TerminationTime	M	unsigned long	UTC 时间，从 1970 年 1 月 1 日 0 点开始的秒数。	1477104949

3.2.2 刷新订阅/HAPI/V1.0/Event/subscription/refresh

URL	/HAPI/V1.0/Event/subscription/refresh
Description	刷新订阅
Method	**POST**

Sample	POST
	/HAPI/V1.0/Event/subscription/refresh?username=admin&password=e10adc3949ba59abbe56e057f20f883e
	HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 40
	
	{
	"ID": 1,
	"Duration": 3600
	}

URL params(简单参数)
Input Data(POST)
	{
	"ID": 2,
	"Duration": 3600
	}

Success Response Data	{
	"Response": {
	"ResponseURL": "/HAPI/V1.0/Event/subscription/refresh",
	"SessionID": "",
	"ResponseCode": 0,
	"ResponseString": "Succeed",
	"Data": {
	"ID": 2,
	"ServerType": 9998,
	"ServerName": "192.168.1.253",
	"ServerPort": 9998,
	"EventType": "all",
	"CurrentTime": 1699931648,
	"TerminationTime": 1699935248
	}
	}
	}

Note	参数定义参见 3.2.1
Status	

3.2.3 删除订阅/HAPI/V1.0/Event/subscription/delete

URL	/HAPI/V1.0/Event/subscription/delete
Description	**删除订阅**
Method	**POST**

Sample	POST
	/HAPI/V1.0/Event/subscription/delete?username=admin&password=e10adc3949ba59abbe56e057f20f883e
	HTTP/1.1
	User-Agent: PostmanRuntime-ApipostRuntime/1.1.0
	Cache-Control: no-cache
	Accept: */*
	Accept-Encoding: gzip, deflate, br
	Connection: keep-alive
	Content-Type: application/x-www-form-urlencoded
	Host: 192.168.1.202
	Content-Length: 17
	
	{
	    "ID": 2
	}

URL params(简单参数)
Input Data(POST)
	{
	    "ID": 2
	 }

Success Response Data	{
	    "Response": {
	        "ResponseURL": "/HAPI/V1.0/Event/subscription/delete",
	        "SessionID": "",
	        "ResponseCode": 0,
	        "ResponseString": "Succeed",
	        "Data": {
	            "ID": 2
	        }
	    }
	}

Note	参数定义参见 3.2.1
Status	

3.3 事件通知

3.3.1 事件通知/HAPI/V1.0/Event/Notification

URL	/HAPI/V1.0/Event/Notification
Description	推送事件通知。
Method	POST

Input Data	{
	"AlarmType":,
	"AlarmSubType":,
	"TimeStamp":
	"ChannelNo":
	"NotificationType":,
	"DeviceID":,
	"Picture":,
	"AlarmInfo":
	}

Success Return Data	None
Note	
Status	

AlarmType	M	string	告警类型 参见表格 1 AlarmType/AlarmSubType事件类型	"ObjectDetect"
AlarmSubType	M	string	告警类型 参见表格 1 AlarmType/AlarmSubType事件类型	"HumanShapeDetect"
TimeStamp	M	unsigned long	告警时间。 UTC 时间，从 1970 年 1 月 1 日 0 点开始的秒数。	1489040894
Channel	M	unsigned long	告警源通道 ID。 单通道产品固定填 0； 多通道产品 0 表示通道无关的事件；通道事件填通道 ID 号（从 1 开始）。	1
NotificationType	M	unsigned long	通知类型： 0：实时通知 1：历史通知	0
OccurFlag	M	String	报警发生标记： “true”:报警发生	true

			“false”:报警结束	
DeviceID	M	string	告警设备 ID，填写设备序列号，或者其他报警服务器要求的定制标识符	“EF000000000000A3”
Picture	C	Json Block	jpg 图片信息，携带图片 size 和 Data，Data 为对 jpg 进行 base64 编码后的字符串	
AlarmInfo	M	Json Block	根据不同的 AlarmType，携带不同的结构化数据信息，以及图片信息等	-

表格 1 AlarmType/AlarmSubType 事件类型

AlarmType/事件类型	AlarmSubType/事件子类型	事件描述
MotiDetect	MotiAlarm	运动检测告警
ObjectDetect	HumanShapeDetect	人形检测报警
	VehicleDetect	车辆检测告警
	FallingObjectsDetectiAlarm	高空抛物报警
	ElectricbicycleDetect	电单车检测
	HumanShapeDetect	人形检测报警消失
	VehicleDetect	车辆检测告警消失
	FallingObjectsDetectiAlarm	高空抛物报警消失
	ElectricbicycleDetect	电单车检测报警消失
MaskDetect	MaskImageAlarm	遮挡侦测告警
AudioDetect	AbnormalAudio	音频异常检测告警
CrosslineDetect	LineDetectorCrossed	越界告警
RegiDetect	EnterArea	进入区域
	LeaveArea	离开区域
	Loitering	区域徘徊
	LoiteringDetecti	徘徊检测
	FenceCrossing	翻越围栏
	ObjectRemoved	物品看护
	ObjectLeftBehind	物品遗留
	PeopleGathering	人员聚集
	AreaPeopleCountingAlarm	区域人数统计告警
FireDetect	FireAlarm	火焰告警
	FumesAlarm	烟雾告警
	FlameAndFumesAlarm	烟火告警
FaceDetect	FaceIsDetected	人脸检测
	FaceRecogniti	人脸比对报警
	FaceMatchAlarm	人脸识别匹配报警
	FaceRecognitiMatchlistAlarm	人脸识别匹配报警

	FaceNotMatchAlarm	人脸识别不匹配报警
	FaceRecognitiMismatchlistAlarm	人脸识别不匹配报警
LprDetect	LprDetectAlarm	车牌识别报警
	LprMatchlistAlarm	车牌识别匹配报警
	LprMismatchlistAlarm	车牌识别不匹配报警
	LprBlacklistAlarm	停车场车辆识别黑名单报警
IllegalLogin	IllegalLogin	非法访问
BatteryDetect	LowBattery	低电量报警
NetworkDetect	NetworkDiscnected	网络断开异常告警
	IPCflict	IP 冲突异常告警
IOInputDetect	InputAlarm	输入开关量告警
TemperatureDetect	TemperatureDetectiAlarm	温度检测事件
	HighTemperatureHighAlarm	温度过高报警
	TemperatureLowAlarm	温度过低报警
	TemperatureAbnormalAlarm	温度异常报警
HumanStatusDetect	SafetyHelmetAlarm	未佩戴安全帽子报警
	TelephingAlarm	打电话报警
	SmokingAlarm	吸烟告警
	BodyTemperatureAlarm	体温异常告警
	NoMaskAlarm	未戴口罩告警

移动侦测报警发生消息示例：

```
POST /HAPI/V1.0/Event/Notification HTTP/1.1
User-Agent: ANJVISION HAPI NOTIFY
Cache-Control: no-cache
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
x-cos-meta-User-ID: 1
Host: 192.168.1.253
Content-Length: 983
```

{"AlarmType":"MotionDetect","AlarmSubType":"MotionAlarm","TimeStamp":1705310501,"ChannelNo":0,"NotificationType":0,"OccurFlag":"true","DeviceID":"EF000000000000A3","AlarmInfo":{"image":null,"MotionDetectInfo":{"motion_col":22,"motion_row":18,"motion_details":[{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000100000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0100000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"},{"motion_row":"0000000000000000000000"}]}}}

移动侦测报警结束消息示例：

```
POST /HAPI/V1.0/Event/Notification HTTP/1.1
User-Agent: ANJVISION HAPI NOTIFY
Cache-Control: no-cache
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
x-cos-meta-User-ID: 1
Host: 192.168.1.253
Content-Length: 182

{"AlarmType":"MotionDetect","AlarmSubType":"MotionAlarm","TimeStamp":1705310512,"ChannelNo":0,"NotificationType":0,"OccurFlag":"false","DeviceID":"EF000000000000A3","AlarmInfo":null}
```

4 JSON 结构化数据说明

4.1 布防方式（ArmingMode）

布防方式可以配置为：禁用、全天启用、白天启用、夜晚启用、自定义时间段。定义如下：

```
typedef enum
{
     ARMING_DISABLE = 0,//不布防
     ARMING_ALLDAY = 1,//全天布防
     ARMING_DAYTIME = 2,//白天布防
     ARMING_NIGHT = 3,//夜晚布防
     ARMING_CUSTOM = 4,//自定义时间段
}ArmingMode;
```

当布防方式为自定义时间段时，报警检测布防配置根据周计划来触发各种报警联动；具体某种报警联动布防配置根据日计划来触发联动。

4.2 报警检测布防配置

报警检测布防配置，区别于报警检测启用开关，主要用于控制所有的报警联动输出。也就是说，报警检测启用的情况下，可以按照布防计划来启用或者统一关闭所有的报警联动输出。在报警检测布防计划时间段内的时候，每种报警联动控制再根据各自独立的布防计划去输出。

报警检测布防方式定义参见 4.1。当配置为自定义时间段时，按照 7X24 小时配置布防周计划表，最小时间单位为 1 小时。JSON 数据表示时，每天用 1 个 32 位整数来表示，每小时占用 1 个 bit 来。

JSON 数据：

```
           "ArmingFlag": {
               "ArmingMode": 1,
    "WeekPlan": {
        "day0": 16777215,
        "day1": 16777215,
        "day2": 16777215,
        "day3": 16777215,
        "day4": 16777215,
        "day5": 16777215,
        "day6": 16777215
    }
}
```

4.3 运动侦测区域配置

运动侦测区域按照最大 18 行 22 列的子区域来配置，每个子区域用字符’0’和’1’来表示启用该区域的运动侦测。

JSON 数据中，区域配置使用 JSON 字符串数组来表示。

```
"AreaConfig": {
    "block_col": 22,
    "block_row": 18,
    "blockconfig": [
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "0001111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
        },
        {
            "block_row": "1111111111111111111111"
```

```
                    },
                    {
                        "block_row": "1111111111110000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    },
                    {
                        "block_row": "0000000000000000000000"
                    }
                ]
            },
```

4.4 报警联动布防配置

报警联动布防方式定义参见 4.1，用于配置联动日计划，当配置为自定义时间段时，配置一日内最多 4 个时间段

JSON 数据：

```
{
    "ArmingMode": 4,
    "TimeSpanNum": 1,
    "DayPlan": [
        {
            "startTime": "00:00:00",
            "endTime": "23:59:59"
        }
    ]
}
```

4.5 警戒音联动配置

警戒音联动用于配置报警触发时设备播放的提示音，以及警戒音联动布防计划。

JSON 数据：

```
"AudioAction": {
    "times": 1,
    "intervalsecnods": 10,
    "filename": "/opt/ch/attention_en.mp3",
    "ArmingSetting": {
        "ArmingMode": 0,
        "TimeSpanNum": 1,
        "DayPlan": [
            {
                "startTime": "00:00:00",
                "endTime": "23:59:59"
            }
        ]
    }
}
```

4.6 IO 输出联动配置

IO 输出联动配置用于配置报警触发时联动控制的 IO 输出列表。IO 输出的布防计划，由 IO 输出配置来决定。

JSON 数据：

```
"IOOutputAction": {
    "channelCnt": 4,
    "channels": [
        {
            "portIndex": 1,
            "enable": 0
        },
        {
            "portIndex": 2,
            "enable": 0
        },
        {
            "portIndex": 3,
            "enable": 0
        },
        {
            "portIndex": 4,
            "enable": 0
        }
    ]
}
```

4.7 目标检测类型

目标检测类型用于配置目标检测算法中的目标类型，目前主要使用人形和车型。

JSON 数据：

```
"type": {
      "CAR": 0,
      "MOTO": 0,
      "ELECTRICBICYCLE": 0,
      "BICYCLE": 0,
      "HUMAN": 1,
      "FACE": 0,
      "NONMOTO_VEHICLE": 0,
      "FIRE": 0,
      "FALLINGOBJECT": 0
}
```

4.8 目标检测区域配置

目标检测区域配置用于配置目标检测的区域范围，支持最多 10 个顶点的不规则多边形区域。每个顶点的使用屏幕百分比 x/y 坐标来表示，不能有线段交叉。

JSON 数据：

```
"Polygon": {
      "PointCnt": 7,
      "Points": [
            {
                  "x": 17,
                  "y": 11
            },
            {
                  "x": 86,
                  "y": 13
            },
            {
                  "x": 95,
                  "y": 83
            },
            {
                  "x": 62,
                  "y": 94
            },
            {
                  "x": 16,
                  "y": 90
            },
            {
                  "x": 5,
                  "y": 37
            },
            {
                  "x": 8,
                  "y": 14
            }
      ]
}
```

4.9 矩形检测区域配置

矩形检测区域配置用于配置人脸检测等区域范围，配置左上顶点位置以及矩形宽、高，使用屏幕百分比 x/y 坐标来表示。

JSON 数据：

```
"Area": {
      "x": 0,
      "y": 0,
      "w": 50,
      "h": 90
}
```

4.10 越界检测规则配置

越界检测最多配置4条规则。type参见4.7.。每条规则配置线条的两个端点（AB）坐标位置，使用屏幕百分比x/y坐标来表示。direction 取值范围：0: A<->B 1: A->B 2: A<-B (SIDE A MEANS LEFT OF THE LINE)

JSON数据：

```
"Rules": [
      {
            "enable": 1,
            "Sensitivity": 80,
            "type": 1,
            "direction": 2,
            "x0": 50,
            "y0": 0,
            "x1": 50,
            "y1": 100
      },
      {
            "enable": 1,
            "Sensitivity": 50,
            "type": 1,
            "direction": 0,
            "x0": 30,
            "y0": 10,
            "x1": 70,
            "y1": 90
      },
      {
            "enable": 0,
            "Sensitivity": 0,
            "type": 0,
            "direction": 0,
            "x0": 0,
            "y0": 0,
            "x1": 0,
            "y1": 0
      },
      {
            "enable": 0,
            "Sensitivity": 0,
            "type": 0,
            "direction": 0,
            "x0": 0,
            "y0": 0,
            "x1": 0,
        "y1": 0
    }
],
```

4.11 区域侦测规则配置

越界检测最多配置3条规则。type参见4.7.。每条规则检测类型、检测模式（STAY:停留，ENTER:进入，LEAVE：离开）。模式为STAY的时候，需要配置stayseconds（报警等停留时间）。

JSON数据：

```
"Rules": [
    {
        "enable": 1,
        "Sensitivity": 80,
        "type": {
            "CAR": 1,
            "MOTO": 0,
            "ELECTRICBICYCLE": 0,
            "BICYCLE": 0,
            "HUMAN": 1,
            "FACE": 0,
            "NONMOTO_VEHICLE": 0,
            "FIRE": 0,
            "FALLINGOBJECT": 0
        },
        "mode": "STAY",
        "stayseconds": 0
    },
    {
        "enable": 0,
        "Sensitivity": 0,
        "type": {
            "CAR": 0,
            "MOTO": 0,
            "ELECTRICBICYCLE": 0,
            "BICYCLE": 0,
            "HUMAN": 0,
            "FACE": 0,
            "NONMOTO_VEHICLE": 0,
            "FIRE": 0,
            "FALLINGOBJECT": 0
        },
        "mode": "STAY",
        "stayseconds": 0
    },
    {
        "enable": 0,
        "Sensitivity": 0,
        "type": {
            "CAR": 0,
            "MOTO": 0,
            "ELECTRICBICYCLE": 0,
            "BICYCLE": 0,
            "HUMAN": 0,
            "FACE": 0,
            "NONMOTO_VEHICLE": 0,
            "FIRE": 0,
            "FALLINGOBJECT": 0
        },
        "mode": "STAY",
        "stayseconds": 0
    }
],
```

4.12 事件通知 Json Block

4.12.1 MotionDetectInfo

```
{
  "motion_col": 22,
  "motion_row": 18,
  "motion_details": [
      {
          "motion_row": "0000000011111111111111"
      },
      {
          "motion_row": "0000000010111111111111"
      },
      {
          "motion_row": "0000000010111111111111"
      },
      {
          "motion_row": "0000111010011111111111"
      },
      {
          "motion_row": "0111110111111111111111"
      },
      {
          "motion_row": "1110011111111111111111"
      },
      {
          "motion_row": "1111110100111111111111"
      },
      {
          "motion_row": "0111000000011111111111"
      },
      {
          "motion_row": "1111001011110000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      },
      {
          "motion_row": "0000000000000000000000"
      }
  ]
}
```

Param	Requirement	Type	Description	Example
motion_col	**M**	**unsigned**	图像宽度分割列数	**22**

		long		
motion_row	M	unsigned long	图像高度分割行数	18
motion_details	M	array	每一行子区域运动检测结果，用字符’0’表示该区域没有检测到运动，’1’来表示该区域检测到运动	

4.12.1 ObjectDetectInfo

暂空，后续根据算法完善