

# 1 预置点专用聚焦

## 1.1 功能介绍

基于预置点的专用聚焦算法，用于专用场景（如仪器仪表画面）等在某位置监控图像不清晰的场景中，以获得更加清晰、迅速的聚焦效果。专用聚焦算法包含前景聚焦、后景聚焦与区域聚焦。

## 1.2 集成流程

- 1.客户端获取设备云台能力，GET /ISAPI/PTZCtrl/capabilities，返回 isSupportPresetsProFocus 为 true，表示支持预置点专用聚焦；
- 2.客户端获取指定通道预置点专用聚焦参数能力，GET /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg/capabilities?format=json,包括：专用聚焦使能、预置点聚焦列表能力；
- 3.客户端配置指定通道预置点专用聚焦参数，控制使能开关，针对全部预置点：  
获取：GET /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg?format=json；  
配置：PUT /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg?format=json；
- 4.客户端配置指定通道单个预置点专用聚焦参数（当个别预置点在步骤3的默认聚焦方式下不清晰时，可单独配置）：  
添加：POST /ISAPI/PTZCtrl/channels/<channelID>/AddPresetsProFocusCfg?format=json；  
获取：POST /ISAPI/PTZCtrl/channels/<channelID>/GetPresetsProFocusCfg?format=json；  
配置：PUT /ISAPI/PTZCtrl/channels/<channelID>/ModifyPresetsProFocusCfg?format=json；  
删除：POST /ISAPI/PTZCtrl/channels/<channelID>/DeletePresetsProFocusCfg?format=json；  
获取多个：GET /ISAPI/PTZCtrl/channels/<channelID>/GetMultiPresetsProFocusCfg?format=json；

## 1.3 异常处理

错误码

statusCode	statusString	subStatusCode	errorCode	errorMsg	中文描述	备注
4	Invalid Operation	numberReachedLlimit	0x40001119	The number reached limit.	配置的条目已经达到上限，不允许再添加，请删除无用条目后，再进行尝试	

## 2 功能接口

### 2.1 图像聚焦

## 2.1.1 获取指定通道图像能力

### URL

GET /ISAPI/Image/channels/<channelID>/imageCap

### 查询参数

参数名称	参数类型	描述
channelID	string	--

### 请求报文

无

### 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ImageCap xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 区域聚焦曝光能力, attr:version{req, string, 协议版本}-->
  <isSupportRegionalExposure>
    <!--ro, opt, bool, 是否支持区域曝光-->true
  </isSupportRegionalExposure>
  <isSupportRegionalFocus>
    <!--ro, opt, bool, 是否支持区域聚焦-->true
  </isSupportRegionalFocus>
  <isSupportManualBackgroundCorrect>
    <!--ro, opt, bool, 是否支持手动外部校正图像背景, desc:对应
uri:/ISAPI/Image/channels/<channelID>/ManualBackgroundCorrect-->true
  </isSupportManualBackgroundCorrect>
  <isSupportManualShutterCorrect>
    <!--ro, opt, bool, 是否支持手动内部校正图像背景, desc:对应
uri:/ISAPI/Image/channels/<channelID>/ManualShutterCorrect-->true
  </isSupportManualShutterCorrect>
  <RegionalExposure>
    <!--ro, opt, object, 区域曝光, desc:返回该节点表示支持GET和PUT
/ISAPI/Image/channels/<channelID>/regionalExposure。仅返回isSupportRegionalExposure为true表示只支持PUT /ISAPI/Image/channels/<channelID>/regionalExposure。支持PUT方法必须返回
isSupportRegionalExposure。-->
    <enabled opt="true,false" def="false">
      <!--ro, opt, bool, 区域曝光使能, attr:opt{req, string, 取值范围},def{req, string, 默认
值}-->true
    </enabled>
    <showRlueEnabled opt="true,false" def="false">
      <!--ro, opt, bool, 区域框显示使能, attr:opt{req, string, 取值范围},def{req, string, 默
认值}-->true
    </showRlueEnabled>
  </RegionalExposure>
</ImageCap>
```

### 2.1.2 删除指定通道单个预置点专用聚焦参数

URL

POST /ISAPI/PTZCtrl/channels/<channelID>/DeletePresetsProFocusCfg?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

```
{
  "presetID": 1
  /*req, int, 预置点ID, range:[1,300]*/
}
```

响应报文

```
{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
  "subStatusCode": "ok",
  /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
  "errorCode": 1,
  /*ro, opt, int, 错误码, desc:当statusCode不为1时，与subStatusCode对应*/
  "errorMsg": "ok",
  /*ro, opt, string, 错误信息, desc:当statusCode不为1时，必须返回,解释信息在协议约束中，允许设备在后续的版本迭代中，进行优化丰富提升（不限制死）*/
  "MErrCode": "0x00000000",
  /*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
  "MErrDevSelfEx": "0x00000000"
  /*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段，自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

### 2.1.3 配置指定通道预置点专用聚焦参数

URL

PUT /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

```
{
  "enabled": false
  /*req, bool, 使能, desc:不启用状态下: 预置点按照/ISAPI/Image/channels/1/focusConfiguration指定的方式聚焦; 启用、关闭切换, 针对通道所有预置点*/
}
```

响应报文

```
{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时 (1表示成功且无特殊状态) 必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时 (OK表示成功且无特殊状态) 必须返回*/
  "subStatusCode": "ok",
  /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示 (ok表示成功且无特殊状态) 必须返回*/
  "errorCode": 1,
  /*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
  "errorMsg": "ok",
  /*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许设备在后续的版本迭代中, 进行优化丰富提升 (不限制死) */
  "MErrCode": "0x00000000",
  /*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
  "MErrDevSelfEx": "0x00000000"
  /*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

2.1.4 获取指定通道预置点专用聚焦参数

URL

GET /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
{
  "enabled": false
  /*ro, req, bool, 使能, desc:不启用状态下: 预置点按
照/ISAPI/Image/channels/1/focusConfiguration指定的方式聚焦; 启用、关闭切换, 针对通道所有预置点*/
}
```

## 2.1.5 配置指定通道单个预置点专用聚焦参数

### URL

PUT /ISAPI/PTZCtrl/channels/<channelID>/ModifyPresetsProFocusCfg?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	--

### 请求报文

```
{
  "presetID": 1,
  /*req, int, 预置点ID, range:[1,300]*/
  "focusMode": "foregroundFocus",
  /*req, enum, 聚焦方式, subType:string, [foregroundFocus#前景聚焦,backgroundFocus#背景聚
焦,regionalFocus#区域聚焦]*/
  "focusRect": {
    /*opt, object, 聚焦矩形框, dep:and,{$.focusMode,eq,regionalFocus}, desc:屏幕左上角为坐标
原点*/
    "height": 1.000,
    /*req, float, 高度, range:[0.000,1.000]*/
    "width": 1.000,
    /*req, float, 宽度, range:[0.000,1.000]*/
    "x": 0.000,
    /*req, float, 区域左上角顶点的水平坐标, range:[0.000,1.000]*/
    "y": 0.000
    /*req, float, 区域左上角顶点的垂直坐标, range:[0.000,1.000]*/
  }
}
```

### 响应报文

```
{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时(1表示成功且无特殊状态)必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时(OK表示成功且无特殊状态)必须
返回*/
  "subStatusCode": "ok",
  /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示(ok表示成功且无特殊状态)必须返
回*/
  "errorCode": 1,
```

```
/*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
"errorMsg": "ok",
/*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许设备
在后续的版本迭代中, 进行优化丰富提升(不限制死)*/
"MErrCode": "0x00000000",
/*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
"MErrDevSelfEx": "0x00000000"
/*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

### 2.1.6 添加指定通道单个预置点专用聚焦参数

#### URL

POST /ISAPI/PTZCtrl/channels/<channelID>/AddPresetsProFocusCfg?format=json

#### 查询参数

参数名称	参数类型	描述
channelID	string	--

#### 请求报文

```
{
  "presetID": 1,
  /*req, int, 预置点ID, range:[1,300]*/
  "focusMode": "foregroundFocus",
  /*req, enum, 聚焦方式, subType:string, [foregroundFocus#前景聚焦,backgroundFocus#背景
聚焦,regionalFocus#区域聚焦]*/
  "focusRect": {
    /*opt, object, 聚焦矩形框, dep:and,{$.focusMode,eq,regionalFocus}, desc:屏幕左上角为坐
标原点*/
    "height": 1.000,
    /*req, float, 高度, range:[0.000,1.000]*/
    "width": 1.000,
    /*req, float, 宽度, range:[0.000,1.000]*/
    "x": 0.000,
    /*req, float, 区域左上角顶点的水平坐标, range:[0.000,1.000]*/
    "y": 0.000
    /*req, float, 区域左上角顶点的垂直坐标, range:[0.000,1.000]*/
  }
}
```

#### 响应报文

```
{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时(1表示成功且无特殊状态)必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时(OK表示成功且无特殊状态)必
```

```
须返回*/
    "subStatusCode": "ok",
    /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示 (ok表示成功且无特殊状态) 必须
返回*/
    "errorCode": 1,
    /*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
    "errorMsg": "ok",
    /*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许
设备在后续的版本迭代中, 进行优化丰富提升 (不限制死) */
    "MErrCode": "0x00000000",
    /*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
    "MErrDevSelfEx": "0x00000000"
    /*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

### 2.1.7 获取指定通道单个预置点专用聚焦参数

URL

POST /ISAPI/PTZCtrl/channels/<channelID>/GetPresetsProFocusCfg?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

```
{
    "presetID": 1
    /*req, int, 预置点ID, range:[1,300]*/
}
```

响应报文

```
{
    "focusMode": "foregroundFocus",
    /*ro, req, enum, 聚焦方式, subType:string, [foregroundFocus#前景聚焦,backgroundFocus#
背景聚焦,regionalFocus#区域聚焦]*/
    "focusRect": {
        /*ro, opt, object, 聚焦矩形框, dep:and,{$.focusMode,eq,regionalFocus}, desc:屏幕左上角
为坐标原点*/
        "height": 1.000,
        /*ro, req, float, 高度, range:[0.000,1.000]*/
        "width": 1.000,
        /*ro, req, float, 宽度, range:[0.000,1.000]*/
        "x": 0.000,
        /*ro, req, float, 区域左上角顶点的水平坐标, range:[0.000,1.000]*/
        "y": 0.000
        /*ro, req, float, 区域左上角顶点的垂直坐标, range:[0.000,1.000]*/
    }
}
```

```
}  
}
```

## 2.1.8 获取指定通道多个预置点专用聚焦参数

### URL

GET /ISAPI/PTZCtrl/channels/<channelID>/GetMultiPresetsProFocusCfg?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	--

### 请求报文

无

### 响应报文

```
{  
  "presetsProFocusList": [  
    /*ro, opt, array, 预置点专用聚焦列表, subType:object*/  
    {  
      "presetID": 1,  
      /*ro, req, int, 预置点ID, range:[1,300]*/  
      "focusMode": "foregroundFocus",  
      /*ro, req, enum, 聚焦方式, subType:string, [foregroundFocus#前景聚  
焦,backgroundFocus#背景聚焦,regionalFocus#区域聚焦]*/  
      "focusRect": {  
        /*ro, opt, object, 聚焦矩形框, dep:and,  
        {$.focusMode,eq,regionalFocus}, desc:屏幕左上角为坐标原点*/  
        "height": 1.000,  
        /*ro, req, float, 高度, range:[0.000,1.000]*/  
        "width": 1.000,  
        /*ro, req, float, 宽度, range:[0.000,1.000]*/  
        "x": 0.000,  
        /*ro, req, float, 区域左上角顶点的水平坐标, range:  
[0.000,1.000]*/  
        "y": 0.000  
        /*ro, req, float, 区域左上角顶点的垂直坐标, range:  
[0.000,1.000]*/  
      }  
    }  
  ]  
}
```

## 2.1.9 获取指定通道最小聚焦距离标定参数

### URL

GET /ISAPI/Thermal/channels/<channelID>/focusCalibrate



查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusCalibrate xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 标定参数, attr:version{req, string, 协议版本}-->
  <enabled>
    <!--ro, req, bool, 使能-->true
  </enabled>
  <distance>
    <!--ro, opt, int, 最小聚焦限位标定参数, range:[0,255]-->0
  </distance>
  <maxMTFFocus>
    <!--ro, opt, int, 最大MTF值所对应的电机聚焦坐标值, range:[0,65535]-->0
  </maxMTFFocus>
</FocusCalibrate>
```

2.1.10 获取指定通道最小聚焦距离标定参数能力

URL

GET /ISAPI/Thermal/channels/<channelID>/focusCalibrate/capabilities

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusCalibrateCap xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 最小聚焦距离标定参数, attr:version{req, string, 协议版本}-->
  <enabled opt="true,false">
    <!--ro, req, bool, 使能, attr:opt{req, string, 取值范围}-->true
  </enabled>
```

```
<distance min="0" max="255">
  <!--ro, opt, int, 最小聚焦限位标定参数, range:[0,255], attr:min{req, int, 最小值},max{req,
int, 最大值}-->0
</distance>
<FocusCalibrateResult>
  <!--ro, opt, object, 标定结果-->
  <focus min="0" max="65535">
    <!--ro, opt, int, 焦距, range:[0,65535], attr:min{req, int, 最小值},max{req, int, 最大
值}-->0
  </focus>
</FocusCalibrateResult>
<maxMTFFocus min="0" max="65535">
  <!--ro, opt, int, 最大MTF值所对应的电机聚焦坐标值, range:[0,65535], attr:min{req, int, 最小
值},max{req, int, 最大值}-->0
</maxMTFFocus>
</FocusCalibrateCap>
```

2.1.11 标定指定通道最小聚焦距离

URL

POST /ISAPI/Thermal/channels/<channelID>/focusCalibrate

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusCalibrate xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--opt, object, 标定参数, attr:version{req, string, 协议版本}-->
  <enabled>
    <!--req, bool, 使能-->true
  </enabled>
  <distance>
    <!--opt, int, 最小聚焦限位标定参数, range:[0,255]-->0
  </distance>
  <maxMTFFocus>
    <!--opt, int, 最大MTF值所对应的电机聚焦坐标值, range:[0,65535]-->0
  </maxMTFFocus>
</FocusCalibrate>
```

响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusCalibrateResult xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 标定结果, attr:version{req, string, 协议版本}-->
  <focus>
    <!--ro, opt, int, 焦距, range:[0,65535]-->0
  </focus>
  <resultCode>
    <!--ro, opt, enum, 标定结果, subType:string, [0#标定成功,1#标定行程超过软限位行程范围,2#无穷远物距未标定,3#镜头温感错误,4#微距镜标定开启失败,5#微距镜标定关闭失败,6#微距镜标定功能未正常打开]-->0
  </resultCode>
</FocusCalibrateResult>
```

### 2.1.12 配置指定通道视频聚焦

#### URL

PUT /ISAPI/System/Video/inputs/channels/<channelID>/focus

#### 查询参数

参数名称	参数类型	描述
channelID	string	--

#### 请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusData xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--opt, object, 聚焦控制参数, attr:version{req, string, 协议版本}-->
  <focus>
    <!--req, int, 聚焦控制, range:[-100,100], desc:负值表示远,正值表示近,数值大小表示变化速率(最大速率的百分比),0表示停止-->0
  </focus>
</FocusData>
```

#### 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>
    <!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]-->0
  </statusCode>
</ResponseStatus>
```

```
</statusCode>
<statusString>
  <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的XML内容,Reboot Required#设备重启]-->OK
</statusString>
<subStatusCode>
  <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
</subStatusCode>
<description>
  <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误信息, 用于快速定位评查问题-->badXmlFormat
</description>
<MErrCode>
  <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
</MErrCode>
<MErrDevSelfEx>
  <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
</MErrDevSelfEx>
</ResponseStatus>
```

### 2.1.13 配置指定通道最小聚焦距离标定参数

#### URL

PUT /ISAPI/Thermal/channels/<channelID>/focusCalibrate

#### 查询参数

参数名称	参数类型	描述
channelID	string	--

#### 请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusCalibrate xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--opt, object, 标定参数, attr:version{req, string, 协议版本}-->
  <enabled>
    <!--req, bool, 使能-->true
  </enabled>
  <distance>
    <!--opt, int, 最小聚焦限位标定参数, range:[0,255]-->0
  </distance>
  <maxMTFFocus>
    <!--opt, int, 最大MTF值所对应的电机聚焦坐标值, range:[0,65535]-->0
  </maxMTFFocus>
</FocusCalibrate>
```

## 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>
    <!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]-->0
  </statusCode>
  <statusString>
    <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的XML内容,Reboot Required#设备重启]-->OK
  </statusString>
  <subStatusCode>
    <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
  </subStatusCode>
  <description>
    <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误信息, 用于快速定位评查问题-->badXmlFormat
  </description>
  <MErrCode>
    <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
  </MErrCode>
  <MErrDevSelfEx>
    <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
  </MErrDevSelfEx>
</ResponseStatus>
```

### 2.1.14 配置ABF控制参数

#### URL

PUT /ISAPI/System/Hardware/ABF

#### 查询参数

无

#### 请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ABF xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--req, object, ABF节点, attr:version{req, string, 协议版本}-->
  <enabled>
    <!--req, bool, 使能-->true
  </enabled>
</ABF>
```

## 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>
    <!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device
Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]--
>0
  </statusCode>
  <statusString>
    <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备
异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的
XML内容,Reboot Required#设备重启]-->OK
  </statusString>
  <subStatusCode>
    <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
  </subStatusCode>
  <description>
    <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误
信息,用于快速定位评查问题-->badXmlFormat
  </description>
  <MErrCode>
    <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
  </MErrCode>
  <MErrDevSelfEx>
    <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
  </MErrDevSelfEx>
</ResponseStatus>
```

### 2.1.15 配置指定通道区域聚焦参数

#### URL

PUT /ISAPI/Image/channels/<channelID>/regionalFocus

#### 查询参数

参数名称	参数类型	描述
channelID	string	--

## 请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<RegionalFocus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--req, object, 区域聚焦, attr:version{req, string, 协议版本}-->
  <StartPoint>
    <!--opt, object, 起始点, desc:屏幕左上角为坐标原点-->
    <positionX>
      <!--req, int, x坐标, range:[0,1000]-->0
    </positionX>
    <positionY>
      <!--req, int, y坐标, range:[0,1000]-->0
    </positionY>
  </StartPoint>
  <EndPoint>
    <!--opt, object, 结束点, desc:屏幕左上角为坐标原点-->
    <positionX>
      <!--req, int, x坐标, range:[0,1000]-->0
    </positionX>
    <positionY>
      <!--req, int, y坐标, range:[0,1000]-->0
    </positionY>
  </EndPoint>
</RegionalFocus>
```

## 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>
    <!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]-->0
  </statusCode>
  <statusString>
    <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的XML内容,Reboot Required#设备重启]-->OK
  </statusString>
  <subStatusCode>
    <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
```

```

</subStatusCode>
<description>
  <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误
信息, 用于快速定位评查问题-->badXmlFormat
</description>
<MErrCode>
  <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
</MErrCode>
<MErrDevSelfEx>
  <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
</MErrDevSelfEx>
</ResponseStatus>

```

## 2.1.16 配置指定通道云台缩放焦点信息

### URL

PUT /ISAPI/PTZCtrl/channels/<channelID>/zoomFocus

### 查询参数

参数名称	参数类型	描述
channelID	string	

### 请求报文

```

<?xml version="1.0" encoding="UTF-8"?>

<ZoomFocus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--opt, object, 缩放焦点, attr:version{req, string, 协议版本}-->
  <pqrsZoom>
    <!--opt, int, pqrs缩放-->1
  </pqrsZoom>
  <mnstFocus>
    <!--opt, int, mnst焦点-->1
  </mnstFocus>
</ZoomFocus>

```

### 响应报文

```

<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>

```



```
<!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device
Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]--
>0
</statusCode>
<statusString>
    <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备
异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的
XML内容,Reboot Required#设备重启]-->OK
</statusString>
<subStatusCode>
    <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
</subStatusCode>
<description>
    <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误
信息, 用于快速定位评查问题-->badXmlFormat
</description>
<MErrCode>
    <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
</MErrCode>
<MErrDevSelfEx>
    <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
</MErrDevSelfEx>
</ResponseStatus>
```

2.1.17 获取AFD辅助聚焦

URL

GET /ISAPI/Image/channels/<channelID>/AFDFoucsAssist?format=json

查询参数

参数名称	参数类型	描述
channelID	string	通道号

请求报文

无

响应报文

```
{
/*req, object, AFD辅助聚焦, range:[,], desc:*/
    "enabled": true,
/*req, boolean, 使能, const:, desc:*/
    "rect":{
/*opt, object, 规则框, range:[,], desc:*/
        "height": 0.0,
/*req, number, 高度, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
        "width": 0.0,
```

```

        /*req, number, 宽度, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
        "x": 0.0,
        /*req, number, X坐标, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
        "y": 0.0
        /*req, number, Y坐标, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
    }
}

```

## 2.1.18 设置AFD辅助聚焦

### URL

PUT /ISAPI/Image/channels/<channelID>/AFDFoucsAssist?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	通道号

### 请求报文

```

{
/*req, object, AFD辅助聚焦, range:[,], desc:*/
  "enabled": true,
  /*req, boolean, 使能, const:, desc:*/
  "rect":{
    /*opt, object, 规则框, range:[,], desc:*/
    "height": 0.0,
    /*req, number, 高度, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
    "width": 0.0,
    /*req, number, 宽度, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
    "x": 0.0,
    /*req, number, X坐标, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
    "y": 0.0
    /*req, number, Y坐标, const:, enum:[], range:[0,1], multipleOf:, unit:,
desc:*/
  }
}

```

### 响应报文

```

{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
  "statusString": "OK",

```

```
/*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
"subStatusCode": "ok",
/*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
"errorCode": 1,
/*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
"errorMsg": "ok",
/*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许设备在后续的版本迭代中, 进行优化丰富提升（不限制死）*/
"MErrCode": "0x00000000",
/*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
"MErrDevSelfEx": "0x00000000"
/*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

### 2.1.19 获取AFD辅助聚焦能力

URL

GET /ISAPI/Image/channels/<channelID>/AFDFoucsAssist/capabilities?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
{
/*req, object, 获取AFD辅助聚焦能力, range:[,], desc:*/
  "AFDFoucsAssistCap":{
    /*opt, object, 获取AFD辅助聚焦能力, range:[,], desc:对应功能:GET/PUT
    /ISAPI/Image/channels/<channelID>/AFDFoucsAssist?format=json*/
    "enabled":{
      /*req, object, 使能取值范围, range:[,], desc:*/
      "@opt":[
        /*opt, array, 使能枚举列表, range:[1,2], uniqueItems:true, desc:*/
        true,false
        /*req, boolean, 枚举值, const:, desc:*/
      ],
      "@def": true
      /*opt, boolean, 默认值, const:, desc:*/
    },
    "rect":{
      /*opt, object, 规则框, range:[,], desc:*/
      "height":{
        /*req, object, 高度取值范围, range:[,], desc:*/
```

```

        "@def": 0.0,
        /*opt, number, 默认值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@min": 0.0,
        /*opt, number, 最小值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@max": 1.0
        /*opt, number, 最大值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
    },
    "width":{
        /*req, object, 宽度取值范围, range:[,], desc:*/
        "@def": 0.0,
        /*opt, number, 默认值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@min": 0.0,
        /*opt, number, 最小值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@max": 1.0
        /*opt, number, 最大值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
    },
    "x":{
        /*req, object, x坐标取值范围, range:[,], desc:*/
        "@def": 0.0,
        /*opt, number, 默认值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@min": 0.0,
        /*opt, number, 最小值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@max": 1.0
        /*opt, number, 最大值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
    },
    "y":{
        /*req, object, y坐标取值范围, range:[,], desc:*/
        "@def": 0.0,
        /*opt, number, 默认值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@min": 0.0,
        /*opt, number, 最小值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
        "@max": 1.0
        /*opt, number, 最大值, const:, enum:[], range:[,],
multipleOf:, unit:, desc:*/
    }
}
}
}

```

## 2.1.20 获取聚焦保持参数

URL

GET /ISAPI/Image/channels/<channelID>/KeepFocusParams?format=json

查询参数

参数名称	参数类型	描述
channelID	string	通道号

请求报文

无

响应报文

```
{
/*req, object, 参数, range:[,], additionalProperties:false, desc:*/
  "holdCurrentPercent": 25
  /*opt, integer, 保持电流百分比, const:, enum:[], range:[0,100], multipleOf:, step:1,
unit:%, desc:*/
}
```

2.1.21 设置聚焦保持参数

URL

PUT /ISAPI/Image/channels/<channelID>/KeepFocusParams?format=json

查询参数

参数名称	参数类型	描述
channelID	string	通道号

请求报文

```
{
/*req, object, 参数, range:[,], additionalProperties:false, desc:*/
  "holdCurrentPercent": 25
  /*opt, integer, 保持电流百分比, const:, enum:[], range:[0,100], multipleOf:, step:1,
unit:%, desc:*/
}
```

响应报文

```
{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
  "subStatusCode": "ok",
  /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示（ok表示成功且无特殊状态）必须
```

```
返回*/
    "errorCode": 1,
    /*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
    "errorMsg": "ok",
    /*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许
    设备在后续的版本迭代中, 进行优化丰富提升(不限制死)*/
    "MErrCode": "0x00000000",
    /*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
    "MErrDevSelfEx": "0x00000000"
    /*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
    模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}
```

## 2.1.22 获取聚焦保持参数能力

### URL

GET /ISAPI/Image/channels/<channelID>/KeepFocusParams/capabilities?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	--

### 请求报文

无

### 响应报文

```
{
  /*req, object, 获取聚焦保持参数能力, range:[,], additionalProperties:false, desc:*/
  "KeepFocusParamsCap":{
    /*opt, object, 聚焦保持参数能力, range:[,], additionalProperties:false, desc:对应功
    能:GET/PUT /ISAPI/Image/channels/<channelID>/KeepFocusParams?format=json*/
    "holdCurrentPercent":{
      /*opt, object, 保持电流百分比取值范围, range:[,], additionalProperties:false,
      desc:*/
      "@def": 25,
      /*opt, integer, 默认值, const:, enum:[], range:[,], multipleOf:,
      step:, unit:, desc:*/
      "@min": 0,
      /*req, integer, 最小值, const:, enum:[], range:[,], multipleOf:,
      step:, unit:, desc:*/
      "@max": 100
      /*req, integer, 最大值, const:, enum:[], range:[,], multipleOf:,
      step:, unit:, desc:*/
    }
  }
}
```

## 2.1.23 获取指定通道预置点专用聚焦参数能力

## URL

GET /ISAPI/PTZCtrl/channels/<channelID>/PresetsProFocusCfg/capabilities?format=json

## 查询参数

参数名称	参数类型	描述
channelID	string	--

## 请求报文

无

## 响应报文

```
{
  "enabled": {
    /*ro, req, object, 使能, desc:启用或关闭切换, 预置点聚焦方式切换均针对所有预置点*/
    "@opt": [true, false],
    /*ro, req, array, 范围, subType:bool*/
    "@def": false
    /*ro, opt, bool, 默认值*/
  },
  "PresetsFocusCap": {
    /*ro, opt, object, 预置点聚焦列表能力, desc:当个别预置点在默认前景聚焦下不合适, 需要更换为后景
    或者区域聚焦方式时, 需要添加预置点聚焦配置.*/
    "@size": 300,
    /*ro, req, int, 最大个数*/
    "focusMode": {
      /*ro, req, object, 聚焦方式*/
      "@opt": ["foregroundFocus", "backgroundFocus", "regionalFocus"],
      /*ro, req, array, 范围, subType:string, desc:前景聚焦
      #foregroundFocus,后景聚焦#backgroundFocus,区域聚焦#regionalFocus*/
      "@def": "foregroundFocus"
      /*ro, opt, string, 默认值*/
    },
    "presetID": {
      /*ro, req, object, 预置点*/
      "@min": 1,
      /*ro, req, int, 最小值, range:[1,300]*/
      "@max": 300
      /*ro, req, int, 最大值, range:[1,300]*/
    },
    "focusRect": {
      /*ro, opt, object, 聚焦矩形框, desc:屏幕左上角为坐标原点,依赖于focusMode支持
      regionalFocus*/
      "height": {
        /*ro, opt, object, 高度*/
        "min": 0.000,
        /*ro, req, float, 最小值, range:[0.000,1.000]*/
        "max": 1.000
        /*ro, req, float, 最大值, range:[0.000,1.000]*/
      },
    },
  },
}
```

```
        "width": {
/*ro, opt, object, 宽度*/
            "min": 0.000,
/*ro, req, float, 最小值, range:[0.000,1.000]*/
            "max": 1.000
/*ro, req, float, 最大值, range:[0.000,1.000]*/
        },
        "x": {
/*ro, opt, object, 区域左上角顶点的水平坐标*/
            "min": 0.000,
/*ro, req, float, 最小值, range:[0.000,1.000]*/
            "max": 1.000
/*ro, req, float, 最大值, range:[0.000,1.000]*/
        },
        "y": {
/*ro, opt, object, 区域左上角顶点的垂直坐标*/
            "min": 0.000,
/*ro, req, float, 最小值, range:[0.000,1.000]*/
            "max": 1.000
/*ro, req, float, 最大值, range:[0.000,1.000]*/
        }
    }
}
}
```

### 2.1.24 配置指定通道普通聚焦参数

#### URL

PUT /ISAPI/Image/channels/<channelID>/focusConfiguration

#### 查询参数

参数名称	参数类型	描述
channelID	string	

#### 请求报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusConfiguration xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--opt, object, 聚焦配置, attr:version{req, string, 协议版本}-->
  <focusStyle>
    <!--req, enum, 聚焦类型, subType:string, [AUTO#自动,MANUAL#手动,SEMIAUTOMATIC#半自动,distanceCalculate#距离推算]-->AUTO
  </focusStyle>
  <focusLimited>
    <!--opt, enum, 聚焦限制, subType:int, [10#10厘米,50#50厘米,100#1米,150#1.5米,300#3米,600#6米,1000#10米,2000#20米,5000#50米,20000#200米,50000#500米,65535#655.35米]-->10
  </focusLimited>
  <focusPosition>
    <!--opt, int, 焦点位置-->1
  </focusPosition>
</FocusConfiguration>
```



```

</focusPosition>
<focusSpeed>
  <!--opt, int, 聚焦速度-->1
</focusSpeed>
<focusSensitivity>
  <!--opt, int, 聚焦灵敏度-->1
</focusSensitivity>
<temperatureChangeAdaptEnabled>
  <!--opt, bool, 温度变化适应使能-->true
</temperatureChangeAdaptEnabled>
<relativeFocusPos>
  <!--opt, int, 相对焦点位置-->1
</relativeFocusPos>
<lensType>
  <!--opt, enum, 聚焦镜头类型, subType:string, [1140#1140,1050#1050]-->1140
</lensType>
<highTemperaturePriority>
  <!--opt, bool, 高温优先模式使能-->true
</highTemperaturePriority>
<focusRange>
  <!--opt, enum, 聚焦调节范围, subType:string, [min#最小,medium#适中,max#最大]-->max
</focusRange>
<focusDistanceMode>
  <!--opt, enum, 聚焦距离模式, subType:string, [compatible#兼容模式], desc:compatible-兼容模式, 在球罩设备上调整聚焦限制距离focusLimited,仍然虚焦的情况, 选择兼容模式, focusLimited参数失效, 设备释放镜头最大聚焦能力, 使镜头获得更大的焦距范围, 解决球罩设备近物距聚焦效果差的问题。开启该功能, 聚焦速度会变慢, 非球罩设备不建议使用。-->compatible
</focusDistanceMode>
<zoomFocusFollowEnabled>
  <!--opt, bool, 变倍聚焦跟随, desc:开启时, 热成像变倍过程中, 聚焦值实时补偿, 画面全程不虚焦-->true
</zoomFocusFollowEnabled>
<focusRegion>
  <!--opt, enum, 聚焦区域, subType:string, [center#中心聚焦,pixelToPixel#全屏聚焦], dep:or,
{$.FocusConfiguration.focusStyle,eq,AUTO},
{$.FocusConfiguration.focusStyle,eq,SEMIAUTOMATIC}, desc:#center中心聚焦: 确保画面中心点周围约
4/16窗口大小的物体能够清晰地呈现。#pixelToPixel全屏聚焦: 确保整个画面中的所有物体都能够清晰地呈现。且当聚
焦模式为自动和半自动时支持聚焦区域的选择; -->center
</focusRegion>
<focusScene>
  <!--opt, enum, 聚焦场景, subType:string, [lowIllumination#低照度,normal#普通], desc:该聚焦
场景与图像场景(/ISAPI/Image/channels/<channelID>/mountingScenario)之间逻辑关系: 1、若聚焦场景为普
通模式时与图像参数场景保持一致;2、聚焦场景为其他模式时,场景参数以聚焦场景的为准-->normal
</focusScene>
<timeFocusEnabled>
  <!--opt, bool, 定时聚焦使能, dep:or,{$.FocusConfiguration.focusStyle,eq,AUTO},
{$.FocusConfiguration.focusStyle,eq,SEMIAUTOMATIC}, desc:仅在聚焦类型为自动和半自动时支持配置--
>true
</timeFocusEnabled>
<timeFocus>
  <!--opt, int, 定时聚焦时间间隔, range:[1,1440], unit:min, unitType:时间, dep:and,
{$.FocusConfiguration.timeFocusEnabled,eq,true}-->1
</timeFocus>
<focusDistance>

```

```

    <!--opt, float, 聚焦推算距离, range:[0.30,99.99], step:0.01, unit:m, unitType:长度,
dep:and,{$.FocusConfiguration.focusStyle,eq,distanceCalculate}, desc:精确到小数点2位-->0.35
  </focusDistance>
  <focusStrategy>
    <!--opt, enum, 聚焦策略, subType:string, [general#普通,DToF#DToF], desc:根据视频或TOF确定聚
焦距离-->DToF
  </focusStrategy>
</FocusConfiguration>

```

## 响应报文

```

<?xml version="1.0" encoding="UTF-8"?>

<ResponseStatus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, 响应消息, attr:version{ro, req, string, 版本}-->
  <requestURL>
    <!--ro, opt, string, 请求的URL, range:[0,1024]-->null
  </requestURL>
  <statusCode>
    <!--ro, req, enum, 状态码, subType:int, [0#OK,1#OK,2#Device Busy,3#Device
Error,4#Invalid Operation,5#Invalid XML Format,6#Invalid XML Content,7#Reboot Required]--
>0
  </statusCode>
  <statusString>
    <!--ro, req, enum, 状态信息, subType:string, [OK#成功,Device Busy#设备忙,Device Error#设备
异常,Invalid Operation#无效的操作,Invalid XML Format#无效的XML格式,Invalid XML Content#无效的
XML内容,Reboot Required#设备重启]-->OK
  </statusString>
  <subStatusCode>
    <!--ro, req, string, 详细错误码英文描述, desc:详细错误码的英文描述-->OK
  </subStatusCode>
  <description>
    <!--ro, opt, string, 自定义错误信息描述, range:[0,1024], desc:设备应用程序返回的自定义详细错误
信息,用于快速定位评查问题-->badXmlFormat
  </description>
  <MErrCode>
    <!--ro, opt, string, 功能模块化错误码, desc:通用错误码都在此字段范围中-->0x00000000
  </MErrCode>
  <MErrDevSelfEx>
    <!--ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义
模块化错误码一共有MErrCode 的后20bit和此字段的32bit-->0x00000000
  </MErrDevSelfEx>
</ResponseStatus>

```

### 2.1.25 获取指定通道普通聚焦参数能力

#### URL

GET /ISAPI/Image/channels/<channelID>/focusConfiguration/capabilities

#### 查询参数

参数名称	参数类型	描述
channelID	string	

## 请求报文

无

## 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusConfiguration xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 普通聚焦, attr:version{req, string, 协议版本}-->
  <focusStyle opt="AUTO,MANUAL,SEMIAUTOMATIC,distanceCalculate">
    <!--ro, req, string, 聚焦类型, attr:opt{req, string, 取值范围}-->test
  </focusStyle>
  <focusLimited opt="10,50,100,150,300,600,1000,2000,5000,20000,50000,65535">
    <!--ro, opt, enum, 聚焦限制, subType:string, [10#10厘米,50#50厘米,100#1米,150#1.5米,300#3米,600#6米,1000#10米,2000#20米,5000#50米,20000#200米,50000#500米,65535#655.35米], attr:opt{req, string, 取值范围}, desc:在室内球罩设备上调整聚焦距离仍然虚焦的情况, 建议focusLimited设置为兼容模式(取值0)。聚焦距离限制为兼容模式时, 设备释放镜头最大聚焦能力, 使镜头获得更大的焦距范围, 解决球罩设备近物距聚焦效果差的问题。但兼容模式会使聚焦速度会变慢, 非球罩设备不建议使用。-->10
  </focusLimited>
  <focusPosition min="1" max="10">
    <!--ro, opt, int, 焦点位置, attr:min{req, int, 最小值},max{req, int, 最大值}-->1
  </focusPosition>
  <focusSpeed min="1" max="10">
    <!--ro, opt, int, 聚焦速度, range:[1,3], attr:min{req, int, 聚焦速度最小值},max{req, int, 聚焦速度最大值}-->1
  </focusSpeed>
  <focusSensitivity min="1" max="10">
    <!--ro, opt, int, 聚焦灵敏度, range:[0,2], attr:min{req, int, 聚焦灵敏度最小值},max{req, int, 聚焦灵敏度最大值}-->1
  </focusSensitivity>
  <temperatureChangeAdaptEnabled opt="true,false">
    <!--ro, opt, bool, 温度变化适应使能, attr:opt{req, string, 取值范围}-->true
  </temperatureChangeAdaptEnabled>
  <relativeFocusPos min="1" max="10">
    <!--ro, opt, int, 相对焦点位置, range:[0,4294967295], attr:min{req, int, 相对焦点位置最小值},max{req, int, 相对焦点位置最大值}-->1
  </relativeFocusPos>
  <highTemperaturePriority opt="true,false">
    <!--ro, opt, bool, 高温优先模式使能, attr:opt{req, string, 高温优先模式使能取值范围}-->true
  </highTemperaturePriority>
  <focusRange opt="min,medium,max">
    <!--ro, opt, enum, 聚焦调节范围, subType:string, [min#最小,medium#适中,max#最大], attr:opt{req, string, 取值范围}-->max
  </focusRange>
  <focusDistanceMode opt="compatible">
    <!--ro, opt, string, 聚焦距离模式, attr:opt{req, string, 取值范围}, desc:支持返回该节点。compatible-兼容模式, 在球罩设备上调整聚焦限制距离focusLimited,仍然虚焦的情况, 选择兼容模式, focusLimited参数失效, 设备释放镜头最大聚焦能力, 使镜头获得更大的焦距范围, 解决球罩设备近物距聚焦效果差的问
```

题。开启该功能，聚焦速度会变慢，非球罩设备不建议使用。-->compatible

```
</focusDistanceMode>
<zoomFocusFollowEnabled opt="true,false">
  <!--ro, opt, bool, 变倍聚焦跟随, attr:opt{req, string, 取值范围}, desc:开启时，热成像变倍过程中，聚焦值实时补偿，画面全程不虚焦-->true
</zoomFocusFollowEnabled>
<focusRegion opt="center,pixelToPixel">
  <!--ro, opt, enum, 聚焦区域, subType:string, [center#中心聚焦,pixelToPixel#全屏聚焦],
dep:or,{$.FocusConfiguration.focusStyle,eq,AUTO},
{$.FocusConfiguration.focusStyle,eq,SEMIAUTOMATIC}, attr:opt{req, string, 取值范围},
desc:#center中心聚焦：确保画面中心点周围约4/16窗口大小的物体能够清晰地呈现。#pixelToPixel全屏聚焦：确保
整个画面中的所有物体都能够清晰地呈现。且当聚焦模式为自动和半自动时支持聚焦区域的选择-->center
</focusRegion>
<focusScene opt="lowIllumination,normal">
  <!--ro, opt, enum, 聚焦场景, subType:string, [lowIllumination#低照度,normal#普通],
attr:opt{req, string, 取值范围}, desc:该聚焦场景与图像场景
(/ISAPI/Image/channels/<channelID>/mountingScenario)之间逻辑关系：1、若聚焦场景为普通模式时与图像
参数场景保持一致；2、聚焦场景为其他模式时,场景参数以聚焦场景的为准-->normal
</focusScene>
<timeFocusEnabled opt="true,false">
  <!--ro, opt, bool, 定时聚焦使能, attr:opt{req, string, 取值范围}-->true
</timeFocusEnabled>
<timeFocus min="1" max="1440">
  <!--ro, opt, int, 定时聚焦时间间隔, range:[1,1440], attr:min{req, int, 定时聚焦时间间隔最小
值},max{req, int, 定时聚焦时间间隔最大值}-->1
</timeFocus>
<focusDistance min="0.30" max="99.99">
  <!--ro, opt, float, 聚焦推算距离, range:[0.30,99.99], step:0.01, unit:m, unitType:长度,
attr:min{req, float, 最小值},max{req, float, 最大值}-->0.35
</focusDistance>
<focusStrategy opt="general,DTOF" def="general">
  <!--ro, opt, enum, 聚焦策略, subType:string, [general#普通,DTOF#DTOF], attr:opt{req,
string, 取值范围},def{req, string, 默认值}, desc:根据视频或TOF确定聚焦距离-->DTOF
</focusStrategy>
</FocusConfiguration>
```

### 2.1.26 配置激光聚焦参数

URL

PUT /ISAPI/Image/channels/<channelID>/ModifyLaserFocusParams?format=json

查询参数

参数名称	参数类型	描述
channelID	string	通道号

请求报文

```
{
/*req, object, 参数, range:[,], desc:*/
    "controlType": "open"
    /*req, string, 控制类型, const:, range:[,], enum:[open#开启, close#关闭], format:,
mediaType:, pattern:, unit:, RAP:, desc:*/
}
```

响应报文

```
{
/*req, object, , range:[,], desc:*/
    "statusCode": 1,
    /*opt, integer, 状态码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
    "statusString": "OK",
    /*opt, string, 状态描述, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
    "subStatusCode": "ok",
    /*opt, string, 子状态码, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
    "errorCode": 1,
    /*opt, integer, 错误码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:当statusCode不为1时，与subStatusCode对应*/
    "errorMsg": "ok",
    /*opt, string, 错误信息, const:, range:[,], enum:[], format:, mediaType:, pattern:,
unit:, RAP:, desc:当statusCode不为1时，必须返回,解释信息在协议约束中，允许设备在后续的版本迭代中，进行
优化丰富提升（不限制死）*/
    "MErrCode": "0x00000000",
    /*opt, string, 功能模块化错误, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:通用错误码都在此字段范围中*/
    "MErrDevSelfEx": "0x00000000"
    /*opt, string, 功能模块化错误码, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:此字段是MErrCode自定义模块化错误码的扩展字段，自定义模块化错误码一共有
MErrCode 的后20bit和此字段的32bit*/
}
```

2.1.27 获取配置激光聚焦参数能力

URL

GET /ISAPI/Image/channels/<channelID>/ModifyLaserFocusParams/capabilities?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
{
/*req, object, 获取配置激光聚焦参数能力, range:[,], desc:*/
  "ModifyLaserFocusParamsCap":{
    /*opt, object, 配置激光聚焦参数能力, range:[,], desc:对应功能:PUT
/ISAPI/Image/channels/<channelID>/ModifyLaserFocusParams?format=json输入参数能力*/
    "controlType":{
      /*req, object, 控制类型取值范围, range:[,], desc:*/
      "@opt":[
        /*opt, array, 控制类型枚举列表, range:[1,2], uniqueItems:true,
desc:*/
          "open", "close"
          /*req, string, 枚举值, const:, range:[,], enum:[open#开启,
close#关闭], format:, mediaType:, pattern:, unit:, RAP:, desc:*/
        ],
        "@def": "open"
        /*opt, string, 默认值, const:, range:[,], enum:[], format:,
mediaType:, pattern:, unit:, RAP:, desc:*/
      ]
    }
  }
}
```

2.1.28 获取指定通道普通聚焦参数

URL

GET /ISAPI/Image/channels/<channelID>/focusConfiguration

查询参数

参数名称	参数类型	描述
channelID	string	

请求报文

无

响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<FocusConfiguration xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 聚焦配置, attr:version{req, string, 协议版本}-->
  <focusStyle>
    <!--ro, req, enum, 聚焦类型, subType:string, [AUTO#自动,MANUAL#手动,SEMIAUTOMATIC#半自动,distanceCalculate#距离推算,regionFocus#区域聚焦,laserFocus#激光聚焦]-->AUTO
  </focusStyle>
  <focusLimited>
    <!--ro, opt, enum, 聚焦限制, subType:int, [10#10厘米,50#50厘米,100#1米,150#1.5米,300#3米,600#6米,1000#10米,2000#20米,5000#50米,20000#200米,50000#500米,65535#655.35米]-->10
  </focusLimited>
  <focusPosition>
    <!--ro, opt, int, 焦点位置-->1
  </focusPosition>
</FocusConfiguration>
```

```

</focusPosition>
<focusSpeed>
  <!--ro, opt, int, 聚焦速度-->1
</focusSpeed>
<focusSensitivity>
  <!--ro, opt, int, 聚焦灵敏度-->1
</focusSensitivity>
<temperatureChangeAdaptEnabled>
  <!--ro, opt, bool, 温度变化适应使能-->true
</temperatureChangeAdaptEnabled>
<relativeFocusPos>
  <!--ro, opt, int, 相对焦点位置-->1
</relativeFocusPos>
<lensType>
  <!--ro, opt, enum, 聚焦镜头类型, subType:string, [1140#1140,1050#1050]-->1140
</lensType>
<highTemperaturePriority>
  <!--ro, opt, bool, 高温优先模式使能-->true
</highTemperaturePriority>
<focusRange>
  <!--ro, opt, enum, 聚焦调节范围, subType:string, [min#最小,medium#适中,max#最大]-->max
</focusRange>
<focusDistanceMode>
  <!--ro, opt, enum, 聚焦距离模式, subType:string, [compatible#兼容模式], desc:compatible-兼容模式, 在球罩设备上调整聚焦限制距离focusLimited,仍然虚焦的情况, 选择兼容模式, focusLimited参数失效, 设备释放镜头最大聚焦能力, 使镜头获得更大的焦距范围, 解决球罩设备近物距聚焦效果差的问题。开启该功能, 聚焦速度会变慢, 非球罩设备不建议使用。-->compatible
</focusDistanceMode>
<zoomFocusFollowEnabled>
  <!--ro, opt, bool, 变倍聚焦跟随, desc:开启时, 热成像变倍过程中, 聚焦值实时补偿, 画面全程不虚焦-->true
</zoomFocusFollowEnabled>
<focusRegion>
  <!--ro, opt, enum, 聚焦区域, subType:string, [center#中心聚焦,pixelToPixel#全屏聚焦], dep:or,{$.FocusConfiguration.focusStyle,eq,AUTO},{$.FocusConfiguration.focusStyle,eq,SEMIAUTOMATIC}, desc:#center中心聚焦: 确保画面中心点周围约4/16窗口大小的物体能够清晰地呈现。#pixelToPixel全屏聚焦: 确保整个画面中的所有物体都能够清晰地呈现。且当聚焦模式为自动和半自动时支持聚焦区域的选择; -->center
</focusRegion>
<focusScene>
  <!--ro, opt, enum, 聚焦场景, subType:string, [lowIllumination#低照度,normal#普通], desc:该聚焦场景与图像场景(/ISAPI/Image/channels/<channelID>/mountingScenario)之间逻辑关系: 1、若聚焦场景为普通模式时与图像参数场景保持一致;2、聚焦场景为其他模式时,场景参数以聚焦场景的为准-->normal
</focusScene>
<timeFocusEnabled>
  <!--ro, opt, bool, 定时聚焦使能, dep:or,{$.FocusConfiguration.focusStyle,eq,AUTO},{$.FocusConfiguration.focusStyle,eq,SEMIAUTOMATIC}, desc:仅在聚焦类型为自动和半自动时支持配置-->true
</timeFocusEnabled>
<timeFocus>
  <!--ro, opt, int, 定时聚焦时间间隔, range:[1,1440], unit:min, unitType:时间, dep:and,{$.FocusConfiguration.timeFocusEnabled,eq,true}-->1
</timeFocus>

```

```
<focusDistance>
  <!--ro, opt, float, 聚焦推算距离, range:[0.30,99.99], step:0.01, unit:m, unitType:长度,
dep:and,{$.FocusConfiguration.focusStyle,eq,distanceCalculate}, desc:保留到小数点后2位-->0.35
</focusDistance>
<focusStrategy>
  <!--ro, opt, enum, 聚焦策略, subType:string, [general#普通,DTOF#DTOF], desc:根据视频或TOF
确定聚焦距离-->DTOF
</focusStrategy>
</FocusConfiguration>
```

## 2.1.29 获取预测聚焦参数

### URL

GET /ISAPI/Image/channels/<channelID>/PredictiveFocusParams?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	通道号

### 请求报文

无

### 响应报文

```
{
/*req, object, 预测聚焦, range:[,], desc:*/
  "enabled": false,
/*req, boolean, 使能, const:, desc:*/
  "trainingPoints": 50,
/*opt, integer, 训练点数, const:, enum:[], range:[0,100], multipleOf:, step:, unit:,
desc:预测聚焦自学习需要的最少点数*/
  "confidenceLevel": 50
/*opt, integer, 置信度, const:, enum:[], range:[0,100], multipleOf:100, step:,
unit:, desc:实际聚焦对预测聚焦结果的信任度*/
}
```

## 2.1.30 设置预测聚焦参数

### URL

PUT /ISAPI/Image/channels/<channelID>/PredictiveFocusParams?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	通道号



## 请求报文

```
{
/*req, object, 预测聚焦, range:[,], desc:*/
    "enabled": false,
    /*req, boolean, 使能, const:, desc:*/
    "trainingPoints": 50,
    /*opt, integer, 训练点数, const:, enum:[], range:[0,100], multipleOf:, step:, unit:,
desc:预测聚焦自学习需要的最少点数*/
    "confidenceLevel": 50
    /*opt, integer, 置信度, const:, enum:[], range:[0,100], multipleOf:100, step:,
unit:, desc:实际聚焦对预测聚焦结果的信任度*/
}
```

## 响应报文

```
{
/*req, object, , range:[,], desc:*/
    "statusCode": 1,
    /*opt, integer, 状态码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
    "statusString": "OK",
    /*opt, string, 状态描述, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
    "subStatusCode": "ok",
    /*opt, string, 子状态码, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
    "errorCode": 1,
    /*opt, integer, 错误码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:当statusCode不为1时，与subStatusCode对应*/
    "errorMsg": "ok",
    /*opt, string, 错误信息, const:, range:[,], enum:[], format:, mediaType:, pattern:,
unit:, RAP:, desc:当statusCode不为1时，必须返回，解释信息在协议约束中，允许设备在后续的版本迭代中，进行
优化丰富提升（不限制死）*/
    "MErrCode": "0x00000000",
    /*opt, string, 功能模块化错误, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:通用错误码都在此字段范围中*/
    "MErrDevSelfEx": "0x00000000"
    /*opt, string, 功能模块化错误码, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:此字段是MErrCode自定义模块化错误码的扩展字段，自定义模块化错误码一共有
MErrCode 的后20bit和此字段的32bit*/
}
```

## 2.1.31 获取预测聚焦参数能力

### URL

GET /ISAPI/Image/channels/<channelID>/PredictiveFocusParams/capabilities?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	--

## 请求报文

无

## 响应报文

```
{
/*req, object, 获取预测聚焦参数能力, range:[,], desc:*/
  "PredictiveFocusParamsCap":{
    /*opt, object, 预测聚焦参数能力, range:[,], desc:对应功能:GET/PUT
/ISAPI/Image/channels/<channelID>/PredictiveFocusParams?format=json*/
    "enabled":{
      /*req, object, 使能取值范围, range:[,], desc:*/
      "@opt":[
        /*req, array, 使能枚举列表, range:[1,2], uniqueItems:true, desc:*/
        true,false
        /*req, boolean, 枚举值, const:, desc:*/
      ],
      "@def": false
      /*opt, boolean, 默认值, const:, desc:*/
    },
    "trainingPoints":{
      /*opt, object, 训练点数取值范围, range:[,], desc:预测聚焦自学习需要的最少点数*/
      "@def": 50,
      /*opt, integer, 默认值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
      "@min": 0,
      /*req, integer, 最小值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
      "@max": 100
      /*req, integer, 最大值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
    },
    "confidenceLevel":{
      /*opt, object, 置信度取值范围, range:[,], desc:实际聚焦对预测聚焦结果的信任度*/
      "@def": 50,
      /*opt, integer, 默认值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
      "@min": 0,
      /*req, integer, 最小值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
      "@max": 100
      /*req, integer, 最大值, const:, enum:[], range:[,], multipleOf:,
step:, unit:, desc:*/
    }
  }
}
```

## 2.1.32 重置预测聚焦训练

URL

PUT /ISAPI/Image/channels/<channelID>/ResetPredictiveFocus?format=json

查询参数

参数名称	参数类型	描述
channelID	string	通道号

请求报文

无

响应报文

```
{
/*req, object, , range:[,], desc:*/
  "statusCode": 1,
  /*opt, integer, 状态码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
  "statusString": "OK",
  /*opt, string, 状态描述, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
  "subStatusCode": "ok",
  /*opt, string, 子状态码, const:, range:[1,64], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
  "errorCode": 1,
  /*opt, integer, 错误码, const:, enum:[], range:[,], multipleOf:, step:, unit:,
desc:当statusCode不为1时，与subStatusCode对应*/
  "errorMsg": "ok",
  /*opt, string, 错误信息, const:, range:[,], enum:[], format:, mediaType:, pattern:,
unit:, RAP:, desc:当statusCode不为1时，必须返回,解释信息在协议约束中，允许设备在后续的版本迭代中，进行
优化丰富提升（不限制死）*/
  "MErrCode": "0x00000000",
  /*opt, string, 功能模块化错误, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:通用错误码都在此字段范围中*/
  "MErrDevSelfEx": "0x00000000"
  /*opt, string, 功能模块化错误码, const:, range:[,], enum:[], format:, mediaType:,
pattern:, unit:, RAP:, desc:此字段是MErrCode自定义模块化错误码的扩展字段，自定义模块化错误码一共有
MErrCode 的后20bit和此字段的32bit*/
}
```

2.1.33 获取指定通道聚焦状态

URL

GET /ISAPI/Image/channels/<channelID>/focusConfiguration/status?format=json

查询参数

参数名称	参数类型	描述
channelID	string	

请求报文

无

响应报文

```
{
  "focusStatus": "success",
  /*ro, req, enum, 聚焦状态, subType:string, [success#成功,failed#失败,working#进行中]*/
  "motorMovementType": "manual"
  /*ro, opt, enum, 电机动作类型, subType:string, [manual#手动,auto#自动,armingLocation#
预置位]*/
}
```

2.1.34 获取指定通道云台缩放焦点信息

URL

GET /ISAPI/PTZCtrl/channels/<channelID>/zoomFocus

查询参数

参数名称	参数类型	描述
channelID	string	

请求报文

无

响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ZoomFocus xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, opt, object, 缩放焦点, attr:version{req, string, 协议版本}-->
  <pqrsZoom>
    <!--ro, opt, int, pqrs缩放-->1
  </pqrsZoom>
  <mnstFocus>
    <!--ro, opt, int, mnst焦点-->1
  </mnstFocus>
</ZoomFocus>
```

2.1.35 获取指定通道聚焦视频模式参数能力

URL

GET /ISAPI/System/Video/inputs/channels/<channelID>/focusVideoMode/capabilities?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

无

响应报文

```
{
  "FocusVideoModeCap": {
    /*ro, opt, object, 聚焦视频模式参数*/
    "mode": {
      /*ro, req, object, 可定时切换的智能资源, desc:normal(正常视
      频),infraredLightMode(红外视频),visibleLightMode (可见光视频) */
      "@opt": ["normal", "visibleLightMode", "infraredLightMode"],
      /*ro, req, array, 取值范围, subType:string*/
      "#text": "normal"
      /*ro, opt, string, 示例值*/
    }
  }
}
```

2.1.36 配置指定通道聚焦视频模式参数

URL

PUT /ISAPI/System/Video/inputs/channels/<channelID>/focusVideoMode?format=json

查询参数

参数名称	参数类型	描述
channelID	string	--

请求报文

```
{
  "FocusVideoMode": {
    /*opt, object, 聚焦视频模式参数*/
    "mode": "normal"
    /*req, enum, 模式, subType:string, [normal#正常视频,infraredLightMode#红外视
    频,visibleLightMode#可见光视频] */
  }
}
```

响应报文

```

{
  "statusCode": 1,
  /*ro, opt, int, 状态码, desc:无法用1表示时（1表示成功且无特殊状态）必须返回*/
  "statusString": "OK",
  /*ro, opt, string, 状态描述, range:[1,64], desc:无法用OK表示时（OK表示成功且无特殊状态）必须返回*/
  "subStatusCode": "ok",
  /*ro, opt, string, 子状态码, range:[1,64], desc:无法用ok表示（ok表示成功且无特殊状态）必须返回*/
  "errorCode": 1,
  /*ro, opt, int, 错误码, desc:当statusCode不为1时, 与subStatusCode对应*/
  "errorMsg": "ok",
  /*ro, opt, string, 错误信息, desc:当statusCode不为1时, 必须返回,解释信息在协议约束中, 允许设备在后续的版本迭代中, 进行优化丰富提升（不限制死）*/
  "MErrCode": "0x00000000",
  /*ro, opt, string, 功能模块化错误, desc:通用错误码都在此字段范围中*/
  "MErrDevSelfEx": "0x00000000"
  /*ro, opt, string, 功能模块化错误码, desc:此字段是MErrCode自定义模块化错误码的扩展字段, 自定义模块化错误码一共有MErrCode 的后20bit和此字段的32bit*/
}

```

## 2.1.37 获取指定通道聚焦视频模式参数

### URL

GET /ISAPI/System/Video/inputs/channels/<channelID>/focusVideoMode?format=json

### 查询参数

参数名称	参数类型	描述
channelID	string	--

### 请求报文

无

### 响应报文

```

{
  "FocusVideoMode": {
    /*ro, opt, object, 聚焦视频模式参数*/
    "mode": "normal"
    /*ro, req, enum, 模式, subType:string, [normal#正常视频,infraredLightMode#红外视频,visibleLightMode#可见光视频]*/
  }
}

```

## 2.1.38 获取ABF控制参数

### URL

GET /ISAPI/System/Hardware/ABF

### 查询参数

无

### 请求报文

无

### 响应报文

```
<?xml version="1.0" encoding="UTF-8"?>

<ABF xmlns="http://www.isapi.org/ver20/XMLSchema" version="2.0">
  <!--ro, req, object, ABF节点, attr:version{req, string, 协议版本}-->
  <enabled>
    <!--ro, req, bool, 使能-->true
  </enabled>
</ABF>
```