

Pilot 手机通讯协议

1 基本规则

1.1 通信协议

- Udp + Json
- Http + Json

Udp在搜索设备时使用，除此之外，其他接口都是使用http。

在使用Http请求时必须添加请求头

请求头 `Content-Type` 类型必须设置成 `application/x-www-form-urlencoded`

请求头version，参照通信接口中的请求头参数说明，接口有不同的version，按照说明，设置相应的version值。

Http请求参数格式(以连接相机接口为例)

```
1 {
2     "name": "camera.startSession",
3     "parameters": string
4 }
5
6 // string为下面接口中的输入参数,参数格式如下(以camera.startSession为例)
7
8 {
9     "parameters" : {
10         "alone" : true,
11         "sessionId" : "53c61259-373e-4eed-975c-5d5500be176b",
12         "timeout" : 1800
13     }
14 }
15 //把上面的json格式转换成字符串(string)
16
17 //完整例子
18 {
19     "name": "camera.startSession",
20     "parameters": "{\"parameters\" : {\"timeout\" : 1800,\"alone\" : true,\"sessionId\" : \"d71b96cc-dd49-4a00-b2eb-ffa0a9f04dc7\"}}\"
21
22 }
```

备注：接口文档里“输入参数”不为空时，需要带“parameters”且是json格式字符串，输入参数“无”时，不需要带parameters

http超时时间建议15s（除camera.startSession，这个可自行定义）

1.2 请求HOST

<http://x.x.x.x:8080>

x.x.x.x代表当前Pilot相机的IP地址，8080代表相机监听的端口号。

1.3 字段说明

字段	字段类型	描述
name	string	请求接口名称
state	string	接口状态1) done:代表已完成; 2) inProgress:正在进行; 3) error:访问出错
id	string	接口唯一id, Pilot相机中name与id相同
error		错误详情信息json节点
code	string	错误代码
message	string	错误详情
parameters		输入参数json节点(实际输入参数根据每个接口需要参数不同而定)
results		返回结果json节点(实际返回参数根据每个接口返回所需要的结果而定)

注意：对于返回的state字段为inProgress都需要再次调用查询状态接口(/osc/commands/status)统一查询返回结果。

2 通信接口：

2.1 搜索设备

功能：

建立与Pilot通信之前，首先要在同一个局域网内搜索到Pilot相机，通过此接口找到可利用的Pilot相机。相机Udp端口号为10000。

调用方式： Udp+json

输入参数：

字段	字段类型	描述
port	int	手机端端口号(10000)
sessionId	string	上一次创建连接时的会话id(即上次连接设备时，返回的sessionId,没连过设备，给空)
version	string	3.1

Udp参数格式：

```
1 {
2     "parameters" : {
3         "port" : 10000,
4         "sessionId" : "559d26f2-081f-43f5-8f93-35788a9321b4",
5         "version" : "3.1"
6     }
7 }
```

输出参数：

字段	字段类型	描述
ip	string	相机端ip地址
port	int	相机端端口号
deviceName	string	相机名称
alreadyConnected	boolean	相机是否被连接,boolean类型

返回格式：

```
1 {
2     "parameters": {
3         "ip": "192.168.4.85",
4         "port": 10000
5     },
6     "results": {
7         "ip": "",
8         "port": 0,
9         "deviceName": "",
10        "alreadyConnected": false
11    },
12 }
```

2.2 连接相机

功能：

连接Pilot相机

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera.startSession

请求头参数(version)： 3.1

输入参数：

字段	字段类型	描述
timeout	int	连接超时时间
sessionId	string	会话id
alone	boolean	心跳超时，会话是否需要保持，true：心跳超时，（相机正在录像、漫游、直播时）会话可保持，即不停止相机当前的工作；false：心跳超时，关闭会话。
language	String	接口返回的数据，根据设置的语言返回接口 英文:en ,中文:zh

输出参数：

字段	字段类型	描述
sessionId	string	会话id
timeout	int	连接超时时间
channel	string	渠道版本号(ali,standard)
mode	string	预览类型 “photo”代表拍照模式 “video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_street_view”代表街景视频 “video_time_lapse”代表延时摄影 “photo_tours”代表漫游拍照
status	boolean	相机工作状态，如果当前模式是录像，则返回录像的状态，反之，则代码拍照状态true代表正在录像或者拍照工作中 false代表空闲状态
firmwareVersion	string	相机app版本号
exposureType	int	曝光模式类型：0代表自动模式，1代表手动模式
modeWay	String	模式操作方式。photo(拍照)、roam(漫游)时可为: NULL(空值)，值为“qry”时(隐藏拍摄者)代表正在拍照。其他模式不支持此字段

返回格式：

```
1 {
2     "id": "camera.startSession",
3     "name": "camera.startSession",
4     "results": {
5         "channel": "standard",
6         "exposureType": 0,
7         "firmwareVersion": "4.3.0",
8         "sessionId": "1bb27add-04e5-4458-8147-3f57b3bb5e17",
9         "status": false,
10        "timeout": 1800
11    },
12    "state": "done"
13 }
```

2.3 建立心跳机制

功能：

相机与手机通讯建立连接之后，要与相机保持一个心跳机制Pilot心跳机制为手机端连接成功之后马上请求心跳接口，相机端收到手机端传来的参数之后马上返回给手机端，手机端收到之后再回传，周而复始。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getHeart

请求头参数(version)： 4.7

输入参数： 无

输出参数：

字段	字段类型	描述
battery	string	电池电量
latitude	double	纬度
longitude	double	经度
lowMemory	boolean	内存状态true表示低内存false表示充足
pushStream	boolean	推流状态true表示正在推流中false没有推流
tick	int	倒计时，打开设置倒计时，相机实时返回。进行“拍摄隐藏拍摄者模式”拍照时，tick总时长为“第一张图片倒计时”+“第一张图片倒计时”，拍第一张照片时tick暂停
videoTime	string	录像时间
tourPrepare	int	漫游标定状态，漫游需标定后使用。0：漫游未标定；1：漫游已标定。
roamStateDetail	int	新版漫游状态详情：0-未初始化 1-正在初始化 2-正在跟踪 3-丢失跟踪（相机5.6.0以上使用）
checkTime	boolean	延时摄影模式下，录制时间是否过短。false时，停止录像，不会生成录像文件
sdCritical	long	储存空间临界值
sdTotalSize	Long	储存空间总大小
sdAvailableSize	Long	储存空间剩余可用
temperature	Int	机器温度(适用于Pilot One户外版)，温度大于80度时，相机Cpu温度过高
taskId	String	当前相机进行中的（首选）任务id，目前仅支持拍照任务

返回格式：

```
1  {
2      "id": "camera._getHeart",
3      "name": "camera._getHeart",
4      "results": {
5          "battery": "97",
6          "latitude": 0.0,
7          "longitude": 0.0,
8          "lowMemory": false,
9          "pushStream": false,
10         "tick": 0,
11         "videoTime": "00:00:00"
12     },
13     "state": "done"
14 }
```

2.4 关闭相机连接

功能：

关闭与相机的连接。

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera.closeSession

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
sessionId	string	连接唯一标识

输出参数： 无

返回格式：

```
1 {  
2     "id": "camera.closeSession",  
3     "name": "camera.closeSession",  
4     "state": "done"  
5 }
```

2.5 开启相机预览

功能：

开启相机预览。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._startPreview

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
targetId	Int	Camera(相机预览): 1 , Live(直播预览): 3

输出参数： 需要调用查询状态返回结果

返回格式：

```
1 {  
2     "id": "camera._startPreview",  
3     "name": "camera._startPreview",  
4     "state": "InProgress"  
5 }
```

2.6 关闭相机预览

功能：

关闭相机的预览。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._stopPreview

请求头参数(version)： 3.0

输入参数： 无

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._stopPreview ",
3     "name": "camera._stopPreview ",
4     "state": "done"
5 }
```

2.7 切换相机工作模式

功能：

切换相机工作的模式(如图片、未拼接视频、实时拼接视频、街景、漫游、延时摄影等模式)

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._changeResolutionEx

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
type	string	相机工作类型 “photo”代表拍照模式 “video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_street_view”代表街景视频 “video_time_lapse”代表延时摄影(新增) “photo_tours”代表漫游拍照

输出参数： 需要调用查询状态结果返回

返回格式：

```
1 {
2     "id": "camera._changeResolutionEx",
3     "name": "camera._changeResolutionEx",
4     "state": "InProgress"
5 }
```

2.8 查询命令

功能：

查询对应请求命令接口进度。

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/status>

命令ID： 需要查询结果的id

字段	字段类型	描述
id	string	命令ID：

请求头参数(version)： 3.0

输入参数：无

输出参数： 需要调用查询状态结果返回

字段	字段类型	描述
dateTimeZone	string	拍照时间(2018-09-22 14:15:38)
fileUrl	string	文件地址
width	int	照片宽
height	int	照片高
isProcessed	boolean	—
lat	double	—
lng	double	—
name	string	文件名称
size	size	文件大小
stitch	boolean	是否拼接, boolean类型
thumbnail	string	缩略图地址

返回格式：以拍照查询结果为例

```
1 {
2   "id": "camera.takePicture",
3   "name": "camera.takePicture",
4   "results": {
5     "dateTimeZone": "2018-09-22 14:15:38",
6     "fileUrl": "/osc/getFile?/storage/emulated/0/DCIM/Pisoft/180922_141536982.jpg",
7     "height": 3072,
8     "isProcessed": false,
9     "lat": 0.0,
10    "lng": 0.0,
11    "name": "180922_141536982.jpg",
12    "size": 7111125,
13    "stitch": false,
14    "thumbnail": "/osc/getFile?/storage/emulated/0/DCIM/Thumb/180922_141536982.jpg",
15    "width": 6144
16  },
17   "state": "done"
18 }
```

2.9 拍照

功能：

调用相机拍照。

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera.takePicture

请求头参数(version)：3.0

输入参数：无

输出参数：处于inProgress时(目前只用于隐藏拍摄者模式开启)

字段	类型	说明
_pIndex	int	拍照时，拍照索引（已拍张数）0代表第一张，1代表第二张
_tick	int	倒计时剩余时间(当_tick值为0时表示正在拍照中)

输出参数： 需要调用查询状态结果返回

字段	字段类型	描述
dateTimeZone	string	拍照时间(2018-09-22 14:15:38)
fileUrl	string	文件地址
width	int	照片宽
height	int	照片高
isProcessed	boolean	—
lat	double	—
lng	double	—
name	string	文件名称
size	int	文件大小
stitch	boolean	是否拼接， boolean类型
thumbnail	string	缩略图地址

返回格式：

```
1 {
2     "id": "camera.takePicture",
3     "name": "camera.takePicture",
4     "results": {
5         "dateTimeZone": "2018-09-22 14:15:38",
6         "fileUrl": "/osc/getFile?/storage/emulated/0/DCIM/Pisoft/180922_141536982.jpg",
7         "height": 3072,
8         "isProcessed": false,
9         "lat": 0.0,
10        "lng": 0.0,
11        "name": "180922_141536982.jpg",
12        "size": 7111125,
13        "stitch": false,
14        "thumbnail": "/osc/getFile?/storage/emulated/0/DCIM/Thumb/180922_141536982.jpg",
15        "width": 6144
16    },
17    "state": "done"
18 }
```

2.10 下载文件

功能：

获取单张照片详情。

调用方式： http+json,Get传参

请求地址： http://x.x.x.x:8080/osc/getFile?/storage/emulated/0/DCIM/Thumb/180922_103443479.jpg

输入参数：无

输出参数 文件流

返回格式：无

2.11 获取文件(照片、视频)

功能：

获取相机文件

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera.listFiles

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
fileType	string	文件类型 1) image:图片类型 2) video:视频类型 3) all:图片与视频
startPosition	int	0
entryCount	int	1
maxSize	int	5
typeVersion	String	必须传： " <code>tours</code> "

输出参数： 需要调用查询状态结果返回

字段	字段类型	描述
dateTimeZone	string	拍照时间(2018-09-22 14:15:38)
fileUrl	string	文件地址
width	int	照片宽
height	int	照片高
isProcessed	boolean	—
lat	double	—
lng	double	—
name	string	文件名称
size	int	文件大小
stitch	boolean	是否拼接， boolean类型
thumbnail	string	缩略图地址
totalEntries	int	文件总数
entries		json节点

返回格式：

```

1  {
2      "id": "camera.listFiles",
3      "name": "camera.listFiles",
4      "results": {
5          "entries": [{
6              "dateTimeZone": "2018-09-22 14:15:38",
7              "fileUrl":
8              "/osc/getFile?/storage/emulated/0/DCIM/Pisoft/180922_141536982.jpg",
9              "height": 3072,
10             "isProcessed": false,
11             "lat": 0.0,
12             "lng": 0.0,
13             "name": "180922_141536982.jpg",
14             "size": 7111125,
15             "stitch": false,
16             "thumbnail": "/osc/getFile?/storage/emulated/0/DCIM/Thumb/180922_141536982.jpg",
17             "width": 6144
18         }],
19         "totalEntries": 5
20     },
21     "state": "done"
22 }

```

2.12 录像

功能：

调用相机录像。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera.startCapture

请求头参数(version)： 3.0

输入参数： 无

输出参数： 需要调用查询状态结果返回

返回格式：

```

1  {
2      "id": "camera.startCapture",
3      "name": "camera.startCapture",
4      "state": "InProgress"
5  }

```

2.13 停止录像

功能：

停止录像。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera.stopCapture

请求头参数(version): 3.0

输入参数: 无

输出参数: 需要调用查询状态结果返回

字段	字段类型	描述
dateTimeZone	string	录像时间(20190314T112118.000Z)
fileUrl	string	文件地址
width	int	照片宽
height	int	照片高
fps	string	帧率
lat	double	纬度
lng	double	经度
name	string	文件名称
size	int	文件大小
stitch	boolean	是否拼接, boolean类型 true表示已拼接 false表示未拼接
thumbnail	string	缩略图地址
codec	string	编码类型(H.264 , AAC)

返回格式:

```
1 {
2   "id": "camera.stopCapture",
3   "name": "camera.stopCapture",
4   "results": {
5     "codec": "H.264 , AAC",
6     "dateTimeZone": "20190314T112118.000Z",
7     "duration": "03:28",
8     "fileUrl": "/osc/getFile?/sdcard/DCIM/Videos/Unstitched/190314_191746118.sti",
9     "fps": "24",
10    "height": 2200,
11    "isProcessed": false,
12    "lat": 0.0,
13    "lng": 0.0,
14    "name": "190314_191746118.sti",
15    "size": 3231933437,
16    "stitch": false,
17    "thumbnail": "/osc/getFile?/storage/emulated/0/DCIM/Thumbs/190314_191746118.jpg",
18    "width": 3520
19  },
20   "state": "done"
21 }
```

2.14 删除文件

功能:

删除相机照片、视频文件。

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera.delete

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	描述
fileList	List	fileUrls 文件名:String isTour 是否是漫游文件：boolean

格式

```
1 {
2     "fileList" : [
3         {
4             "fileUrls" : "testPhoto.jpg",
5             "isTour" : 0
6         },
7         {
8             "fileUrls" : "testPhoto.jpg",
9             "isTour" : 0
10        },
11        {
12            "fileUrls" : "testPhoto.jpg",
13            "isTour" : 0
14        }
15    ]
16 }
17 {"parameters" : {"fileList" : [{"fileUrls" : "201026_105149651.mp4","isTour" : false}]}}
```

输出参数：无

返回删除成功:

```
1 {
2     "id": "camera.delete",
3     "name": "camera.delete",
4     "results": [],
5     "state": "done"
6 }
```

返回删除失败

```
1 {
2     "id": "camera.delete",
3     "name": "camera.delete",
4     "results": ["190314_194559664.jpg", "190314_194553689.jpg"],
5     "state": "done"
6 }
```

2.15 设置延时开关

功能：

延时拍照、录像。

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setDelayOpen

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
type	string	设置类型“photo”代表拍照模式 “video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_street_view”代表街景视频 “video_time_lapse”代表延时摄影 “photo_tours”代表漫游拍摄
isOpen	boolean	延时开关是否打开true 打开 false关闭

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setDelayOpen ",
3     "name": "camera._setDelayOpen ",
4     "state": "done"
5 }
```

2.16 设置延时

功能：

设置延时时间

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setDelay

请求头参数(version)： 3.0

输入参数：

字段	字段类型	描述
type	string	类型“photo”代表拍照模式 “video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_street_view”代表街景视频 “video_time_lapse”代表延时摄影 “photo_tours”代表漫游拍摄
delay	int	参数3、5、10、15

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setDelay",
3     "name": "camera._setDelay",
4     "state": "done"
5 }
```

2.17 获取相机延时配置

功能：

设置延时配置

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getDelay

请求头参数(version): 3.0

输入参数： 无

输出参数：

字段	字段类型	描述
delayphoto	int	照片延时时间
delayvideo	int	实时拼接视频延时时间
delayvideo_feye	int	未拼接视频延时时间
delayvideo_street_view	int	街景延时时间
delayvideo_time_lapse	int	延时拍摄延时时间
delaytours	Int	漫游延时时间(新增)

返回格式：

```
1 {
2     "id": "camera._getDelay",
3     "name": "camera._getDelay",
4     "results": {
5         "delayphoto": 5,
6         "delayvideo": 0,
7         "delayvideo_feye": 0,
8         "delayvideo_street_view": 3,
9         "delayvideo_time_lapse": 0,
10        "delaytours": 0
11    },
12    "state": "done"
13 }
```

2.18 设置分辨率

功能：

设置延时配置

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._setResolution

请求头参数(version): 3.0

输入参数:

字段	字段类型	描述
type	string	设置分辨率类型“photo”代表拍照模式 “video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_street_view”代表街景视频 “video_time_lapse”代表延时摄影 “photo_tours”代表漫游拍摄
resolution	string	分辨率 “photo”和“photo_tours”对应分辨率 “8192*4096”、“6144*3072” 4096*2048 3072*1536 2048*1024 “video”对应分辨率 7680*3840、5760*2880、3840*1920、1920*960 “video_feye”对应分辨率 7680*3840、7040*3520、5760*2880、3840*1920 “video_time_lapse” 对应分辨率 7680*3840、5760*2880、3840*1920、1920*960

输出参数: (需要调用查询状态结果返回)

返回格式:

```
1 {
2     "id": "camera._setResolution",
3     "name": "camera._setResolution",
4     "state": "InProgress"
5 }
```

2.19 获取分辨率配置

功能:

获取相机分辨率配置

调用方式: http+json,Post传参

请求地址: <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._getResolution

请求头参数(version): 3.0

输入参数: 无

输出参数:

字段	字段类型	描述
resolutionphoto	string	图片分辨率
resolutionvideo	string	实时视频类型分辨率
resolutionvideo_feye	string	未拼接视频类型分辨率
resolutionvideo_lapse_time	string	延时摄影类型分辨率
resolutiontours	String	漫游分辨率

返回格式:


```
1 {
2     "id": "camera._getResolution",
3     "name": "camera._getResolution",
4     "results": {
5         "resolutionphoto": "6144*3072",
6         "resolutionvideo": "1920*960",
7         "resolutionvideo_feye": "7040*3520",
8         "resolutionvideo_lapse_time": "7680*3840"
9     },
10    "state": "done"
11 }
```

2.20 设置专业设置

功能：

设置相机专业设置参数

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setProfessionalEx

请求头参数(version)： 5.0.0

输入参数：

字段	字段类型	描述
type	string	专业设置值类型 iso 感光度 ev 曝光补偿 distance 拼接距离 exposure 曝光时间 wb白平衡
value	int	对应不同专业设置，设置相应类型的值iso 感光度(0到6) ev 曝光补偿(-4 到 4) distance 拼接距离(0 到 100) exposure 曝光时间 (0到6) wb白平衡(0到4)
prold	int	图片、未拼接、实时、街景类型不同类型 0图片 1未拼接 2实时拼接 3街景 4延时摄影 5漫游
mode	int	模式类型 1表示手动模式 0表示自动模式

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setProfessionalEx",
3     "name": "camera._setProfessionalEx",
4     "state": "done"
5 }
```

2.21 获取专业设置

功能：

获取相机专业设置参数

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getProfessionalEx

请求头参数(version)： 5.0.0

输入参数：

字段	字段类型	描述
prold	int	图片、未拼接、实时、街景类型不同类型 0图片 1未拼接 2实时拼接 3街景 4延时摄影 5 漫游

输出参数：

字段	字段类型	描述
distance	int	拼接距离
ev	int	曝光补偿
exposure	int	曝光时间
exposureType	int	曝光类型0自动模式1手动模式
iso	int	感光度
manualISO	int	手动模式iso值
prold	int	图片、未拼接、实时、街景类型不同类型0图片 1未拼接 2实时拼接 3街景 4延时摄影 5 漫游

返回格式：

```
1 {
2     "id": "camera._getProfessionalEx",
3     "name": "camera._getProfessionalEx",
4     "results": {
5         "distance": 100,
6         "ev": 0,
7         "exposure": 1,
8         "exposureType": 1,
9         "iso": 0,
10        "manualISO": 3,
11        "proId": 0
12    },
13    "state": "done"
14 }
```

2.22 图片参数设置

功能：

设置图片参数

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setPhotoArgs

请求头参数(version)： 5.0.0

输入参数：

字段	字段类型	描述
type	string	图片参数类型 <code>hdr</code> 是否打开hdr <code>gyroscope</code> 是否开启画面稳定 <code>flow</code> 是否开启光流
mode	int	0:图片 1:鱼眼 2:拼接 3:谷歌街景 4:延时摄影 5:漫游
value	bool	

输出参数：无

返回格式：

```
1 {
2   "id": "camera._setPhotoArgs",
3   "name": "camera._setPhotoArgs",
4   "state": "done"
5 }
```

2.23 获取图片参数

功能：

获取图片参数

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._getPhotoArgs

请求头参数(version)：5.0.0

输入参数：无

输出参数：

字段	字段类型	描述
gyroscope	boolean	是否开启画面稳定
flow	boolean	是否开启光流
hdr	boolean	是否开启hdr
hdrState	int	HDR 选中模式

注明： 0： auto 1:晴天sunny 2:阴天cloudy 3:室内indoor

返回格式：

```
1 {
2   "id": "camera._getPhotoArgs",
3   "name": "camera._getPhotoArgs",
4   "results": {
5     "gyroscope": true,
6     "flow": false,
7     "hdr": true
8   },
9   "state": "done"
10 }
```

2.24 获取平台信息

功能：

获取直播某个平台信息

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._getLivePlatformInfo

请求头参数(version): 4.11.0

输入参数:

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: self

输出参数:

字段	字段类型	描述
platformId	Int	平台id
account	String	平台账号名
autoDefinition	Boolean	是否打开自动清晰度
definition	String	清晰度(分辨率)
privacy	Int	平台隐私
title	String	平台标题
login	Boolean	平台是否登录
facebookPubType	Int	时间线 (个人主页) : 0; 主页: 1; 小组: 2
facebookPubId	String	id
facebookPubName	String	Name

平台对应隐私:

Facebook: 1-仅自己 (Self) 3-所有朋友(All friend) 4-公开(Public)

Youtube: 0-私享(Priate) 1-公开(Public) 2-不公开(Unlisted)

Weibo: 0-不公开(Priate) 1-公开(Public)

返回格式:

```
1 {
2   "id": "camera._getLivePlatformInfo ",
3   "name": "camera._getLivePlatformInfo ",
4   "results": {
5     "platformId": 2,
6     "account": "",
7     "autoDefinition": false,
8     "definition": "1920*960",
9     "privacy": 1,
10    "title": "",
11    "login": false
12  },
13   "state": "done"
14 }
```

2.25 设置平台标题

功能:

设置平台标题

调用方式: http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setLiveTitle

请求头参数(version)：4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	Int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: self
title	String	标题内容	

输出参数：无

```
1 {
2     "id": "camera._setLiveTitle ",
3     "name": "camera._setLiveTitle ",
4     "state": "done"
5 }
```

返回格式：

2.26 设置平台隐私

功能：

设置平台隐私

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setLivePrivacy

请求头参数(version)：4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	Int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: self
privacy	Int	平台隐私	平台隐私对应参数参照一下规则

平台对应隐私：

Facebook: 1-仅自己 (Self) 3-所有朋友(All friend) 4-公开(Public)

Youtube: 0-私享(Public) 1-公开(Public) 2-不公开(Unlisted)

Weibo: 0-不公开(Public) 1-公开(Public)

输出参数：无

返回格式：

```
1 {
2     "id": "camera._setLivePrivacy ",
3     "name": "camera._setLivePrivacy ",
4     "state": "done"
5 }
```

2.27 获取平台支持的分辨率列表

功能：

获取每个平台支持的分辨率列表

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getLiveDefinitionList

请求头参数(version)： 4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: self
language	String	语言类型	Zh_cn,En_cn 默认英文

输出参数：

字段	字段类型	描述
platformId	Int	平台id
definition	String	选中的分辨率
definitionStrList	String[]	分辨率文案列表
definitionList	String[]	分辨率列表

返回格式：

```
1 {
2     "id": "camera._getLiveDefinitionList ",
3     "name": "camera._getLiveDefinitionList ",
4     "results": {
5         "platformId": 2,
6         "definition": "3648*2280",
7         "definitionStrList": ["8k", "4k", "HD", "SD", "Fluent"],
8         "definitionList": ["3648*2280", "3840*1920", "2560*1280", "1920*960", "1280*640"]
9     },
10    "state": "done"
11 }
```

2.28 设置平台分辨率

功能：

设置分辨率

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setLiveDefinition

请求头参数(version)： 4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	Int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: pi
definition	String	分辨率	如:3648*2280

输出参数：无

返回格式：

```

1 {
2     "id": "camera._setLiveDefinition ",
3     "name": "camera._setLiveDefinition ",
4     "state": "done"
5 }
```

2.29 获取每个平台每个分辨率对应选中的码率

获取每个平台支持的分辨率列表

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._getLiveSelectBitrateList

请求头参数(version)：4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: self
language	String	语言类型	Zh_cn,En_cn 默认英文

输出参数：

字段	字段类型	描述	标准参数
platformId	Int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: pi
k8BitRation	String	60 Mbps	输出8k选中的码率
k4BitRation	String	12 Mbps	输出4k选中的码率
highBitRation	String	8 Mbps	输出HD选中的码率
standardBitRation	String	5 Mbps	输出SD选中的码率
smoothBitRation	String	3 Mbps	输出Fluent选中的码率
definitionStrList	String[]	分辨率文案列表	
definitionList	String[]	分辨率列表	

返回格式：

```

1 {
2     "id": "camera._getLiveBitrateList ",
3     "name": "camera._getLiveBitrateList ",
4     "results": {
5         "platformId": 2,
6         "selectBitrateList": ["80 Mbs", "60 Mbs"],
7         "definitionStrList": ["8k", "4k", "HD", "SD", "Fluent"],
8         "definitionList": ["3648*2280", "3840*1920", "2560*1280", "1920*960", "1280*640"]
9     },
10    "state": "done"
11 }

```

2.30 获取每个平台每个分辨率对应的码率列表

获取每个平台对应每个分辨率支持的码率列表

调用方式: http+json,Post传参

请求地址: <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._getLiveBitrateList

请求头参数(version): 4.11.0

输入参数:

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp 1: weibo 2: facebook 3: youtube 4: pi
definition	String	分辨率	如:3648*2280

输出参数:

字段	字段类型	描述
platformId	Int	平台id
bitrate	String	选中的码率
bitrateList	String[]	码率列表

返回格式:

```

1 {
2     "id": "camera._getLiveBitrateList ",
3     "name": "camera._getLiveBitrateList ",
4     "results": {
5         "platformId": 2,
6         "bitrate": "80 Mbps",
7         "bitrateList": ["80 Mbps", "70 Mbps", "60 Mbps", "50 Mbps", "40 Mbps"]
8     },
9     "state": "done"
10 }

```

2.31 设置码率

调用方式: http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setLiveBitrate

请求头参数(version)：4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp;1: weibo;2: facebook;3: youtube;4: self
definition	String	分辨率	如:3648*2280
bitrate	String	码率	如：80 Mbps

输出参数：无

返回格式：

```
1 {
2     "id": "camera._setLiveBitrate ",
3     "name": "camera._setLiveBitrate ",
4     "state": "done"
5 }
```

2.32 开始直播

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setLiveStart

请求头参数(version)：4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	int	平台id	0: rtmp;1: weibo;2: facebook;3: youtube;4: self
type	int	直播类型	根据分辨率-平台ID 判断是否是8K或者其他 0-8k 其他值（其他） type 推流类型 0: 8k 1:普通直播
language	String	语言类型	Zh_cn,En_cn 默认英文

注意：type 推流类型 0: 8k 1:普通直播

返回格式：

输出参数：无

返回格式：

```
1 {
2     "id": "camera._setLiveStart ",
3     "name": "camera._setLiveStart",
4     "state": "done"
5 }
```

2.33 停止直播

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setLiveStop

请求头参数(version): 4.11.0

输入参数： 无

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setLiveStop ",
3     "name": "camera._setLiveStop ",
4     "state": "done"
5 }
```

2.34 获取专业设置

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getProfessionalEx

请求头参数(version): 5.0.0

输入参数：

字段	字段类型	描述	标准参数
prold	int	平台id	传0就行

输出参数：

字段	字段类型	描述	标准参数
ev	Int	Ev值	
iso	Int	Iso值	
distance	Int	拼接距离	

返回格式：

```
1 {
2     "id": "camera._getProfessionalEx ",
3     "name": "camera._getProfessionalEx ",
4     "results": {
5         "ev": 0,
6         "iso": 0,
7         "distance ":100
8     },
9     "state": "done"
10 }
```

2.35 设置专业设置

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setProfessionalEx

请求头参数(version)： 5.0.0

输入参数：

字段	字段类型	描述	标准参数
type	String	类型	ev 、 distance 、 iso
value	Int	值	ev:0-8 distance :0 - 100 iso:0-6

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setProfessionalEx ",
3     "name": "camera._setProfessionalEx ",
4     "state": "done"
5 }
```

2.36 通知接收到错误码

功能：

调用camera._setLiveStart接口，返回error时调用

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setLiveCode

请求头参数(version)： 4.11.0

输入参数：

字段	字段类型	描述	标准参数
code	Int	类型	0

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setLiveCode ",
3     "name": "camera._setLiveCode",
4     "state": "done"
5 }
```

2.37 心跳增加字段code

对应解析：

0:相机闲置状态

1:相机直播中

2:普通直播间创建失败

3:8k直播创建失败

<0: 直播错误

2.38 获取对应显示的平台ID

获取对应显示的平台ID

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getLiveShowPlatform

请求头参数(version): 4.11.0

输入参数： 无

输出参数： 无

输出参数： 无

字段	字段类型	描述	标准参数
platformId	Int	平台ID	0: rtmp 1: weibo 2: facebook 3: youtube 4: self

返回格式：

```
1 {
2     "id": "camera._getLiveShowPlatform",
3     "name": "camera._getLiveShowPlatform",
4     "results": {
5         "platformId": 2
6     },
7     "state": "done"
8 }
```

2.39 设置自动分辨率

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setLiveAutodefinition

请求头参数(version): 4.11.0

输入参数：

字段	字段类型	描述	标准参数
platformId	Int	类型	0: rtmp 1: weibo 2: facebook 3: youtube 4: self
autoDefinition	boolean	自动	false or true

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setLiveAutodefinition",
3     "name": "camera._setLiveAutodefinition",
4     "state": "done"
5 }
```

2.40 退出直播还原相机分辨率

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._resetResolution

请求头参数(version)： 4.11.0

输入参数： 无

输出参数： 无

```
1 {
2     "id": "camera._resetResolution",
3     "name": "camera._resetResolution",
4     "state": "done"
5 }
```

返回格式：

2.41 重试直播

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._restartLive

请求头参数(version)： 4.11.0

输入参数： 无

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._restartLive ",
3     "name": "camera._restartLive ",
4     "state": "done"
5 }
```

2.42 谷歌帧率列表以及选中的帧率

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getGoogleRateList

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	"video_street_view"代表街景视频

输出参数：

字段	字段类型	描述	标准参数
googleratelist	String[]	谷歌帧率列表	["1fps", "2fps", "7fps"]
rate	String	选中的帧率	1fps

返回格式：

```
1 {
2   "id": "camera._getGoogleRateList ",
3   "name": "camera._getGoogleRateList ",
4   "results": {
5     "googleratelist":["1fps","2fps","7fps"],
6     "rate": " 1fps"
7   },
8   "state": "done"
9 }
```

2.43 谷歌帧率设置

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setGoogleRate

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	“video_street_view”代表街景视频
rate	String	如： 7fps

输出参数： 无

返回格式：

```
1 {
2   "id": "camera._setGoogleRate",
3   "name": "camera._setGoogleRate",
4   "state": "done"
5 }
```

2.44 获取PilotSteady稳定(防抖)

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getOrientationState

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	“video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_time_lapse”代表延时摄影

输出参数：

字段	字段类型	描述	标准参数
orientation	boolean	画面朝向	False:固定 true:跟随相机朝向

返回格式：

```
1 {
2     "id": "camera._getOrientationState ",
3     "name": "camera._getOrientationState ",
4     "results":{
5         "orientation": false
6     },
7     "state": "done"
8 }
```

2.45 设置PilotSteady稳定(防抖)

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setOrientationState

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	“video”代表实时拼接模式 “video_feye”代表鱼眼模式 “video_time_lapse”代表延时摄影
orientation	boolean	false: 固定 true 跟随相机朝向

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setOrientationState",
3     "name": "camera._setOrientationState",
4     "state": "done"
5 }
```

2.46 获取缩时倍数列表和当前选中

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getTimeLapseTimeList

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	“video_time_lapse”代表延时摄影

输出参数：

字段	字段类型	引入版本	描述	标准参数
timeLapseTimeList	String[]	4.12.0	缩时倍数列表	["20x (0.6spf)"," 50x (1.7spf)",.....," 10000x (5.6mpf)"]
timeLapseTimes	String	4.12.0	选中的缩时倍数	"20x (0.6spf)"

返回格式：

```

1 {
2     "id": "camera._getTimeLapseTimeList ",
3     "name": "camera._getTimeLapseTimeList ",
4     "results": {
5         "timeLapseTimeList": ["20x (0.6spf)", " 50x (1.7spf)", ..... , " 10000x (5.6mpf)"],
6         "timeLapseTimes": " 20x (0.6spf)"
7     },
8     "state": "done"
9 }

```

2.47 设置缩时倍数

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera_setLapseTime

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	"video_time_lapse"代表延时摄影
timeLapseTimes	String	20x (0.6spf)

输出参数： 无

返回格式：

```

1 {
2     "id": "camera._setLapseTime",
3     "name": "camera._setLapseTime",
4     "state": "done"
5 }

```

2.48 获取声音参数

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera_getVoiceParams

请求头参数(version)： 4.12.0

输入参数： 无

输出参数：

字段	字段类型	描述	标准参数
micVoiceMax	int	麦克风最大值	100
micVoiceMin	int	麦克风最小值	0
micVoiceValue	int	麦克风当前值	70
speakerVoiceMax	int	扬声器最大值	15
speakerVoiceMin	int	扬声器最小值	0
speakerVoiceValue	int	扬声器当前值	3
pictureVoiceSwitch	boolean	拍照快门声开关	false
videoVoiceSwitch	boolean	录像快门声开关	false
liveVoiceSwitch	boolean	直播快门声开关	false

返回格式：

```
1 {
2   "id": "camera._getVoiceParams",
3   "name": "camera._getVoiceParams",
4   "results": {
5     "micVoiceMax":100,
6     "micVoiceMin":0,
7     "micVoiceValue":70,
8     "speakerVoiceMax":15,
9     "speakerVoiceMin":0,
10    "speakerVoiceValue":3,
11    "pictureVoiceSwitch":false,
12    "videoVoiceSwitch":false,
13    "liveVoiceSwitch":false,
14  },
15  "state": "done"
16 }
```

2.49 设置声音参数

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setVoiceParams

请求头参数(version)： 4.12.0

输入参数：

字段	字段类型	描述
type	String	“speaker”代表扬声器 “mic”代表麦克风 “picture”代表拍照快门声 “video”代表视频快门声 “live”代表直播快门声
value	int	speaker(int)、mic(int)、picture(int)、video(int)、live(int)

注意：speaker 和mic 一定要传 最小值和最大值之间的值

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setVoiceParams",
3     "name": "camera._setVoiceParams",
4     "state": "done"
5 }
```

2.50 获取风扇状态

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getFanParms

请求头参数(version): 4.12.0

输入参数： 无

输出参数：

字段	字段类型	描述
value	Int	1: auto 2:一直开启 0:一直关闭

返回格式：

```
1 {
2     "id": "camera._getFanParms",
3     "name": "camera._getFanParms",
4     "results": {
5         "value":1
6     },
7     "state": "done"
8 }
```

2.51 设置风扇状态

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setFanParms

请求头参数(version): 4.12.0

输入参数：

字段	字段类型	描述
value	Int	1: auto 2:一直开启 0:一直关闭

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._setFanParms",
3     "name": "camera._setFanParms",
4     "state": "done"
5 }
```

2.52 设置HDR 列表

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setPhotoHdrState

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	描述
hdrState	Int	0: auto 1:晴天 2:阴天 3:室内

注明：0: auto 1:晴天sunny 2:阴天cloudy 3:室内indoor

输出参数：无

返回格式：

```
1 {  
2     "id": "camera._setPhotoHdrState",  
3     "name": "camera._setPhotoHdrState",  
4     "state": "done"  
5 }
```

2.53 获取底部logo相关配置

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._getCameraWatermarkList

请求头参数(version)：4.14.0

输入参数：无

输出参数：

字段	字段类型	描述
watermarkState	Int	0:打开状态 1:关闭状态
logoSize	Int	进度值
imageSelect	String	获取当前设置的水印文件名
imgList	String[]	Logo 列表

2.54 设置logo开关

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setCameraWatermarkState

请求头参数(version)：4.14.0

输入参数：

字段	字段类型	描述
watermarkState	Int	0:打开状态 1:关闭状态

输出参数： 无

返回格式：

```

1  {
2      "id": "camera._setCameraWatermarkState",
3      "name": "camera._setCameraWatermarkState",
4      "state": "done"
5  }
```

2.55 设置logo 大小

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setCameraWatermarkSize

请求头参数(version)： 4.14.0

输入参数：

字段	字段类型	描述
watermarkSize	Int	最小： 0 最大 26

输出参数： 无

返回格式：

```

1  {
2      "id": "camera._setCameraWatermarkSize",
3      "name": "camera._setCameraWatermarkSize",
4      "state": "done"
5  }
```

2.56 设置选中logo

调用方式： http+json,Post传参

请求地址： <http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._setCameraWatermarkSelect

请求头参数(version)： 4.14.0

输入参数：

字段	字段类型	描述
imageSelect	String	文件名 ""空 字符串代表设置默认logo!

输出参数： 无

返回格式：

```
1 {  
2     "id": "camera._setCameraWatermarkSelect",  
3     "name": "camera._setCameraWatermarkSelect",  
4     "state": "done"  
5 }
```

2.57 logo文件上传(上传完成，需要调用2.61)

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera_uploadWatermarkFile

输入参数：

字段	字段类型	描述
fileName	文件名	文件名

输出参数：无

返回格式：无

2.58 获取提示灯相关参数

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera_getSettingLedParams

请求头参数(version)：4.14.0

输入参数：无

输出参数：

字段	字段类型	描述
ledPhoto	Int	1:打开状态 0:关闭状态
ledVideo	Int	1:打开状态 0:关闭状态
ledLive	Int	1:打开状态 0:关闭状态
ledBattery	Int	1:打开状态 0:关闭状态

返回格式：

2.59 设置提示灯相关参数

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera_setSettingLedParams

请求头参数(version)：4.14.0

输入参数：

字段	字段类型	描述
type	Int	0: 拍照 1:录像 2:直播 3:低电量
value	Int	1打开 0:关闭

输出参数：无

返回格式：

```

1  {
2      "id": "camera._setSettingLedParams",
3      "name": "camera._setSettingLedParams",
4      "state": "done"
5  }
```

2.60 查询上传图片接口

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._getUploadProgress

请求头参数(version)：4.14.0

输入参数：无

输出参数：

字段	字段类型	描述
uploadMsg	String	提示信息：上传成功 上传失败
imgPath	String	图片路径，成功：正确路径，失败：""

2.61 设置pushID

http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera._setLivePushId

请求头参数(version)：4.14.0

输入参数：

字段	字段类型	描述
pushId	String	自身推流 标题id:只接收 数字类型, "" 或者null 代表清空，也要传
platformId	int	0: rtmp 1: weibo 2: facebook 3: youtube 4: self

输出参数：无

返回格式：

```

1  {
2      "id": "camera._setSettingLedParams",
3      "name": "camera._setSettingLedParams",
4      "state": "done"
5  }
```

2.62 初始化相机预览流（PS：初始化预览流 退出底部logo时调用）

http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._initSurfacePreview

请求头参数(version)： 4.14.0

输入参数： 无

输出参数： 无

返回格式：

```
1 {
2     "id": "camera._initSurfacePreview",
3     "name": "camera._initSurfacePreview",
4     "state": "done"
5 }
```

2.63 获取直播画面稳定开关 和 画面朝向

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID： camera._getLiveOrientationState

请求头参数(version)： 4.14.0

输入参数： 无

输出参数：

字段	字段类型	描述
steadyValue	int	0:关闭 1:开
orientationValue	int	0: 固定 1:跟随相机朝向

返回格式：

```
1 {
2     "id": "camera._getLiveOrientationState",
3     "name": "camera._getLiveOrientationState",
4     "results":{
5         "steadyValue": 0.
6         "orientationValue":0
7     },
8     "state": "done"
9 }
```

2.64 设置直播画面稳定开关或者 画面朝向

调用方式： http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._setLiveOrientationState

请求头参数(version): 4.14.0

输入参数:

字段	字段类型	描述
type	int	0:pilotSteady稳定 1:画面朝向
value	int	PilotSteady稳定(0:关 1:开) 画面朝向: (0:固定 1:跟随相机朝向)

输出参数: 无

返回格式:

```
1 {
2     "id": "camera._setLiveOrientationState",
3     "name": "camera._setLiveOrientationState",
4     "state": "done"
5 }
```

2.65 开始漫游

调用方式: http+json,Post传参

请求地址: <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._startTour

请求头参数(version): 5.0.0

输入参数:

字段	字段类型	描述
tourName	String	漫游名字 空串、null、不传 将返回错误

返回格式:

```
1 {
2     "id": " camera._startTour",
3     "name": " camera._startTour",
4     "state": "done"
5 }
```

2.66 结束漫游

调用方式: http+json,Post传参

请求地址: <http://x.x.x.x:8080/osc/commands/execute>

命令ID: camera._stopTour

请求头参数(version): 5.0.0

输入参数:

返回格式:


```
1 {
2     "id": " camera._stopTour",
3     "name": " camera._stopTour",
4     "state": "done"
5 }
```

2.67 获取（Camera）选项

功能：

5.2.0 版本增加，获取（Camera）选项

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera.getOptions

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	描述
optionNames	String Array	需要获取值的选项名称数组。参加下方支持选项表格中选项名称。

输出参数：

字段	字段类型	描述
Options	map	参加下方支持选项表格中选项名称及选项值类型。

返回格式：

目前支持的选项有：

选项名称	值类型	描述
<code>_camera\$roam\$height</code>	String	漫游拍摄高度，浮点数值，取值范围[0,1000]，不在此范围内的设置值会处理为边界值。
<code>_camera\$video\$encode</code>	String	拼接视频编码，默认H.264。
<code>_camera\$video\$encodeSupport</code>	String Array	只读，拼接视频支持的编码。当前为["H.264","H.265"]
<code>_camera\$videoFishEye\$encode</code>	String	未拼接视频编码。
<code>_camera\$videoFishEye\$encodeSupport</code>	String Array	只读，未拼接视频支持的编码。当前为["H.264","H.265"]
<code>_camera\$videoStreetView\$encode</code>	String	街景视频编码，默认H.264。
<code>_camera\$videoStreetView\$encodeSupport</code>	String Array	只读，街景视频支持的编码。当前为["H.264","H.265"]
<code>_camera\$videoTimeLapse\$encode</code>	String	延时摄影视频编码，默认H.264。
<code>_camera\$videoTimeLapse\$encodeSupport</code>	String Array	只读，延时摄影视频支持的编码。当前为["H.264","H.265"]
<code>_camera\$photo\$qry\$able</code>	int	拍照模式，隐藏拍摄者模式下的开关值，可选值：0（关闭）、1（开启）（拍照模式）

<code>_camera\$photo\$qry\$p1_cd</code>	int	拍照模式，隐藏拍摄者模式下的第一张图片倒计时，可选值：3（3s）、5（5s）、10（10s）（拍照模式）
<code>_camera\$photo\$qry\$p2_cd</code>	int	拍照模式，隐藏拍摄者模式下的第二张图片倒计时，可选值：3（3s）、5（5s）、10（10s）（拍照模式）
<code>_camera\$roam\$qry\$able</code>	int	漫游模式，隐藏拍摄者模式下的开关值,可选值：0（关闭）、1（开启）（漫游模式）
<code>_camera\$roam\$qry\$p1_cd</code>	int	漫游模式，隐藏拍摄者模式下的第一张图片倒计时，可选值：3（3s）、5（5s）、10（10s）（漫游模式）
<code>_camera\$roam\$qry\$p2_cd</code>	int	漫游模式，隐藏拍摄者模式下的第二张图片倒计时，可选值：3（3s）、5（5s）、10（10s）（漫游模式）
<code>_camera\$video\$storagePartSupport</code>	Array<>	只读，实时视频支持的分段存储列表值， <code>Map<String,String></code> 中包含 <code>name</code> ：显示名称， <code>value</code> :实际值
<code>_camera\$video\$storagePart</code>	String	当前实时视频分段存储值，为空或空串时不分段存储
<code>_camera\$videoFishEye\$storagePartSupport</code>	Array<Map<String,String>>	只读，未拼接视频支持的分段存储列表值，具体格式参考 <code>_camera\$video\$storagePartSupport</code>
<code>_camera\$videoFishEye\$storagePart</code>	String	当前未拼接视频分段存储值，为空或空串时不分段存储
<code>_camera\$videoStreetView\$storagePartSupport</code>	Array<Map<String,String>>	只读，街景视频支持的分段存储列表值，具体格式参考 <code>_camera\$video\$storagePartSupport</code>
<code>_camera\$videoStreetView\$storagePart</code>	String	当前街景视频分段存储值，为空或空串时不分段存储
<code>_camera\$videoTimeLapse\$storagePartSupport</code>	Array<Map<String,String>>	只读，延时视频支持的分段存储列表值，具体格式参考 <code>_camera\$video\$storagePartSupport</code>
<code>_camera\$videoTimeLapse\$storagePart</code>	String	当前延时视频分段存储值，为空或空串时不分段存储

示例

```
1
2 //获取一些相机的值，如获取“实时视频模式”当前选中的视频编码、“拼接视频模式”支持的编码、“未拼接模式”当前选中的视频编
   码、“未拼接视频模式”支持的编码
3 {
4     "name" : "camera.getOptions",
5     "parameters" : {"parameters" : {"optionNames" : ["_camera$video$encode",
   "_camera$video$encodeSupport", "_camera$videoFishEye$encode",
   "_camera$videoFishEye$encodeSupport"]}}
6 }
```

2.68 设置（Camera）选项

功能：

5.2.0 版本增加，设置（Camera）选项

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：camera.setOptions

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	引入版本	描述
options	map	5.2.0	参加设置选项接口中支持选项表格中名称及值类型。

输出参数：无

示例

```
1 {
2     "name" : "camera.setOptions",
3     "parameters" : {"parameters" : {"options" : {"_camera$photo$qry$able" : 0}}}
4 }
```

2.69 获取（Live）选项(数据格式参考2.67示例)

功能：

5.2.0 版本增加，获取（Live）选项

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.getOptions

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	描述
optionNames	String Array	需要获取值的选项名称数组。参加下方支持选项表格中选项名称。

输出参数：

字段	字段类型	描述
options	map	参加下方支持选项表格中选项名称及选项值类型。

返回格式：

目前支持的选项有：

选项名称	值类型	描述
<code>_live\$rtmp\$encode</code>	String	rtmp编码，默认H.264。
<code>_live\$rtmp\$encodeSupport</code>	String Array	只读，rtmp支持的编码。当前为["H.264","H.265"]
<code>_live\$self\$encode</code>	String	自身推流编码，默认H.264。
<code>_live\$self\$encodeSupport</code>	String Array	只读，自身推流支持的编码。当前为 ["H.264","H.265"]

2.70 设置（Live）选项(数据格式参考2.68示例)

功能：

5.2.0 版本增加，设置（Live）选项

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.setOptions

请求头参数(version)：5.0.0

输入参数：

字段	字段类型	描述
options	map	参加设置选项接口中支持选项表格中名称及值类型。

输出参数：无

返回格式：

2.71 设置facebook直播类型(未开放)

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.facebook.setPubType

请求头参数(version)：可不设置

输入参数：

字段	字段类型	描述
pubType	Int	设置直播类型：时间线（个人主页）：0；主页：1；小组：2
pubId	String	id

输出参数：无

返回格式：

```
1 {  
2     "id": "_live.facebook.setPubType",  
3     "name": "_live.facebook.setPubType",  
4     "state": "done"  
5 }
```

2.72 获取facebook的pages(未开放)

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.facebook.pages

请求头参数(version)：可不设置

输入参数：无

输出参数：需要调用查询状态结果返回

字段	字段类型	描述
curId	String	当前显示的ID
list	Array	从Facebook获取到的page数据(返回结果中没有该字段，说明获取数据失败，可重试)
id	String	一个page的id
name	String	一个page的名字

返回格式：

```

1  {
2    "curId": "xxxx", // 当前选择的id
3    "list": [
4      {
5        "id": "xxxx",
6        "name": "显示名称"
7      }
8    ]
9  }
```

2.73 获取facebook的groups(未开放)

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.facebook.groups

请求头参数(version)：可不设置

输入参数：无

输出参数：需要调用查询状态结果返回

字段	字段类型	描述
curId	String	当前显示的ID
list	Array	从Facebook获取到的group数据(返回结果中没有该字段，说明获取数据失败，可重试)
id	String	一个group的id
name	String	一个group的名字

返回格式：

```

1  {
2    "curId": "xxxx", // 当前选择的id
3    "list": [
4      {
5        "id": "xxxx",
6        "name": "显示名称"
7      }
8    ]
9  }
```

2.74 获取facebook发布类型列表（未开放）

调用方式：http+json,Post传参

请求地址：<http://x.x.x.x:8080/osc/commands/execute>

命令ID：_live.facebook.pubTypes

请求头参数(version)：可不设置

输入参数：

字段	字段类型	描述
language	String	中文：zh 英文：en

输出参数：

字段	字段类型	描述
curlId	String	当前显示的ID
list	Array	从Facebook获取到的page数据
id	String	一个page的id
name	String	一个page的名字

返回格式：

```
1 {
2   "curType": Int    -->当前选择type
3   "list": [{
4     "type": Int     -->类型值, 0、1、2
5     "name": "String" -->名称
6   }]
7 }
```

2.75 取消任务接口(目前仅支持“隐藏拍摄者模式”开关开启后，调用camera.takePicture")

调用方式：http+json,Post传参

请求地址：http://x.x.x.x:8080/osc/commands/_cancel

请求头参数(version)：可不设置

命令ID：需要查询结果的id

字段	字段类型	描述
id	string	需要取消的任务id

输入参数：无

输出参数：无