

CamSDK 接口标准规范web版 (VersionV1.38)

文档修改记录

版本号	版本描述	修改人	日期	备注
V1.0	初稿	Emily	2021-02-06	

版本号	版本描述	修改人	日期	备注
V1.1	将高拍仪和扫描仪接口分开	Emily	2021-02-06	
V1.2	1.重新定义Json字符串格式和接口名称 2.高拍仪增加旋转接口	Emily	2021-04-25	
V1.3	1. Camera接口加入ID, 可同时控制多个摄像头 2. 发送和回传加入reqId, 以对应消息编号 3. 修改了命名方式 4. 返回结果用result 5. 打开摄像头后, 需要调用获取视频才会回传视频	Mike	2021-04-29	
V1.4	1.将Json里的result和devNum改成int型	Emily	2021-04-29	
V1.5	1. 增加错误信息errorMsg 2. 增加条码识别接口 3. 增加身份证读卡器接口 4. 增加指纹接口 5. 增加人脸接口 6. 增加上传和合并pdf接口	Emily	2021-05-10	
V1.6	整理了人像比对 此文档改为md文档, word文档不再维护	Jean	2021-08-23	
V1.6.1	FaceMatchr的回应消息中, 增加了matched、stoped、stopMode属性	Jean	2021-09-01	
V1.7	增加 概述	Jean	2021-09-14	
V1.8	增加CameraCaptureBook、CameraAutoCapture, CameraCapture命令增加mode属性; rotate支持自动文字方向	Jean	2021-09-30	
V1.9	增加其他功能: GetVersion、GetCommandList、GetOcrSupportInfo	Jean	2021-10-11	
V1.9.1	ReadIDCardInfo返回属性中增加外国人身份证与港澳台居住证的信息	Jean	2021-10-15	
V1.9.2	SetCameraImageInfo支持多区域手动裁切	Jean	2021-10-22	
V1.10	增加功能: FileOCR	Luke	2021-10-28	
V1.10.1	在 FileOCR 中增加mode属性	Jean	2021-11-01	
V1.10.2	在 CameraCapture 与 CameraCaptureBase64 中支持水印 watermark	Jean	2021-11-02	

版本号	版本描述	修改人	日期	备注
V1.11	增加外接USB按键: ExternalButton	Jean	2021-11-05	
V1.12	增加 LaunchExe	Jean	2021-11-29	
V1.13	MergeFile 支持图片合并	Jean	2021-12-17	
V1.14	在 ReadIDCardInfo 中增加返回指纹的base64数据和指纹位置信息	Luke	2022-12-29	
V1.15	在 ReadIDCardInfo 中支持身份证模拟照片的返回	Jean	2021-12-31	
V1.16	增加 AiRecognize ; 支持桌面裁切	Jean	2022-03-22	
V1.17	增加DPI、曝光、StillPin接口说明	Luke	2022-03-23	
V1.18	CameraCapture 支持自动裁切证件照片	Jean	2022-04-05	
V1.19	watermark 与 IdPhoto 属性, 从 CameraCapture 移到了 SetCameraImageInfo	Jean	2022-04-21	
V1.20	增加扩展屏的接口	Jean	2022-05-26	
V1.21	增加视频设置相机控制接口 CameraVideoParam	Jean	2022-07-15	
V1.22	CameraCapture 的imagePath参数支持指定其他可接受的扩展名	Jean	2022-07-29	
V1.23	增加 OcrConvert2Txt 接口	Jean	2022-08-02	
V1.24	RecogBarCode 支持Datamatrix的识别	Jean	2022-08-10	
V1.25	FileOCR 支持区域识别; 增加 GetErrorCodeList 接口	Jean	2022-08-27	
V1.26	增加 UploadFtpFile ; UploHttpFile 支持一次上传多个文件	Jean	2022-10-18	
V1.27	增加 ZoomPreview ; 社保卡返回的结果信息中, idCardInfo改名为cardInfo	Jean	2022-11-24	
V1.28	GetCameraInfo 增加licenseCheck	Jean	2023-02-06	

版本号	版本描述	修改人	日期	备注
V1.29	GetCameraInfo 增加deviceType OpenCamera 支持userName、password参数，用于网络摄像头设置访问认证信	Jean	2023-02-08	
V1.30	完善拓展屏接口描述。	Luke	2023-03-03	
V1.31	修改概述	Jean	2023-03-06	
V1.32	增加条码枪接口。	Luke	2023-03-16	
V1.33	增加录像接口RecordVideo 与 GetAudioDevInfo	Jean	2023-05-04	
V1.34	增加发起签字&获取指纹请求接口	Luke	2023-07-20	
V1.35	ReadFingerInfo增加timeout属性	Jean	2023-08-01	
V1.36	修改扫描接口	Jean	2023-09-28	
V1.37	GetVideoParameter接口增加branch与property属性	Jean	2023-11-02	
V1.37	补充 SetCameraInfo 接口	Jean	2024-01-08	

概述

CamSDKServer 是款基于WebSocket服务架构的SDK 开发包。提供各类代码调用说明及运用描述，其可被运用集成于高拍仪、身份证读卡器、扫描仪等硬件产品开发。

CamSDKServer 实现了一个WebSocket的服务端，把高拍仪等设备的功能进行了封装。应用程序也可使用CamSDKClient DLL与CamSDKServer进行连接（提供了C++及C# 的Demo程序）；在支持WebSocket的浏览器（Google Chrome, Microsoft Edge, FireFox以及IE 11）中，可直接与CamSDKServer进行连接（提供了Demo页面）；对于不支持WebSocket的浏览器（如IE8），提供了CamSDKClient OCX控件，来实现与CamSDKServer的连接（提供了Demo页面）。



CamSDKServer 与其客户端之间，通过JSON消息（也称为请求/响应，在有些场景下也会使用二进制消息格式）进行交互。客户端发送命令（也称为请求）给服务端，服务端执行命令后把结果（也称为响应）发回给客户端。交互的消息，着眼于业务功能层面，简单易用，可减少大量的设备操作细节，能更方便快捷的开发自己的应用。

CamSDKServer 也实现了与其他WebSocket服务端的代理连接。这个功能有得于功能扩展。

高拍仪接口

获取设备信息：GetCameraInfo

获取摄像设备信息，包括所有摄像头的名称、媒体名称、分辨率等。一般在打开摄像头设备之前，通过这个命令来获取所有的摄像头设备信息。这个命令的调用不民必需的，如果在应用中使用固定的高拍仪设备，可以不调用GetCameraInfo而直接打开摄像设备。

请求格式

```
{
  "func": "GetCameraInfo",
  "reqId": 123
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "GetCameraInfo",
  "reqId": 123,
  "result": 0,
  "licenseCheck": 1,
  "devInfo": [
    {
      "id": 0,
      "devName": "UDS CamScanner",
      "camMode": 0,
      "funcType": 1,
      "deviceType": 1,
      "mediaTypes": [
        {
          "mediaType": "MJPG",
          "resolutions": [
            "2592x1944",
            "2048x1536",
            "1920x1080",
            "1600x1200",
            "1280x1024",

```

```

        "1280x720",
        "1024x768",
        "800x600",
        "640x480"
    ]
},
{
    "mediaType": "YUY2",
    "resolutions": [
        "2592x1944",
        "2048x1536",
        "1920x1080",
        "1600x1200",
        "1280x1024",
        "1280x720",
        "1024x768",
        "800x600",
        "640x480"
    ]
}
],
"stillPin": {
    mediaType: "MJPG",
    resolutions: [
        "4896x3672",
        "4608x3456",
        "3648x2740",
        "3264x2448",
        "2592x1944",
        "2048x1536",
        "1920x1080",
        "1600x1200",
        "1280x720",
        "640x480"
    ]
}
},
{
    "id": 1,
    "devName": "S520-2",
    "camMode": 1,
    "funcType": 2,
    "deviceType": 1,
    "mediaTypes": [
        {
            "mediaType": "YUY2",
            "resolutions": [
                "1600x1200",
                "1280x1024",
                "1280x960",
                "1280x720",
                "800x600",
                "640x480",
                "352x288",
                "320x240",
                "160x120"
            ]
        }
    ]
}

```

```
]
},
{
  "id": 2,
  "devName": "Integrated Camera",
  "camMode": 99,
  "funcType": 2,
  "deviceType": 1,
  "mediaTypes": [
    {
      "mediaType": "MJPG",
      "resolutions": [
        "1280x720",
        "960x540",
        "848x480",
        "640x480",
        "640x360",
        "424x240",
        "352x288",
        "320x240",
        "320x180"
      ]
    },
    {
      "mediaType": "YUY2",
      "resolutions": [
        "1280x720",
        "960x540",
        "848x480",
        "640x480",
        "640x360",
        "424x240",
        "352x288",
        "320x240",
        "320x180"
      ]
    }
  ]
}
]
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
errorMsg	字符串	result不为0时，result所表示的错误信息。
licenseCheck	数字	1表示授权检测成功，0授权检测不成功。
devInfo	对象数组	表示多个摄像头设备的详细信息
devName	字符串	摄像头设备的名称
id	数字	摄像头的设备ID号，也称为设备号码(devNum)。打开摄像头时需要指定的设备号码就是这个值。
camMode	数字	摄像头的模式类型，有如下取值： 0:主头；1:副头；2:外接副头；11:双目彩色；12:双目红外；13: 3D人像摄像头；99:其他摄像头。
funcType	数字	摄像头的功能类型：0-未知功能类型；1-文档摄像头；2-人像摄像头
deviceType	数字	摄像头设备类型设备类型： 1:USB摄像头；2:RTSP网络摄像头；3:RTMP网络摄像头；4:HTTP网络摄像头
mediaTypes	对象数组	表示此摄像头支持的多种媒体类型
mediaName	字符串	媒体类型名称，一般为：MJPEG 或 YUY2
resolutions	字符串数组	表示在这种媒体类型下，此摄像头所支持的分辨率。单个分辨率以"宽x高"的形式表示。
stillPin	json对象	表示设备支持StillPin出图，包含StillPin下所支持的分辨率数组和媒体类型。

提示：不同功能场景下，可通过camMode或funcType来过滤可使用的设备。

打开摄像头设备：OpenCamera

请求格式


```
{
  "func": "OpenCamera",
  "reqId": 125,
  "devNum": 0,
  "mediaNum": 0,
  "resolutionNum": 0,
  "fps": 10,
  "userName": "admin",
  "password": "9999"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要打开的设备的号码，GetCameraInfo时返回的设备的id值。
subtype	数字	视频格式编号，即GetCameraInfo返回的mediaTypes数组中对应的序号（从0开始）。与mediaNum只需指定一个即可
width	数字	分辨率的宽，指定resolutionNum时不用指定width和height
height	数字	分辨率的高，指定resolutionNum时不用指定width和height
mediaNum	数字	视频格式编号,即GetCameraInfo返回的mediaTypes数组中对应的序号（从0开始）。与subtype只需指定一个即可
resolutionNum	数字	分辨率编号,即GetCameraInfo返回的resolutions数组中对应分辨率的序号（从0开始），指定width、height时不用指定resolutionNum
fps	数字	帧率
userName	字符串	打开网络摄像头时，需要提供用户名来进行验证。打开USB摄像头时,不需指定此参数。
password	字符串	打开网络摄像头时，需要密码来进行验证。打开USB摄像头时,不需指定此参数。

返回格式

```
{
  "func": "OpenCamera",
  "reqId": 125,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

获取视频流：GetCameraVideoBuff

在需要预览摄像头的视频时，可通这个命令来开启或停止视频的回传。开启时，只需要发送一条GetCameraVideoBuff开启的命令，视频会自动不停的回传，直到收到GetCameraVideoBuff停止命令或关闭摄像头，或者从服务器断开连接。也就是说，启动视频流命令，是一问多答；而停止视频流命令，是一问一答。

请求格式

```
{
  "func": "GetCameraVideoBuff",
  "reqId": 126,
  "devNum": 0,
  "width": 640,
  "height": 480,
  "fps": 10,
  "mode": "bin",
  "enable": "true"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要打开的设备的号码，GetCameraInfo时返回的设备的id值。
width	数字	回传视频的宽，可省略，默认为640。实际返回的视频宽度根据取图分辨率按比例计算，并不正好是这个值。指定比实际取图的宽度大的值，并不会有效果。
height	数字	回传视频的高，可省略，默认为480。实际返回的视频高度根据取图分辨率按比例计算，并不正好是这个值。指定比实际取图的高度大的值，并不会有效果。
fps	数字	回传视频的帧率，可省略，默认为10。以软件丢帧的方式实现，并不会高于实际取图的帧率。
mode	字符串	回传视频的方式，base64 或 bin，可省略，默认为base64。bin方式回传效率比base64方式高。
enable	布尔型或字符串型或数字	是否继续回传视频。 true 或 字符串true 或 数字1：打开视频回传， false 或 字符串false 或 数字0：关闭视频回传。

返回格式

以base64方式返回视频时，Json消息格式如下：

```
{
  "func": "GetCameraVideoBuff",
  "reqId": 126,
  "devNum": 0,
  "result": 0,
  "width": 640,
  "height": 480,
  "mime": "image/jpeg",
  "imgBase64Str": "...."
}
```

以bin方式返回视频时，字节消息格式如下：

```
[头部字节数(1字节)] -----\
[数据类型(1字节，视频帧数据类型：0x01)] |
[摄像头号码devNum(1字节)] |
[mime类型(1字节：R表示RGB；J表示JPG)] |
[视频宽(2字节)] > 消息头部
[视频高(2字节)] |
[视频帧字节数(4字节)] |
[func名称(以/0结尾)] |
```

```
[reqId(以0结尾)]-----/
[视频帧的数据.....]

[协议头,固定的0x22 1个字节] -----\
  [协议字节码长度,4个字节,高位在前]//{"func": "CloseCamera","reqId": 127} |
  [附带数据长度,4个字节,高位在前]                                     >消息头
[协议json数据,长度等于头文件固定的json协议字节码,ascii编码]         |
[附带数据,如果有就有就没有]-----/
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	摄像头的号码
width	数字	回传视频的宽，与请求中的width可能不相同。因为实际返回的视频宽度根据取图分辨率按比例计算，并不正好是请求的那个值。
height	数字	回传视频的高，与请求中的height可能不相同。因为实际返回的视频高度根据取图分辨率按比例计算，并不正好是请求的那个值。
mime	字符串	MIME类型，一般是image/jpg
imgBase64Str	字符串	视频帧jpg图像数据经base64转码后的串。

缩放预览视频：ZoomPreview

对预览视频，进行软件缩放。设置缩放的百分比，以及中心点百分数坐标。在重新打开摄像头时，会自动复位缩放百分比为100，也就是不缩放。

请求格式

```
{
  "func": "ZoomPreview",
  "reqId": 156,
  "devNum": 0,
  "percent": 300,
  "centrePercentX": 10,
  "centrePercentY": 10
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要调整缩放预览的设备的号码，GetCameraInfo时返回的设备的id值。
percent	数字	缩放的百分比，最小值100，最大值2000。即最小为不放大；最大为放大20倍。
centrePercentX	数字	预览图的中心点，相对于原图的中心点，在水平方向偏移的百分数(相对于原图宽度)。取值范围：0-50
centrePercentY	数字	预览图的中心点，相对于原图的中心点，在垂直方向偏移的百分数(相对于原图高度)。取值范围：0-50

返回格式

```
{
  "func": "ZoomPreview",
  "reqId": 156,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

关闭摄像头：CloseCamera

关闭指定的摄像头。如果指定摄像头已处于关闭状态，则不进行处理。

请求格式

```
{
  "func": "CloseCamera",
  "reqId": 127,
  "devNum": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要关闭的设备的号码，GetCameraInfo时返回的设备的id值。

返回格式

```
{
  "func": "CloseCamera",
  "reqId": 127,
  "devNum": 0,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

设置摄像头：SetCameraInfo

用于在摄像头打开的情况下，直接改变摄像头出图的媒体类型与分辨率。

请求格式

```
{
  "func": "SetCameraInfo",
  "reqId": 361,
  "devNum": 0,
  "mediaNum": 1,
  "resolutionNum": 2
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码， GetCameraInfo 时返回的设备的id值。
mediaNum	数字	视频格式编号, 即GetCameraInfo返回的mediaTypes数组中对应的序号（从0开始）
resolutionNum	数字	分辨率编号, 即GetCameraInfo返回的resolutions数组中对应分辨率的序号（从0开始）

返回格式

```
{
  "func": "SetCameraInfo",
  "reqId": 361,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

设置摄像头算法：SetCameraImageInfo

用于设置像头算法参数：裁切方式、旋转方式、颜色处理方式、填充方式、去除手指或去除装钉孔等。

请求格式

```
{
  "func": "SetCameraImageInfo",
  "reqId": 128,
  "devNum": 0,
  "cropType": 9,
  "posArray": [640, 480, 100, 20, 300, 220],
  "imageType": 0,
  "fillBorderType": 0,
  "removalForeign": 0,
  "rotate": 0,
  "dpi": 188,
  "jpgQuanlity": 75,
  "watermark": {
    "font": "宋体",
    "fontSize": 60,
    "color": "#F1F1F1",
    "transparent": 200,
    "text": "水印测试",
    "layout": 1
  },
  "IdPhoto": {
    "bgColor": "#0000FF"
  }
}
```



```
}  
}
```

请求参数说明

属性名称	类型	含义
------	----	----

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码， GetCameraInfo 时返回的设备的id值。
cropType	数字	裁切方式：0-不裁切;1-单图裁切; 2-多图裁切; 6桌面裁切; 9-手动裁切 单图裁切和多图裁切视频会有绿框显示，曲面矫正会显示中线。 是手动裁切方式时，支持多个裁切框，还需要指定 posArray 属性。
posArray	数字数组	手动裁切时，指定视频预览区（即各裁切框所在的平面）的宽高及多个裁切框的位置。 数组元素依次为：视频预览区的宽、视频预览区的高、框1的left、框1的top、框1的right、框1的bottom、框2的left、框2的top、框2的right、框2的bottom、.... 根据这个规则，数组元素的个数至少为6个、或者10个、14个，依此类推。
imageType	数字	色彩处理方式：0-彩色原色;1-灰度图片;2-黑白文档;3-彩色文档;4-红印文档;5-蓝印文档;6-彩色照片;7-票据增强
fillBorderType	数字	补边方式：0-不填充; 1-映射填充; 2-填充白色; 3-自动纯色填充 只有在使用自动裁切功能时，自动补边才有效。
removalForeign	数字	异物去除方式：0-无； 1-去除手指(仅用于拍书功能)； 2-去除装订孔
rotate	数字	旋转方式：0-不旋转； 90； 180； 270。顺时针旋转方向。 361-自动按文字方向旋转，需要有OCR库支持，并且需要指定 textOrientationLangId 属性。
textOrientationLangId	数字	指定OCR识别时使用的语言ID。使用 GetOcrSupportInfo 命令获取支持的OCR语言列表。
dpi	数字	设置拍照的DPI值
watermark	对象	在拍摄的照片上打上水印。

属性名称	类型	含义
text	字符串	水印输出的文字内容。
font	字符串	水印输出的字体，要求系统必需有这种字体。默认为 Arial。
fontSize	数字	水印输出的字体大小。默认为72。
color	字符串	水印输出的字体颜色，如：#FF0000 表示红色。默认为红色。
transparent	数字	水印输出的文本的透明度，0-255，255表示不透明，默认为255。
layout	数字	水印输出的文本的布局方式：0： LeftTop 1： Center 2： Diagonal 3： RightBottom
IdPhoto	对象	指定此属性时，拍摄时自动裁切证件照片，证件照片像素大小：358 * 441
bgColor	字符串	自动裁切证件照时的背景颜色，如：#0000FF 表示蓝色。默认为蓝色。
jpgQuanlity	数字	设置压缩率，范围 0~100:值越大压缩率最小 质量最好 尺寸最大

返回格式

```
{
  "func": "SetCameraImageInfo",
  "reqId": 128,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

拍摄照片：CameraCapture

拍摄图片到指定位置，根据后缀名确定文件格式，文件格式超出范围即报错。不指定文件路径时会自动生成文件名，默认为jpg格式。也可把拍摄图片以Base64编码方式返回。

在使用自动拍照（CameraAutoCapture）时，返回拍照（isBook为false）数据也是CameraCapture的返回格式。

请求格式

```
{
  "func": "CameraCapture",
  "reqId": 129,
  "devNum": 0,
  "imagePath": "pic1.jpg",
  "mode": "base64;path",
  "watermark": {
    "font": "宋体",
    "fontSize": 60,
    "color": "#F1F1F1",
    "transparent": 200,
    "text": "水印测试",
    "layout": 1
  },
  "IdPhoto": {
    "bgColor": "#0000FF"
  }
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码，GetCameraInfo时返回的设备的id值。
imagePath	字符串	拍摄照片到指定的路径。可以是全路径，也可以只有文件名，也可省略此参数。 只指定文件名时，会在%TEMP%/CamSDK目录下生成文件。省略此参数时会自动生成文件名 在多图裁切时，拍照会生成多个文件（在文件名后面加_0这样的后缀）。 文件名支持的扩展名有：jpg, bmp, png, gif, tif, pcx, pdf等，默认扩展名为.jpg。指定不支持的扩展名时会自动改为默认的.jpg扩展名。
mode	字符串	返回照片的方式: base64; path, 默认为path, 也可同时使用,中间用;或, 或&隔开，如: base64;path
watermark	对象	在拍摄的照片上打上水印。CameraCapture中指定的的watermark优先SetCameraImageInfo中指定的的watermark。
text	字符串	水印输出的文字内容。
font	字符串	水印输出的字体，要求系统必需有这种字体。默认为 Arial。
fontSize	数字	水印输出的字体大小。默认为72。
color	字符串	水印输出的字体颜色，如：#FF0000 表示红色。默认为红色。
transparent	数字	水印输出的文本的透明度，0-255，255表示不透明，默认为255。
layout	数字	水印输出的文本的布局方式：0： LeftTop 1： Center 2： Diagonal 3： RightBottom
IdPhoto	对象	指定此属性时，拍摄时自动裁切证件照片, 证件照片像素大小：358 * 441。 CameraCapture中指定的的IdPhoto优先SetCameraImageInfo中指定的的IdPhoto。
bgColor	字符串	自动裁切证件照时的背景颜色，如：#0000FF 表示蓝色。默认为蓝色。

返回格式

```
{
  "func": "CameraCapture",
  "reqId": 129,
  "result": 0,
```

```
"devNum": 0,
"imagePath": [
  "F:\\CamSDK\\release_ws\\tmp\\03629548\\pic1_0.jpg",
  "F:\\CamSDK\\release_ws\\tmp\\03629548\\pic1_1.jpg"
],
"imgBase64": [
  ".....",
  ".....",
],
"mime": "image/jpg"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	拍照的摄像头设备的号码
imagePath	字符串 数组	拍照生成的文件的全路径的数组，至少有一个，多图裁切时可能会有多个。 仅在请求参数mode中有path时才会出现此属性。
imgBase64	字符串 数组	拍照生成的文件的Base64编码串，至少有一个，多图裁切时可能会有多个。 仅在请求参数mode中有base64时才会出现此属性。
mime	字符串	MIME类型，一般是image/jpg。仅在请求参数mode中有base64时才会出现此属性。

拍摄成Base64字符串：CameraCaptureBase64

按jpg格式返回图片base64字符串，如果生成多张图片，会回传多次。

不推荐使用此命令，建议使用CameraCapture命令，指定mode为base64。

请求格式

```
{
  "func": "CameraCaptureBase64",
  "reqId": 130,
  "devNum": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码， GetCameraInfo时返回的设备的id值。
watermark	对象	在拍摄的照片上打上水印。
text	字符串	水印输出的文字内容。
font	字符串	水印输出的字体，要求系统必需有这种字体。默认为 Arial。
fontSize	数字	水印输出的字体大小。默认为72。
color	字符串	水印输出的字体颜色，如： #FF0000 表示红色。默认为红色。
transparent	数字	水印输出的文本的透明度， 0-255， 255表示不透明，默认为255。
layout	数字	水印输出的文本的布局方式： 0： LeftTop 1： Center 2： Diagonal 3： RightBottom

返回格式

```
{
  "func": "CameraCaptureBase64",
  "reqId": 130,
  "result": 0,
  "devNum": 0,
  "imageCount": 4,
  "imageIndex": 1,
  "mime": "image/jpg",
  "imageBase64Str": "....."
}
```

拍照生成有多张图时，会回发多次，每次回发一张图。

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	拍照的摄像头设备的号码，GetCameraInfo时返回的设备的id值。
imageCount	数字	总共有几张图
imageIndex	数字	第几张图
mime	字符串	MIME类型，一般是image/jpg
imageBase64Str	字符串	图片数据经base64转码后的串。

拍摄书本：CameraCaptureBook

拍书功能，可把展开的书本，一次拍摄，经曲面展平、去手指，再自动切分成左右两页。拍出的左右两页，可以path或（和）Base64方式在回应消息中返回。拍书时，把书本的中缝对齐绿色中分线（需要界面负责显示），以保证拍摄效果，如下图。此功能只有在支持拍书功能的高拍仪上可用。

在使用自动拍照（CameraAutoCapture）时，返回拍书（isBook为true）数据也是CameraCaptureBook的返回格式。



请求格式

```
{
  "func": "CameraCaptureBook",
  "reqId": 131,
  "devNum": 0,
  "mode": "base64,path",
  "imagePath": "book1.jpg"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码，GetCameraInfo时返回的设备的id值。
mode	字符串	返回照片的方式: base64; path, 默认为path, 也可同时使用,中间用,隔开，如: base64;path
imagePath	字符串	拍摄到指定的路径，可以是全路径。左右页的文件名会在扩展名前面加上L 或R这样的后缀。此参数可省略，省略时会自动生成文件名。

返回格式

```
{
  "func": "CameraCaptureBook",
  "reqId": 131,
  "result": 0,
  "devNum": 0,
  "pageL": "F:\\CamSDK\\release_ws\\tmp\\03629548\\book1_L.jpg",
  "pageR": "F:\\CamSDK\\release_ws\\tmp\\03629548\\book1_R.jpg",
  "pageLBase64Str": ".....",
  "pageRBase64Str": ".....",
  "mime": "image/jpg"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	拍照的摄像头设备的号码。
pageL	字符串	左页的文件全路径，请求时没有指定mode为path，则无此属性。识别不出左页也无此属性。
pageR	字符串	右页的文件全路径，请求时没有指定mode为path，则无此属性。识别不出右页也无此属性。
pageLBase64Str	字符串	左页图像的Base64串，请求时没有指定mode为base64，则无此属性。识别不出左页也无此属性。
pageRBase64Str	字符串	右页图像的Base64串，请求时没有指定mode为base64，则无此属性。识别不出右页也无此属性。
mime	字符串	MIME类型，一般是image/jpg

自动拍摄：CameraAutoCapture

自动拍摄功能有两种工作模式：一是自动检测图像变化，在图像稳定一段时间之后自动拍摄；二是按给定的定时时间，自动触发拍摄。自动拍摄时，可以拍普通文档、也可拍书本。

启动自动拍摄的命令，是一问多答；停止自动拍摄的命令，是一问一答。与同是一问多答的GetCameraVideoBuff命令不同的是，CameraAutoCapture在回应操作结果的消息中，func是CameraAutoCapture；而在返回拍摄数据时的消息中，func却是CameraCapture或CameraCaptureBook。也就是说，服务端自动触发了拍摄照片或拍摄书本的命令。

以拍书方式启动自动拍摄时，要求高拍仪要支持拍书功能。

断开连接或关闭启动自动拍摄的摄像头，会自动停止自动拍摄。

请求格式

启动自动拍摄：

```
{
  "func": "CameraAutoCapture",
  "reqId": "223391",
  "devNum": 0,
  "enable": true,
  "isBook": false,
  "detectionType": 0,
  "param": 6,
  "mode": "path",
  "imagePath": "....."
}
```

停止自动拍摄：

```
{
  "func": "CameraAutoCapture",
  "reqId": "223392",
  "devNum": 0,
  "enable": false
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	启动或停止自动拍摄的摄像头设备的号码，GetCameraInfo时返回的设备的id值。
enable	布尔型或字符串型	启动或停止自动拍摄，true-启动自动拍摄，false-停止自动拍摄。停止自动拍摄时不用指定后面的属性。
detectionType	数字	自动拍摄的工作模式：0-自动检测图像变化；1-定时拍摄
param	数字	detectionType为0时，表示检测到图像稳定几次后，触发拍摄 detectionType为1时，表示定时拍摄的秒数。
notify	数字	是否在触发时发送事件通知。发送事件通知的格式见返回格式中的说明。 0: 不发送通知，这是缺省值。 1: 在触发时只发送通知，不执行拍照动作。这样应用程序可在收到通知后进行其他的动作，比如识别条码等等。在notify为1时，因为不需要拍照，所以isBook、mode、imagePath这三个参数会被忽略。 2: 在触发时既发送通知，也执行拍照动作。
isBook	布尔型或字符串型	启动自动拍摄是否是书本模式， true-书本模式，在返回拍摄数据时的消息中，func是CameraCaptureBook false-普通文档模式，在返回拍摄数据时的消息中，func是CameraCapture
mode	字符串	返回照片的方式: base64; path, 默认为path, 也可同时使用,中间用,;隔开，如: base64;path

属性名称	类型	含义
imagePath	字符串	拍摄到指定的路径，可以是全路径。左右页的文件名会在扩展名前面加上L或R这样的后缀。此参数可省略，省略时会自动生成文件名。

返回格式

操作回应消息：

```
{
  "func": "CameraAutoCapture",
  "reqId": 223391,
  "devNum": 0,
  "enable": true,
  "isBook": false,
  "detectionType": 0,
  "result": 0
}
```

或

```
{
  "func": "CameraAutoCapture",
  "reqId": 223391,
  "devNum": 0,
  "enable": true,
  "isBook": true,
  "detectionType": 0,
  "result": 267,
  "errorMsg": "不支持拍书"
}
```

返回拍摄数据时的消息格式，参见 [CameraCapture](#) 或 [CameraCaptureBook](#)

在启用事件通知后，当触发时，服务端会发回一条以下格式的消息。

```
{
  "func": "Notify",
  "event": "OnCameraAutoCapture",
  "reqId": <CameraAutoCapture开启时的reqId>,
  "time": "<触发时的系统时间的时间戳>",
  "result": 0
}
```

其中OnCameraAutoCapture表示是CameraAutoCapture触发事件。

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	请求参数中的 devNum 属性值，复制到返回参数中，方便业务处理。
enable	布尔型	请求参数中的 enable 属性值，复制到返回参数中，方便业务处理。
isBook	布尔型	请求参数中的 isBook 属性值，复制到返回参数中，方便业务处理。
detectionType	数字	请求参数中的 detectionType 属性值，复制到返回参数中，方便业务处理。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
errorMsg	字符串	result不是0时，表示result对应的错误信息。

启动exe：LaunchExe

可使用LaunchExe来启动指定的Exe程序来完成相应的功能。这适用于业务不需要指定复杂的拍照参数，完全由操作人员灵活处理的场景。目前仅支持SmartCapture这一种exe程序。

请求格式

```
{
  "func": "LaunchExe",
  "reqId": 1112,
  "exeName": "SmartCapture",
  "caption": "测试智能扫描功能",
  "mode": "base64;path"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
exeName	字符串	要启动的Exe程序的名称，目前仅支持SmartCapture这一种exe程序。
caption	字符串	指定启动有Exe程序的窗口标题，一般设置为提示用户如何操作的简要信息。
mode	字符串	返回照片的方式: base64; path, 默认为base64, 也可同时使用,中间用;,隔开，如: base64;path

不同的Exe程序 会有各自不同的请求参数。

返回格式

这个LaunchExe命令会分两次返回结果，第一次是LaunchExe本身的返回结果，表示exe启动成功或失败，如下所示：

```
{
  "func": "LaunchExe",
  "reqId": 1112,
  "result": 0
}
```

第二次返回结果的func为ExecutionResult，reqId与LaunchExe请求的reqId相同。这表示Exe执行成功后返回的结果，如下所示：

```
{
  "func": "ExecutionResult",
  "reqId": 1112,
  "result": 0,
  "imageCount": 2,
  "imagePath": [
    "F:\\CamSDK\\release_ws\\tmp\\03629548\\295657781_0.jpg",
    "F:\\CamSDK\\release_ws\\tmp\\03629548\\295657781_1.jpg"
  ],
  "mime": "image/jpg",
  "imgBase64": [
    ".....",
    ".....",
  ]
}
```

返回参数说明

属性名称	类型	含义
func	字符串	返回结果的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
imageCount	数字	Exe拍照返回的图片的数量。当拍照r参加吗的图片很从时，ExecutionResult的结果消息会拆分成多条，这时imagePath或imgBase64的length会小于imageCount.
imagePath	字符串数组	拍照返回的图片的路径的数组，仅当LaunchExe请求时指定的mode包含path才会有这个属性。
imgBase64	字符串数组	拍照返回的图片的Base64串的数组，仅当LaunchExe请求时指定的mode包含base64才会有这个属性。
mime	字符串	MIME类型，一般是image/jpg。仅在请求参数mode中有base64时才会出现此属性。

设置StillPin拍照参数：SetStillPinInfo

在设备支持StillPin采集的情况下，对StillPin拍照的分辨率进行设置。

请求格式

```
{
  "func": "SetStillPinInfo",
  "reqId": 11123456,
  "devNum": 0,
  "resolutionNum": 0
}
```


请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	设备编号（与获取设备名称列表顺序一致）。
resolutionNum	数字	摄像头所支持的分辨率（与获取到的分辨率列表的顺序一致）。

返回格式

操作回应消息：

```
{
  "func": "SetStillPinInfo",
  "reqId": 11123456,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

获取设备序列号：GetDevSN

获取设备序列号。

请求格式

```
{
  "func": "GetDevSN",
  "reqId": 11123456,
  "devNum": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	设备编号（与获取设备名称列表顺序一致）。

返回格式

操作回应消息：

```
{
  "func": "GetDevSN",
  "reqId": 11123456,
  "devSN": "abcdef123456",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devSN	字符串	表示设备的SN码。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

获取设备视频/相机控制参数：GetVideoParameter

获取设备视频或相机控制参数值。

请求格式

```
{
  "func": "GetVideoParameter",
  "reqId": 11123456,
  "devNum": 0,
  "branch": 1,
  "property": 4
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	设备编号（与获取设备名称列表顺序一致）。
branch	数字	0：视频类参数，1：摄像头控制类参数
property	数字	表示要获取哪一个参数值： branch为0时表示：0-亮度，1-对比度，2-色相，3-饱和度，4-清晰度，5-伽玛，6-启用颜色，7-白平衡，8-背光补偿，9-增益。 branch为1时表示：0-全景，1-倾斜，2-滚动，3-缩放，4-曝光，5-低照度补偿，6-对焦。

返回格式

操作回应消息：

```
{
  "func": "GetVideoParameter",
  "reqId": 11123456,
  "branch": 0,
  "property": 4,
  "param": {
    "min": -9,
    "max": -1,
    "value": -4,
    "default": -6,
    "flag": 1
  },
}
```

```
"result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
branch	数字	复制请求中的branch参数
property	数字	复制请求中的property参数
param	对象	包含当前属性的详细值
min	数字	属性值范围的最小值
max	数字	属性值范围的最大值
value	数字	当前属性值
default	数字	初始默认值
flag	数字	标志位，1：仅自动调节 2：可手动设置 3：可自动也可手动设置
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

设置视频/相机控制参数：SetVideoParameter

设置设备的视频参数或相机控制参数。并不是每个设备都会支持所有属性的设置。

请求格式

```
{
  "func": "SetVideoParameter",
  "reqId": 11123456,
  "devNum": 0,
  "branch": 1,
  "property": 4,
  "value": -5,
  "flag": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	设备编号（与获取设备名称列表顺序一致）。
branch	数字	0：视频类参数，1：摄像头控制类参数
property	数字	表示要获取哪一个参数值： branch为0时表示：0-亮度，1-对比度，2-色相，3-饱和度，4-清晰度，5-伽玛，6-启用颜色，7-白平衡，8-背光补偿，9-增益。 branch为1时表示：0-全景，1-倾斜，2-滚动，3-缩放，4-曝光，5-低照度补偿，6-对焦。
value	数字	设置的属性的值
flag	数字	是否自动调节；1：自动，0：手动

返回格式

操作回应消息：

```
{
  "func": "chGetDevSN",
  "reqId": 11123456,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

视频设置相机控制：CameraVideoParam

可对亮度、对比度、饱和度、伽马等视频设置参数；相机的曝光、对焦、缩放等调整控制参数；电源频率进行查询与设置。请求信息中有items属性或powerFreq属性，表示设置；否则表示查询。GetVideoParameter与SetVideoParameter是对单个属性的查询或修改，而CameraVideoParam可一次获取全部的属性或者设置多个（也可全部）的属性。

请求格式

查询的请求格式如下，查询时仅需指定查询的摄像头的设备号devNum。

```
{
  "func": "CameraVideoParam",
  "reqId": 356,
  "devNum": 0
}
```

设置的请求格式如下，其中items属性与powerFreq属性至少要有有一个，也可两个都有。items属性中，可有多个设置项。

```
{
  "func": "CameraVideoParam",
  "reqId": 357,
  "devNum": 0,
  "powerFreq": 50,
  "items": [
    {
      "id": 0,
      "val": 28,
      "flag": 2
    }
  ]
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	要设置的摄像头设备的号码，GetCameraInfo时返回的设备的id值。
powerFreq	数字	要设置的电源频率，为50或60，表示相应的赫兹。
items	对象数组	要设置的项目的数组，每项包括id, val, flag属性。
id	数字	要设置的项目的ID号。详见下表。
val	数字	要设置的值
flag	数字	1表示自动；2表示手动

项目ID含义表

项目ID号	分类	中文含义	英文含义
0	视频设置	亮度	Brightness
1	视频设置	对比度	Contrast
2	视频设置	色调	Hue
3	视频设置	饱和度	Saturation
4	视频设置	锐度，清晰度	Sharpness
5	视频设置	伽玛	Gamma
6	视频设置	启用颜色	Color Enable
7	视频设置	白平衡	White Balance
8	视频设置	背光补偿	Backlight Compensation
9	视频设置	增益	Gain
100	相机控制	全景	Pan
101	相机控制	倾斜	Tilt
102	相机控制	滚动	Roll
103	相机控制	缩放	Zoom
104	相机控制	曝光	Exposure
105	相机控制	低亮度补偿	Iris
106	相机控制	对焦	Focus

返回格式

查询时支返回所有项目信电源频率。查询时，项目的name，会根据操作系统的语言，返回中文或英文。查询返回格式如下：

```
{
  "devNum": 0,
  "func": "CameraVideoParam",
  "items": [
    {
      "capsFlag": 2,
      "defVal": 0,
      "flag": 2,
      "id": 0,
      "max": 64,
      "min": -64,
      "name": "亮度",
      "step": 1,
      "val": 28
    },
    {
```

```
"capsFlag": 2,
"defVal": 32,
"flag": 2,
"id": 1,
"max": 64,
"min": 0,
"name": "对比度",
"step": 1,
"val": 32
},
{
  "capsFlag": 2,
  "defVal": 0,
  "flag": 2,
  "id": 2,
  "max": 40,
  "min": -40,
  "name": "色调",
  "step": 1,
  "val": 0
},
{
  "capsFlag": 2,
  "defVal": 64,
  "flag": 2,
  "id": 3,
  "max": 128,
  "min": 0,
  "name": "饱和度",
  "step": 1,
  "val": 64
},
{
  "capsFlag": 2,
  "defVal": 5,
  "flag": 2,
  "id": 4,
  "max": 6,
  "min": 0,
  "name": "锐度",
  "step": 1,
  "val": 5
},
{
  "capsFlag": 2,
  "defVal": 100,
  "flag": 2,
  "id": 5,
  "max": 500,
  "min": 72,
  "name": "伽玛",
  "step": 1,
  "val": 100
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
```



```
"id": 6,
"max": 0,
"min": 0,
"name": "启用颜色",
"step": 0,
"val": 0
},
{
  "capsFlag": 3,
  "defVal": 4600,
  "flag": 1,
  "id": 7,
  "max": 6500,
  "min": 2800,
  "name": "白平衡",
  "step": 1,
  "val": 4600
},
{
  "capsFlag": 2,
  "defVal": 1,
  "flag": 2,
  "id": 8,
  "max": 2,
  "min": 0,
  "name": "背光补偿",
  "step": 1,
  "val": 1
},
{
  "capsFlag": 2,
  "defVal": 0,
  "flag": 2,
  "id": 9,
  "max": 100,
  "min": 0,
  "name": "增益",
  "step": 1,
  "val": 0
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 100,
  "max": 0,
  "min": 0,
  "name": "全景",
  "step": 0,
  "val": 0
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 101,
  "max": 0,
  "min": 0,
```

```
"name": "倾斜",
"step": 0,
"val": 0
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 102,
  "max": 0,
  "min": 0,
  "name": "滚动",
  "step": 0,
  "val": 0
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 103,
  "max": 0,
  "min": 0,
  "name": "缩放",
  "step": 0,
  "val": 0
},
{
  "capsFlag": 3,
  "defVal": -6,
  "flag": 1,
  "id": 104,
  "max": -1,
  "min": -13,
  "name": "曝光",
  "step": 1,
  "val": -6
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 105,
  "max": 0,
  "min": 0,
  "name": "低亮度补偿",
  "step": 0,
  "val": 0
},
{
  "capsFlag": 0,
  "defVal": 0,
  "flag": 0,
  "id": 106,
  "max": 0,
  "min": 0,
  "name": "对焦",
  "step": 0,
  "val": 0
}
```

```
    }  
  ],  
  "powerFreq": 50,  
  "reqId": 356,  
  "result": 0  
}
```

设置时的返回如下：

```
{  
  "func": "CameraVideoParam",  
  "reqId": 357,  
  "devNum": 0,  
  "result": 0  
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	查询或设置的摄像头设备的号码
powerFreq	数字	电源频率，为50或60，表示相应的赫兹。
items	对象数 组	查询到的项目的数组，每项包括id, name, capsFlag, defVal, max, min, step, val, flag属性。
id	数字	要设置的项目的ID号。详见上表。
name	字符串	项目的名称，根据操作系统不同，会返回中文或英文
capsFlag	数字	项目的标志：0不支持，1仅自动，2仅手动，3可设置自动或手动
defVal	数字	此项的默认值
max	数字	此项的最大值
min	数字	此项的最小值
step	数字	此项的值调整时的步长
val	数字	此项的当前值
flag	数字	此项的自动/手动设置：1表示自动；2表示手动

获取麦克风设备列表：GetAudioDevInfo

获取麦克风设备列表。在开启视频录制功能时，需要指定从哪个麦克风设备录音。使用这个接口，可以获取麦克风设备列表信息。

请求格式

```
{
  "func": "GetAudioDevInfo",
  "reqId": 358
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "closeCamera",
  "microphones": [
    "外部麦克风 (Realtek(R) Audio)",
    "麦克风 (USB Camera)",
    "麦克风阵列 (Realtek(R) Audio)"
  ],
  "reqId": 358,
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
microphones	字符串 数组	包含所有的麦克风设备的名称。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

录像：RecordVideo

开启或停止视频录制。

请求格式

开启录像：

```
{
  "func": "RecordVideo",
  "mainCamIndex": 0,
  "mainwidth": 1920,
  "mainHeight": 1080,
  "viceCamIndex": 1,
  "vicewidth": 480,
  "vicePosX": 10,
  "vicePosY": 10,
  "audioDevIndex": 0,
  "switchMainVideo": false,
  "reqId": 359
}
```

停止录像：

```
{
  "func": "RecordVideo",
  "stop": true,
  "reqId": 360
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号, 服务端在返回结果时会附带这个reqId属性, 这样客户端可通过这个属性来区分不同的请求。
mainCamIndex	数字	要录制的主摄像头索引号,-1表示未指定, 255表示录制桌面; 254表示录制指定窗口。
mainWidth	数字	录制的视频的宽(旋转后的。如果录制时需要旋转, 则先打开摄像头, 用SetCameraImageInfo设置旋转角度之后再录制)。
mainHeight	数字	录制的视频的高(旋转后的。如果录制时需要旋转, 则先打开摄像头, 用SetCameraImageInfo设置旋转角度之后再录制)。
viceCamIndex	数字	要录制的副摄像头索引号,-1表示未指定(不录副头视频)
viceWidth	数字	副视频贴在主视频之前缩小后的宽度,高度等比缩小
vicePosX	数字	副视频贴在主视频中的位置的水平位置
vicePosY	数字	副视频贴在主视频中的位置的垂直位置
areaX	数字	录制区域的左上角的水平座标,仅在mainCamIndex=255或254时才有效,4个区域参数全为0表示录制整个桌面或窗口
areaY	数字	录制区域的左上角的垂直座标,仅在mainCamIndex=255或254时才有效,4个区域参数全为0表示录制整个桌面或窗口
areaWidth	数字	录制区域的宽度,仅在mainCamIndex=255或254时才有效,要求是4的倍数,4个区域参数全为0表示录制整个桌面或窗口
areaHeight	数字	录制区域的高度,仅在mainCamIndex=255或254时才有效,要求是2的倍数,4个区域参数全为0表示录制整个桌面或窗口
windowHWND	数字	录制窗口时,指定窗口的句柄
frameSkipOrFPS	数字	录制摄像头时:录制一帧后需要跳过的帧数, 或者录制桌面/窗口时的帧率
bitrateKbps	数字	生成的视频的码率(Kbps)
audioDevIndex	数字	要录制的声音设备索引号,-1表示未指定(不录声音)
audioFilePath	字符串	要使用的声音文件全路径,此声音混入录制的视频文件中,长度不足时循环使用【暂不支持】
switchMainVideo	布尔型	是否切换主头的视频回调(仅在合并主副头时有效,把原先主头的视频回调,切换为合并后的视频数据)
savePath	字符串	要录制生成的视频文件路径, 可以是全路径, 也可以只有文件名。也可省略。省略时自动在temp目录下生成文件。
fileFormat	字符串	指定录制文件的格式, 默认为mp4, 目前仅支持mp4的文件扩展名。视频编码格式为H264。

属性名称	类型	含义
stop	布尔型	在要停止录像时，指定些属性为true。开始录像时可省略此属性或指定为false。

返回格式

开启录像：

```
{
  "func": "RecordVideo",
  "reqId": 359,
  "result": 0,
  "savePath": "C:\\Windows\\TEMP\\CamSDK\\01e2ead8\\1683194342_4823.mp4",
  "stop": false
}
```

停止录像：

```
{
  "func": "RecordVideo",
  "reqId": 360,
  "result": 0,
  "savePath": "C:\\Windows\\TEMP\\CamSDK\\01e2ead8\\1683194342_4823.mp4",
  "stop": true
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
savePath	字符串	生成的录制文件的全路径。
stop	布尔型	复制的请求指令中的stop属性值，便于页面处理。

扫描仪(TWAIN)接口

这节所描述的接口，都是通过TWAIN协议来实现的扫描。

获取扫描仪名称列表：GetScannerList

获取扫描仪设备名称的列表。

请求格式

```
{
  "func": "GetScannerList",
  "reqId": 1201
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "GetScannerList",
  "reqId": 1201,
  "result": 0,
  "scannerName": [
    "Canon DRC230",
    "Uniscan F40D"
  ]
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
GetScannerList	字符串数组	扫描仪名称列表。列表中的序号（从0开始）表示该设备的devNum号。

获取扫描仪参数信息：GetScannerParametes

获取指定索引号的扫描仪能支持的扫描参数，包括扫描尺寸、DPI、进纸方式、颜色类型等。

请求格式

```
{
  "func": "GetScannerParametes",
  "reqId": 1695885422038,
  "devNum": 1
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devNum	数字	扫描设备编号（与获取设备名称列表顺序一致，从0开始）

返回格式

```
{
  "func": "GetScannerParametes",
  "reqId": 1695885422038,
  "devNum": 1,
  "result": 0,
  "colorType": {
    "0": "黑白",
    "1": "灰度",
    "2": "彩色"
  },
  "currentColorType": 2,
```

```
"pageSize": {
  "0": "None",
  "1": "A4",
  "13": "A6",
  "2": "B5",
  "22": "A7",
  "5": "A5"
},
"currentPageSize": 1,
"pageSource": [
  "平板",
  "ADF正面",
  "ADF双面"
],
"currentPageSource": 0,
"scanDPI": [
  "150",
  "200",
  "300",
  "600"
],
"currentScanDPI": 150
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
devNum	数字	从请求命令中复制的扫描设备编号。
colorType	map对象	扫描仪支持的颜色类型，key为代号，value为颜色类型的名称
currentColorType	数字	扫描仪当前的颜色类型key值。
pageSize	map对象	扫描仪支持扫描纸张类型，key为代号，value为纸张类型的名称
currentPageSize	数字	扫描仪当前的纸张类型key值。
pageSource	字符串数组	扫描仪支持的进纸来源。平板扫描仪只会有一项“平板”。
currentPageSource	数字	扫描仪当前的进纸来源的序号（从0开始）
scanDPI	字符串数组	扫描仪支持的扫描分辨率的列表。
currentScanDPI	数字	扫描仪当前的扫描分辨率，300表示300DPI

获取当前扫描状态：GetScanStatus

获取当前扫描状态。扫描状态是针对所有扫描仪来说的，只要有一台扫描仪在扫描，返回的扫描状态就是“正在扫描”。

请求格式

```
{
  "func": "GetScanStatus",
  "reqId": 1206
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "GetScanStatus",
  "reqId": 1206,
  "result": 0,
  "scanning": 1
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
scanning	数字	当前扫描状态: 1-正在扫描中；0-没有扫描 扫描状态是针对所有扫描仪来说的，只要有一台扫描仪在扫描，返回的scanning就是1。

扫描：Scan

Scan命令可控制扫描开始或停止扫描（stop属性为1）。正常情况下，开始的扫描任务会自动结束，无需手动停止。在有些异常情况下，程序无法知晓扫描仪的扫描是否已经结束。这时需要调用带stop属性为1的Scan命令来终止之前的扫描任务。

请求格式

开始扫描：

```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "mode": "path",
  "sannnerParam": {
```

```
    "devNum": 1,  
    "showUI": 0,  
    "pageSource": 2,  
    "pageSize": 1,  
    "colorType": 2,  
    "scanDPI": 300,  
    "savePath": "DDD"  
  },  
  "processParam": {  
    "cropType": 2,  
    "imageType": 0,  
    "fillBorderType": 2,  
    "removalForeign": 2,  
    "removeBlank": 1,  
    "jpgQuanlity": 85,  
    "watermark": {  
      "font": "Arial",  
      "fontSize": 32,  
      "color": "#7a7a7a",  
      "transparent": 200,  
      "text": "机密文件",  
      "layout": 2  
    }  
  }  
}
```

停止扫描:

```
{  
  "func": "Scan",  
  "reqId": 1695888926558,  
  "stop": 1  
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
stop	数字或布尔型	1或true，表示停止扫描。开始扫描时可缺省此属性。
mode	字符串	返回图片的方式: base64; path, 默认为path, 也可同时使用,中间用;或,或&隔开，如: base64;path
sannnerParam	对象	控制扫描的参数集合
devNum	数字	扫描设备编号（与获取设备名称列表顺序一致，从0开始）

属性名称	类型	含义
showUI	数字或布尔型	1或true，表示扫描时使用扫描的UI界面，在其UI界面中，可选择相关的扫描参数。showUI时，可以省略pageSource、pageSize、colorType、scanDPI四个参数。 0或false，表示扫描时不使用扫描的UI界面。
pageSource	数字	扫描时使用的进纸来源的序号（从0开始）。
pageSize	数字	扫描时使用的纸张类型key值。
colorType	数字	扫描时使用的颜色类型key值。
scanDPI	数字	扫描时使用的扫描分辨率，300表示300DPI。
savePath	字符串	扫描文件存储（在默认存储目录下）的子目录，可省略。默认存储目录是： C:\Users<用户名>\AppData\Local\Temp\CamSDK<客户端代号> 可以通过CamSDKServer2.ini文件中的TmpPath配置项来修改默认存储目录。如果不想使用<客户端代号>的子目录，可以把UseClientDir配置为0。
processParam	对象	扫描后的图像处理参数集合
cropType	数字	自动裁切：0-不裁切；1-单图裁切；2-多图裁切
imageType	数字	色彩处理方式：0-彩色原色;1-灰度图片;2-黑白文档;3-彩色文档;4-红印文档;5-蓝印文档;6-彩色照片;7-票据增强
fillBorderType	数字	补边方式：0-不填充; 1-映射填充; 2-填充白色; 3-自动纯色填充。
removeBlank	数字	是否去除空白页：0-不去除;1-去除空白页
jpgQuanlity	数字	jpg文件保存时的图像质量（1-100），可缺省，默认值75
watermark	对象	扫描后的图像处理时，增加水印，不加水印时，可缺少这个属性。
text	字符串	水印输出的文字内容。

属性名称	类型	含义
font	字符串	水印输出的字体，要求系统必需有这种字体。默认为 Arial。
fontSize	数字	水印输出的字体大小。默认为72。
color	字符串	水印输出的字体颜色，如：#FF0000 表示红色。默认为红色。
transparent	数字	水印输出的文本的透明度，0-255，255表示不透明，默认为255。
layout	数字	水印输出的文本的布局方式：0： LeftTop 1： Center 2： Diagonal 3： RightBottom

返回格式

Scan是“一问多答”的命令。在发现的Scan命令正确，被接收处理之后，会返回如下内容：

```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "result": 0
}
```

如果Scan命令参数不对：

```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "result": 6,
  "errorMsg": "参数错误:缺少sannnerParam参数."
}
```

如果当时正在扫描：

```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "result": 9,
  "errorMsg": "前一次的扫描任务还没有结束"
}
```

在每扫描到一个文件，会收到如下消息，其中OnScannedFile表示扫描到了一个文件。


```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "result": 0,
  "event": "OnScannedFile",
  "imagePath": "D:\\Test\\新建 文件夹\\02a4b730\\DDD\\pic_0002.jpg",
  "imgBase64": ".....",
  "mime": "image/jpeg",
  "scanState": 0
}
```

在扫描结束时，会收到如下消息，其中OnScanEnd表示扫描结束。

```
{
  "func": "Scan",
  "reqId": 1695888919552,
  "result": 0,
  "event": "OnScanEnd",
  "scanState": 1
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
errorMsg	字符串	result不为0时，result所表示的错误信息。
event	字符串	扫描事件：OnScannedFile表示扫描到了一个文件。OnScanEnd表示扫描结束。
scanState	数字	当前扫描任务的状态：0-进行中；1-已完成；2-异常结束；3-手动结束。
imagePath	字符串	扫描并处理完成后的图像文件的全路径。仅在请求参数mode中指定了path才有此属性返回。
imgBase64	字符串	扫描并处理完成后的图像文件数据经base64转码后的串。仅在请求参数mode中指base64才有此属性返回。
mime	字符串	MIME类型，一般是image/jpg。仅在请求参数mode中指base64才有此属性返回。

扩展屏接口

扩展屏一般都附带有压感手写笔，能在扩展屏上进行签字与界面操作。扩展屏上的界面显示、签字与界面操作，由本节描述的相关接口进行交互。扩展屏上集成的射频读卡器、指纹读取器、摄像头等其他设备的交互，不由本节描述的接口负责。

扩展屏的功能由PadViewer4程序实现，与CamSdkServer的框图如下。



从上图可以看出，负责控制扩展屏的PadViewer4，实际上也是CamSdkServer的一个客户端。在“**页务整合页面**”（即扩展屏功能的调用端）中，只要在请求命令中通过设置**to属性为2**（详见下表说明），CamSdkServer就会把请求命令转发给PadViewer4。由PadViewer4处理完成后，再发给CamSdkServer，CamSdkServer再转发给功能调用者“页务整合页面”。PadViewer4在处理收到的命令时，除了**打开展示页面**、**扩展屏截屏**等几条固定的命令之外，其他命令都会转发给在Web browser控件中显示的页面，由这个在扩展屏上显示的页面来处理。

在扩展屏显示的页面中，有需要发给PadViewer4或“**页务整合页面**”的消息，PadViewer4在页面植入了一个CB_Common的函数，页面中可通过以下js代码来调用。

- 1、显示签名框。触发PadViewer4在指定的位置，按指定的颜色等，显示出签名框（浮在页面上）

```
// divSign是扩展屏显示的页面中的一个div，签名框控件就按这个div的位置与大显示
var r = document.getElementById("divSign").getBoundingClientRect();
// 签名区的位置区域与颜色，各元素依次为：left,top,right,bottom,前景色(默认"#000000"),背景色(默认"#FFFFFF"),
// 图像文件格式 bmp/jpg/png)，返回值方式(file/base64),笔迹线宽
window.CB_Common("displaySign", r.left, r.top, r.right, r.bottom, "#F0F0F0", "#051A05", "jpg", "base64", 10.0);
```

- 2、提交签名。触发PadViewer4隐藏签名框，并抓取签名图片，拼成消息发给“**页务整合页面**”

```
// 在扩展屏显示的页面中的“提交签名”按钮点击后的执行方法
function submitMySign() {
    var signType = '这是签名的类型,会与签名一起回传给【页务整合页面】,可任意写,用于区分一个业务中的多个不同签名';
    // 第三个参数是个回调函数,用于回显签名图片在本页面中,不需要时此参数可缺省。
    window.CB_Common('submitSign', signType, receiveSign);
}

function receiveSign(fileType, msg) {
    if (fileType == "signature_base64") {
        if (msg.substr(0,1) == '{') {
            var jsonObj = JSON.parse(msg);
            signature1.src = "data:image/jpg;base64," + jsonObj.file;
            signature1.style.display = '';
        }
    }
}
```

- 3、接收“**页务整合页面**”发来的消息

```
// 【页务整合页面】发来的消息，经PadViewer4后，都会通过onPvControlMsg来传给扩展屏显示的页面
function onPvControlMsg(msg) {
    var jsonObj = JSON.parse(msg);
    // 解析msg并处理.....
}
```

4、回传其他消息。用于向“**页务整合页面**”回传其他消息。

```
// 这个页面演示了银行取钱的场景，由客户在扩展屏上自己输入要取钱的金额，提交后这个金额传回到银行柜员的【页务整合页面】中。
function submitMoney() {
    var valMoney = document.getElementById('money').value;
    if (valMoney > 0) {
        // 这个消息格式可以任意，只要是json格式就行
        var msg = JSON.stringify({'myFunction': 'submitMoney', 'value': valMoney});
        window.CB_Common("customMsg", msg);
    }
}
```

关于to属性的说明

to属性的值	含义
1	消息转发给 连接的安卓一体机 处理
2	消息转发给 扩展屏程序PadViewer4 处理

打开展示页面：navigate

在扩展屏中打开展示页面。若CamSdkServer发现扩展屏程序PadViewer4还没有启动，会先启动PadViewer4。

请求格式

```
{
  "reqId": 1501,
  "to": 2,
  "func": "navigate",
  "url": "/html/welcome.html"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
to	数字	固定为2，表示会转发给扩展屏程序PadViewer4去处理
url	字符串	要显示的页面的URL。可以是PadViewer4程序所在目录的相对路径；也可是标准的URL，如： http://www.mydomain.com/welcome.html ；url的值为空时，关闭当前展示的页面，显示广告页。

返回格式

```
{
  "reqId": 1501,
  "from": 2,
  "func": "navigate",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
from	数字	固定为2，表示是由扩展屏程序PadViewer4发来的处理结果

打开签字窗口：openSign

在扩展屏中显示签字窗口，

请求格式

```
{
  "reqId": 1502,
  "to": 2,
  "func": "openSign",
  "sign": "open",
}
```

```
    "remark": "mark1",
    "color": {
      "background": "#051A05",
      "pen": "#F0F0F0"
    },
    "pos": {
      "top": 200,
      "left": 50,
      "width": 800,
      "height": 600
    }
  }
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
to	数字	固定为2，表示会转发给扩展屏程序PadViewer4去处理
sign	字符串	"open"或"close"; sign=open时，显示签字框； sign=close时，关闭签字框，下方的参数无效可不传。
remark	字符串	签字数据的标签，用来鉴别是哪次业务发起的请求；
color	json对象	前景色和背景色；
background	字符串	签字框的背景色。以“#”开头的6位十六进制数值表示一种颜色。
pen	字符串	签字框的前景色（笔的颜色）。以“#”开头的6位十六进制数值表示一种颜色。
pos	json对象	签字框的坐标和大小（相对签字屏）。
top	数字	签字框的top属性；
left	数字	签字框的left属性；
width	数字	签字框的宽；
height	数字	签字框的高；

返回格式

```
{
  "reqId": 1502,
  "from": 2,
  "func": "openSign",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
from	数字	固定为2，表示是由扩展屏程序PadViewer4发来的处理结果

扩展屏截屏：snapshot

对扩展屏拍快照（截屏）。一般用于在“**页务整合页面**”要求回显扩展屏上显示的内容的情况下。
这个命令可以是一问一答或者一问多答（fps大于0时，会连续返回截屏内容，直到停止截屏）。

请求格式

```
{
  "reqId": 1502,
  "to": 2,
  "func": "snapshot",
  "fps": 5,
  "width": 640
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
to	数字	固定为2，表示会转发给扩展屏程序PadViewer4去处理
fps	数字	表示每秒快照次数。为0表示停止；为-1表示只拍1次； 大于0时，表示每秒快照次数，考虑效率，不要超过10.
width	数字	截屏图像的宽度，默认为扩展屏的显示分辨率的宽度。指定一个小于分辨率宽度的值，则会在截图之后进行等比例 缩小截图。fps为0时不用指定此属性。

返回格式

```
{
  "reqId": 1502,
  "from": 2,
  "func": "snapshot",
  "imgBase64": ".....",
  "mime": "image/png",
  "result": 0
}
```

在停止截屏时，返回的消息中没有imgBase64属性。

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
imgBase64	字符串	截屏图像的Base64串。
mime	字符串	MIME类型，一般是image/png。
from	数字	固定为2，表示是由扩展屏程序PadViewer发来的处理结果。

签名返回：pvSignature

这个消息在扩展屏页面上提交签名时，会发给“**页务整合页面**”。“**页务整合页面**”是通过在适当的时机，发**navigation**命令来让扩展屏显示具有签名功能的页面，从而实现签名功能整合的。

pvSignature中的pv是PadViewer的首字母，用于区分其他设备的签名功能。

请求格式

```
无
```

请求参数说明

无

返回格式

```
{
  "reqId": 1503,
  "from": 2,
  "func": "pvSignature",
  "signType": ".....",
  "imgBase64": ".....",
  "mime": "image/jpg",
  "result": 0
}
```

在停止截屏时，返回的消息中没有imgBase64属性。

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
from	数字	固定为2，表示是由扩展屏程序PadViewer发来的处理结果。
signType	字符串	签名的类型，扩展屏页面中提交签名时，一并提交的signTypeec值，用于区分一个业务中的多个不同签名
imgBase64	字符串	签名图片的Base64串。
mime	字符串	MIME类型，一般是image/jpg。

自定义消息返回：pvCustomMsg

这个消息在扩展屏页面上提交签名时，会发给“**业务整合页面**”。假如扩展屏页面上通过下面的代码发送自定义消息。

```
var msg = JSON.stringify({'myFunction': 'submitMoney', 'value': 500});
window.CB_Common("customMsg", msg);
```

pvCustomMsg中的pv是PadViewer的首字母，用于区分其他设备的自定义消息功能。

请求格式

无

请求参数说明

无

返回格式

```
{
  "reqId": 1504,
  "from": 2,
  "func": "pvCustomMsg",
  "msg": { 'myFunction': 'submitMoney', 'value': 500 }
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
msg	对象	页面发的自定义消息对象，对应 window.CB_Common("customMsg", msg) 中的msg。
from	数字	固定为2，表示是由扩展屏程序PadViewer发来的处理结果。

发起签字&获取指纹请求：startSignAndGetFingerprint

这个消息会在拓展屏页面打开一个签字页面，客户在提交完签字后，会立马提示用户输入指纹。客户提交指纹数据后，会将指纹图片数据返回到消息发送方。签字图片数据和指纹图片数据会分两次结果返回。

pvSignature中的pv是PadViewer的首字母，用于区分其他设备的签名功能。

请求格式

```
{
  "reqId": 1501,
  "to": 2,
  "func": "startSignAndGetFingerprint"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
to	数字	固定为2，表示会转发给扩展屏程序PadViewer4去处理

签字图片数据返回格式

```
{
  "reqId": 1503,
  "from": 2,
  "func": "pvSignature",
  "signType": ".....",
  "imgBase64": ".....",
  "mime": "image/jpg",
  "result": 0
}
```

签字图片数据返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
from	数字	固定为2，表示是由扩展屏程序PadViewer发来的处理结果。
signType	字符串	签名的类型，扩展屏页面中提交签名时，一并提交的signType值，用于区分一个业务中的多个不同签名
imgBase64	字符串	签名图片的Base64串。
mime	字符串	MIME类型，一般是image/jpg。

指纹图片数据返回格式

```
{
  "func": "ReadFingerInfo",
  "result": 0,
  "fingerFeatureBase64": ".....",
  "fingerImageBase64": ".....",
  "mime": "image/bmp"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
fingerFeatureBase64	字符串	指纹特征的Base64串。
fingerImageBase64	字符串	指纹图像的Base64串。
mime	字符串	MIME类型，一般是image/bmp

条码枪接口

条码枪的启停：BarcodeScanner

开启或停止条码枪的扫描功能。在开启扫描之后，扫描到的条码会通过**Notify**事件的通知消息返回。详见后面的**条码枪数据返回格式**

请求格式

```
{
  "func": "BarcodeScanner",
  "enable": true,
  "reqId": 1201
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
enable	布尔型或字符串型或数字	是否开启条码枪扫描。 true 或 字符串true 或 数字1：开启条码枪扫描， false 或 字符串false 或 数字0：关闭开启条码枪扫描。
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "StartHidPosRead",
  "reqId": 1201,
  "devSN": "201B220018000405",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devSN	字符串	在开启条码枪扫描时，会有条码枪的设备SN返回。 在关闭条码枪扫描时没有此属性。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

条码枪数据返回格式

```
{
  "func": "Notify",
  "event": "OnReceivePosData",
  "data": "http://weixin.qq.com/r/3UwUDLDEK01XrfRv9xnP\r\n",
  "time": "1678952917",
  "result": 0
}
```

条码枪数据返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称；"Notify"表示服务端发送来的通知。
event	字符串	消息通知的类型；"OnReceivePosData"表示数据是条码枪返回的结果。
data	字符串	条码枪识别到的数据。
time	字符串	此条消息返回的时间戳。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

获取条码枪的设备SN：ReadBarcodeScannerDevSN

读取条码枪设备的设备序列号。只有在开启了条码枪扫描之后，才能使用此功能。

请求格式

```
{
  "func": "ReadBarcodeScannerDevSN",
  "reqId": 1201
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "ReadBarcodeScannerDevSN",
  "reqId": 1201,
  "devSN": "201B220018000405",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
devSN	字符串	条码枪设备的设备序列号。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。

识别

识别条码或二维码：RecogBarCode

识别指定图片上或高拍仪立即拍照中的条码或二维码。支持的条码类型见下表。

条码/二维码类型	是否支持
Code128	√
Code39	√
EAN13	√
EAN8	√
UPC-A	√
UPC-E	√
Codabar	√
Interleaved 2 of 5	√
Code 93	√
UCC_128	√
QR code	√
Data Matrix	√
Aztec	√
PDF 417	√
MaxiCode	√
RSS-14	√
RSS-Expanded	√

请求格式

```
{
  "func": "RecogBarCode",
  "reqId": 1208,
  "imagePath": "",
  "imageBase64": "",
  "devNum": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
imagePath	字符串	要识别的图片的全路径，包含后缀名。
imageBase64	字符串	要识别的图片数据经base64转码后的串。
devNum	数字	从指定的摄像头拍照后立即识别条码，GetCameraInfo时返回的设备的id值。

imagePath，imageBase64，devNum这三个参数至少要指定一个，指定多个时，优先级从前往后依次降低，有高优先级的，则低优先级的忽略。

返回格式

```
{
  "func": "RecogBarCode",
  "reqId": 1208,
  "result": 0,
  "codeInfos": [
    {
      "codeNum": "xxxxxxxx",
      "codeType": "code128",
      "x": [10, 200, 205, 16],
      "y": [20, 32, 256, 251]
    }, {
      "codeNum": "xxxxx",
      "codeType": "QR",
      "x": [110, 300, 305, 118],
      "y": [22, 30, 285, 259]
    }
  ]
}
```

返回参数说明

属性名称	类型	含义
func	字符串	被执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
codeInfos	对象数组	表示多个识别到的条码信息。
codeNum	字符串	条码信息
codeType	字符串	条码类型
x	数字数组	条码所在图像区域的四个点的X坐标，依次为左上、右上、右下、左下。
y	数字数组	条码所在图像区域的四个点的Y坐标，依次为左上、右上、右下、左下。

OCR文字识别：FileOCR

识别指定图片上的文字，识别成功后在本地生成指定格式的文件。

请求格式

```
{
  "func": "FileOCR",
  "reqId": 123456,
  "language": "Simplified chinese+English",
  "extName": ".docx",
  "detectTextOrientation": false,
  "mode": "base64;path",
  "imagePath": [
    "C:\\Program Files (x86)\\CamSDK\\tmp\\02967dd8\\1634283148.jpg",
    "C:\\Program Files (x86)\\CamSDK\\tmp\\02967dd8\\1634283148.jpg"
  ],
  "zones": [
    [{left: 100, top: 150, right: 500, bottom: 450, absolute: 0, type: 0},
    {.....}, .....],
    [{left: 160, top: 180, right: 560, bottom: 480, absolute: 0, type: 0},
    {.....}, .....]
  ]
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
imagePath	字符串或字符串数组	要识别的图片的全路径，包含后缀名。 只有一个文件，可以为字符串；有多个文件时，为字符串数组。
zones	对象数组（一维或二维）	每张图片要识别的区域。一张图片可指定多个识别区域。每个识别区域对象由left、top、right、bottom、absolute、type六个属性组成，详见下面的描述。 imagePath为字符串数组时，zones必须是二维数组； imagePath为字符串时，zones必须是一维数组；不使用区域时，可以缺省zones属性。
left	数字	识别区域的左上角距图片左上角的水平方向的位置
top	数字	识别区域的左上角距图片左上角的垂直方向的位置
right	数字	识别区域的右下角距图片左上角的水平方向的位置
bottom	数字	识别区域的右下角距图片左上角的垂直方向的位置
absolute	数字或布尔型	为true或1时，表示left、top、right、bottom四个数字表示的是图片的绝对位置（即图像的像素）；为false或0时，表示left、top、right、bottom四个数字表示的是相对数（0-10000,把图片高度宽高都视为10000）。缺省值为false。
type	数字	识别区域的类型：0文字区；1插图照片区；2表格区；3条码区；4手写文字区。缺省值为0。
language	字符串	要识别的语言类型。可从 GetOcrSupportInfo 的 languages 属性中获取。
extName	字符串	识别结果保存文件的后缀名(文件类型)。可从 GetOcrSupportInfo 的 fileTypes 属性中获取。
indexExtName	数字	指定识别结果保存文件的后缀名的索引号（ GetOcrSupportInfo 返回的 fileTypes 数组的序号）。指定extName时，不用指定indexExtName，extName比indexExtName优先级高
detectTextOrientation	布尔值	是否启用自动文字方向。启用自动文字方向时识别会慢一些。在能确保图像中文字是正向的情况下，可以不开启。
mode	字符串	返回文件的方式: base64; path, 默认为path, 也可同时使用, 中间用;隔开, 如: base64;path

属性名称	类型	含义
filePath	字符串	生成文件全路径；可以不指定，则由SDK自动生成路径。在指定filePath时，因为路径中包含了扩展名，因而不需要extName及indexExtName属性。

返回格式

```
{
  "func": "FileOCR",
  "reqId": 123456,
  "result": 0,
  "filePath": "D:\\release_ws\\tmp\\02a4ca18\\1635403727_3d6c.docx",
  "fileBase64": ".....",
  "mime": "application/vnd.openxmlformats-officedocument.wordprocessingml.document"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	被执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
filePath	字符串	返回OCR识别结果保存的文件的全路径。
fileBase64	字符串	OCR识别结果文件的base64串。仅在请求参数mode中有base64时才会出现此属性。
mime	字符串	MIME类型。仅在请求参数mode中有base64时才会出现此属性。

智能识别：AiRecognize

用于识别身份证、驾驶证、行驶证等固定格式文档的关键信息。

请求格式

```
{
  "func": "AiRecognize",
  "reqId": 123457,
  "imagePath": "C:\\Program Files (x86)\\CamSDK\\tmp\\02967dd8\\1634283148.jpg",
  "orcInfo": true
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
imagePath	字符串	要识别的图片的全路径，包含后缀名。
orcInfo	布尔型	是否识别图片的关键信息，默认false。 不识别关键信息时，只会识别图片的类型（身份证、驾驶证、行驶证等）

返回格式

```
{
  "func": "AiRecognize",
  "reqId": 123457,
  "result": 0,
  "layoutID": 27,
  "layoutName": "行驶证",
  "info": {
    "车辆类型": "小型轿车",
    "车辆识别代号": "HHGHH55577*****69",
    "车牌号码": "京AA5599",
    "发动机号码": "1122****",
    "发证日期": "20180614",
    "品牌型号": "讴歌牌GHA1150JAD5B",
    "使用性质": "非营运",
    "所有人": "王****",
    "证件类型": "机动车行驶证",
    "住址": "北京市石景山区",
    "注册日期": "20180614"
  }
}
```

返回格式中，根据不同文档类型，info中会有不同的属性。如果请求参数中orcInfo为false，则返回格式如下。

```
{
  "func": "AiRecognize",
  "reqId": 123457,
  "result": 0,
  "layoutID": 27,
  "layoutName": "行驶证"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	被执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
layoutID	数字	文档的布局ID（即文档的类型代号）
layoutName	字符串	文档的布局名称（即文档的类型名称）。
info	对象	文档的关键信息，不同的文档有不同的属性。 只有请求参数中orclInfo为true时才会有info属性返回。

OCR识别或转换为txt文件：OcrConvert2Txt

把指定的文件OCR识别或转换为txt文件。支持pdf、tif、word、excel文件。此接口不要求在其他平台中实现。

请求格式

```
{
  "func": "OcrConvert2Txt",
  "reqId": 123458,
  "language": "Simplified chinese+English",
  "detectTextOrientation": true,
  "filePath": [
    "F:\\Test\\zbtq\\abc.pdf",
    "F:\\Test\\zbtq\\2报告.doc"
  ]
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
language	字符串	要识别的语言类型。可从 GetOcrSupportInfo 的 languages 属性中获取。
detectTextOrientation	布尔	是否启用自动文字方向。启用自动文字方向时识别会慢一

属性名称	值类型	含义
		在能确保图像中文字是正向的情况下，可以不开启。
filePath	字符串或字符串数组	要识别或转换的文件的全路径名。 只有一个文件，可以为字符串；有多个件时，为字符串数组。

返回格式

```
{
  "func": "OcrConvert2Txt",
  "reqId": 123458,
  "result": 0,
  "outTxtPaths": [
    "C:\\Users\\CHJ\\AppData\\Local\\Temp\\CamSDK\\03517108\\abc_4823.txt",
    "C:\\Users\\CHJ\\AppData\\Local\\Temp\\CamSDK\\03517108\\2报告_5671.txt",
  ]
}
```

返回参数说明

属性名称	类型	含义
func	字符串	被执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
outTxtPaths	字符串数组	识别或转换输出的txt文件的全路径。

文件处理

文件HTTP上传：UploadHttpFile

以HTTP或HTTPS方式，上传文件到web服务器上。上传成功后，返回web服务器的回应信息。

请求格式

```
{
  "func": "UploadHttpFile",
```

```

"reqId": 1666057336437,
"url": "https://www.xxxxx.com/file/upload",
"cookie": ".....",
"files": {
  "File1":
"C:\\Users\\CHJ\\AppData\\Local\\Temp\\CamSDK\\02fa0590\\1666057281.jpg",
  "File2":
"C:\\Users\\CHJ\\AppData\\Local\\Temp\\CamSDK\\02fa0590\\1666057285.jpg"
},
"params": {
  "Param1": "value1",
  "Param2": "value2"
}
}

```

上面的请求，经过CamSdkServer处理后，会向www.xxxxx.com服务器发出如下的http POST请求报文：

```

POST /file/upload HTTP/1.1
Host: www.xxxxx.com
Accept: */*
Content-Length: 144338
Content-Type: multipart/form-data; boundary=-----
-4a89490dbd6fb820

-----4a89490dbd6fb820
Content-Disposition: form-data; name="File1"; filename="1666057281.jpg"
Content-Type: image/jpeg

.....JFIF.....C.....
.....F.F.h.b....|.....ofe....F`2.#.8?.r..r>..Q....^m....L...._.8<E....8...p...
..]...I....
-----4a89490dbd6fb820
Content-Disposition: form-data; name="File2"; filename="1666057285.jpg"
Content-Type: image/jpeg

.....JFIF.....C.....
.....aJ...5s.t.;...h....h.K.+.....".E..C.....|Q@...
-----4a89490dbd6fb820
Content-Disposition: form-data; name="Param1"

value1
-----4a89490dbd6fb820
Content-Disposition: form-data; name="Param2"

value2
-----4a89490dbd6fb820--

```

也可使用文件Base64的方式，但Base64的方式一次只能上传一个文件。如下：

```
{
  "func": "UploadHttpFile",
  "reqId": "1666057336438",
  "url": "https://www.xxxx.com/admin/file/upload",
  "cookie": ".....",
  "fieldName": "UserHeadPicture",
  "fileName": "myHeadPic.jpg",
  "imageBase64": ".....",
  "params": {
    "userId": "12345678",
    "userName": "Jack"
  }
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
url	字符串	上传文件的URL。
cookie	字符串	对于上传时需要认证的服务器，可以提供认证之后的cookie串；如果不提供，则在上传时会弹出登录窗口来让用户登录（但要求安装了cef3组件，即bin目录下有cef3的子目录及cefLogin.exe等相关文件）。此参数可省略。
files	Map对象	要上传的多个文件，每个文件在Map对象中用一个键值对说明，其中“键”为这个文件在HTTP上传报文中的form-data的name值；“值”为这个文件的全路径。 files与fileBase64必需指定一个。指定了files属性时，会忽略对fileBase64、fieldName、fileName这三个属性。
fileBase64	字符串	要上传的文件的Base64编码串。Base64编码串会先保存为fileName的文件，之后再向HTTP服务器进行POST提交。 与files必需指定一个，若同时指定时，fileBase64将被忽略。
fieldName	字符串	与fileBase64配合使用。用于指明fileBase64这个文件在HTTP上传报文中的form-data的name值。
fileName	字符串	与fileBase64配合使用。用于指明fileBase64这个文件的文件名，只需要文件名与扩展名，不需要全路径。
params	Map对象	上传时需要附加的其他参数，可包含多个参数。此参数可省略。

返回格式

```
{
  "func": "UploadHttpFile",
  "reqId": "1666057336437",
  "result": 0,
  "responseCode": 200,
  "response": "{\"viewUrl\":\"/file/download?f=\\\",\\\"result\\\":\\\"success\\\",\\\"files\\\":\n[\\\"fileName\\\":\\\"File1\\\",\\\"originalFilename\\\":\\\"1666057281.jpg\\\",\\\"serverFileName\\\":\\\"1666057337471_0.jpg\\\"},\n{\\\"fileName\\\":\\\"File2\\\",\\\"originalFilename\\\":\\\"1666057285.jpg\\\",\\\"serverFileName\\\":\\\"1666057337471_0.jpg\\\"}],\\\"parameters\\\":{\\\"Param2\\\":\n[\\\"value2\\\"],\\\"Param1\\\":[\\\"value1\\\"]}}\",
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
responseCode	数字	上传时，web服务器返回的HTTP代码，200表示成功。
response	字符串	上传时，web服务器返回回应结果。如果是json串，需要再自行解析。

文件FTP上传： UploadFtpFile

把上传文件到FTP服务器上。

请求格式

```
{
  "func": "UploadFtpFile",
  "reqId": 1666062483050
  "localFilePath":
"C:\\\\Users\\CHJ\\AppData\\Local\\Temp\\CamSDK\\02b67920\\1666062481.jpg",
  "remoteFilePath": "ftp://<ip address>/myfile/1666062481.jpg",
  "userName": "....",
  "password": ".....",
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
localFilePath	字符串	要上传文件的本地文件路径。
remoteFilePath	字符串	文件上传到FTP服务器上的URL。
userName	字符串	连接FTP服务器的用户名。
password	字符串	连接FTP服务器的密码。

返回格式

```
{
  "func": "UploadFtpFile",
  "reqId": 1666062483050,
  "remoteFilePath": "ftp://123.57.223.145/myfile/1666062481.jpg",
  "result": 0
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
remoteFilePath	字符串	文件上传到FTP服务器上的URL。

合并文件：MergeFile

用于将多个jpg文件合并成一个pdf文件（或tif文件、或jpg文件）。

请求格式

```
{
  "func": "MergeFile",
  "reqId": "223388",
  "fileType": 0,
  "mode": "path;base64",
  "imagePath": [
    "F:\\CamSDK\\release_ws\\tmp\\03629548\\pic1_0.jpg",
    "F:\\CamSDK\\release_ws\\tmp\\03629548\\pic1_1.jpg"
  ]
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
fileType	数字	合并文件的类型。0-pdf; 1-tif; 2-jpg 合并为jpg或tif文件时，输入的文件要求是jpg文件；合并为pdf文件时，输入文件可以是jpg或pdf。合并为jpg时，要指定两个输入jpg文件，超过将被忽略。
imagePath	字符串数组	要合并的（输入）文件的的全路径的数组，至少包含一项或两项(合并为jpg时)。
filePath	字符串	合并输出的文件路径，可以是文件名、可以是全路径，也可以缺省。缺省时会自动生成输出文件名。
mode	字符串	返回文件的方式: base64; path, 默认为path, 也可同时使用,中间用“;”隔开，如: base64;path
combineType	数字	合并为jpg时，指定合并时两个图片的布局方式（0-9）
offsetX	数字	合并为jpg时 且combineType为0时，指定第2个图片的位置的偏移量x
offsetY	数字	合并为jpg时 且combineType为0时，指定第2个图片的位置的偏移量y

返回格式

```
{
  "func": "MergeFile",
  "reqId": "223388",
  "result": 0,
  "filePath": "F:\\CamSDK\\release_ws\\tmp\\03629548\\56856541.pdf",
  "fileBase64": "JVBERi0xLjcKJEtjz9IKNCAw....."
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
filePath	字符串	合并后的文件的全路径。
fileBase64	字符串	合并后的文件的base64数据。

外设接口

读卡信息：ReadIDCardInfo

读取身份证读卡器信息。

请求格式

```
{
  "func": "ReadIDCardInfo",
  "reqId": "223389",
  "idType": 0,
  "fpNumber": 1,
  "simulateImageType": 1,
  "mode": "base64;path"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
idType	数字	ID卡的类型：0-读身份证信息；1-读社保卡信息；2-读银行卡信息；
fpNumber	数字	返回指纹的个数，缺省值为1。0-不返回指纹信息；1-返回第一个指纹；2-返回第一和第二个指纹。此属性仅对身份证有效。
simulateImageType	数字	返回身份证模拟照片的类型。0-不生成；1-生成不合并；2-生成左右合并；3-生成上下合并；4-生成上下合并有空隙。此属性仅对身份证有效。
mode	字符串	返回身份证模拟照片的方式: base64; path, 默认为path, 也可同时使用,中间用;隔开, 如: base64;path 此属性仅对身份证有效。

返回格式

读取身份证返回信息格式：

```
{
  "func": "ReadIDCardInfo",
  "reqId": "223387",
  "result": 0,
  "cardInfo": {
    "id": "430xxxxxxxxxxxxxx",
    "name": "xxx",
    "folk": "汉",
    "gender": "男",
    "address": "长沙市xxxxxx",
    "birth": "19851025",
    "startDate": "20091013",
    "endDate": "20291013",
    "fingerFeature": [
      {
        "featureBase64": "QwFEeGELWAAAAAAAAA.....",
        "fpPath": "C:\\windows\\Temp\\CamSDK\\02a31db0\\1640743300",
        "type": 11
      }
    ],
    "imageBase64": "",
    "imagePath": "C:\\windows\\Temp\\CamSDK\\02a31db0\\1628740497.jpg",
```

```
    "issue": "长沙市xxxxxx分局",
    "mime": "image/jpeg"
  },
  "simulateImage": {
    "paths": [
      "C:\\windows\\Temp\\CamSDK\\02a31db0\\1628740489.jpg",
      "C:\\windows\\Temp\\CamSDK\\02a31db0\\1628740490.jpg",
    ],
    "base64s": [
      "/9j/4AAQSkZJRgAB....",
      "/9j/4AFBCKZHGD EB...."
    ],
    "mime": "image/jpeg"
  }
}
```

读取社保卡返回信息格式：

```
{
  "func": "ReadIDCardInfo",
  "reqId": "",
  "result": 0,
  "cardInfo": {
    "name": "xxx",
    "id": "",
    "gender": "女",
    "folk": "汉",
    "birth": "1999-05-01",
    "address": ""
  }
}
```

读取银行卡返回信息格式：

```
{
  "func": "ReadIDCardInfo",
  "reqId": "",
  "result": 0,
  "idNum": "1234456789"
}
```

返回参数说明

读取身份证的返回参数：

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
cardInfo	对象	身份证的信息。
id	字符串	身份证号码。
name	字符串	姓名
gender	字符串	性别
genderCode	数字	性别代码
folk	字符串	民族，如果是外国人身份证，如果没有folk字段，那就置为空。
folkCode	数字	民族代码
birth	字符串	生日，形如：YYYYMMDD
startDate	字符串	有效期开始日期，形如：YYYYMMDD
endDate	字符串	有效期到期日期，形如：YYYYMMDD

属性名称	类型	含义
imageBase64	字符串	身份证头像base64
mime	字符串	MIME类型，一般是image/jpg
imagePath	字符串	身份证头像的文件全路径
fingerFeature	对象数组	保存指纹的base64数据、特征文件位置、指纹位置的json对象数组
featureBase64	字符串	指纹特征的base64数据
fpPath	字符串	指纹特征码二进制文件全路径。
type	数字	指纹位置代码：11 右手拇指，12 右手食指，13 右手中指，14 右手环指， 15 右手小指，16 左手拇指，17 左手食指，18 左手中指，19 左手环指， 20 左手小指，97 右手不确定指位，98 左手不确定指位，99 其他不确定指位
issue	字符串	发证机关
eName	字符串	英文名，仅外国人身份证才有此属性
version	字符串	版本号，仅外国人身份证才有此属性
passPortID	字符串	通行证号码，仅港澳台身份证才有此属性
time	字符串	时间，仅港澳台身份证才有此属性

属性名称	类型	含义
simulateImage	对象	身份证模拟照片信息。根据请求参数中的simulateImageType与mode不同，会有不同的值。simulateImageType为0时，返回结果中不会有此属性。
paths	字符串数组	身份证模拟照片的路径数组，simulateImageType为合并时，数组只会有一个元素。 否则会有两个，分别表示身份证模拟照片的正面、反面。 在请求参数mode中包含path时才会有此属性。
base64s	字符串数组	身份证模拟照片的Base64串的数组，simulateImageType为合并时，数组只会有一个元素。 否则会有两个，分别表示身份证模拟照片的正面、反面。 在请求参数mode中包含base64时才会有此属性。
mime	字符串	身份证模拟照片的MIME类型，一般是image/jpg

读取社保卡的返回参数：

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
idCardInfo	对象	社保卡的信息。
id	字符串	社保卡卡号
name	字符串	姓名
gender	字符串	性别
folk	字符串	民族
birth	字符串	生日，形如：YYYYMMDD
address	字符串	住址

读取银行卡的返回参数：

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
idNum	字符串	银行卡卡号。

指纹采集：ReadFingerInfo

使用指纹采集功能，需要有指纹设备。

请求格式

```
{
  "func": "ReadFingerInfo",
  "reqId": 1256,
  "timeout": 3000
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
timeout	数字	读取指纹的超时时间（毫秒），不指定时只读取一次。

返回格式

```
{
  "func": "ReadFingerInfo",
  "reqId": 1256,
  "result": 0,
  "fingerFeatureBase64": ".....",
  "fingerImageBase64": ".....",
  "mime": "image/bmp"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
fingerFeatureBase64	字符串	指纹特征的Base64串。
fingerImageBase64	字符串	指纹图像的Base64串。
mime	字符串	MIME类型，一般是image/bmp

指纹比对：FingerMatch

用于比对两个指纹特征是否匹配。

请求格式

```
{
  "func": "FingerMatch",
  "reqId": 1601,
  "fingerFeature1": "QwFEeGELWAAAAAAAAA.....",
  "fingerFeature2": "QwFEeGELWAAAAAAAAA.....",
  "timeout": 5000,
  "threshold": 0.9
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
fingerFeature1	字符串	要比对的指纹的Base64编码串。与fingerFeature2必需两者都指定
fingerFeature2	字符串	要比对的指纹的Base64编码串。
timeout	数字	对比超时时间，单位毫秒。
threshold	浮点数	相似度阈值（0-1.0之间），可省略，默认0.9。

返回格式

```
{
  "func": "FingerMatch",
  "reqId": 1601,
  "result": 0,
  "similarity": 0.92,
  "threshold": 0.9
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功； 23表示指纹匹配不成功（similarity小于请求时指定的threshold）； 其他表示~~ 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
similarity	浮点数	比对得出的的相似度（0-1.0之间）。
threshold	浮点数	请求参数中的相似度阈值（0-1.0之间），复制到返回结果中，便于业务处理。

外接USB按键：ExternalButton

用外接USB按键，可以在按下后触发拍照、拍书、条码识别等动作，方便在大批量拍摄文档时操作。

请求格式

打开外接按键功能：

```
{
  "func": "ExternalButton",
  "reqId": 1801,
  "enable": true,
  "callbackFunc": "CameraCapture",
  "<callbackFunc所指命令的其他属性1>": ".....",
  "<callbackFunc所指命令的其他属性2>": ".....",
  "<callbackFunc所指命令的其他属性n>": ".....",
  ...
}
```

打开外接设备成功后，连接的USB按键会亮红灯，表示USB按键已启用；当按下USB按键时，会亮一下绿灯。

不再使用USB按键时，可能过下面的命令关闭外接按键功能：

```
{
  "func": "ExternalButton",
  "reqId": 1802,
  "enable": false
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
enable	布尔型	true表示开启外接按键功能；false表示停用外接按键功能。为false时不需再指定下面的其他属性。
callbackFunc	字符串	按键按下时要触发的功能。目前支持：Notify、CameraCapture、CameraCaptureBase64、CameraCaptureBook、RecogBarCode。其中Notify是指在按键按下时，发一条消息进行通知，通知消息的格式见后表说明。 CameraCapture表示在按键按下时，自动进行一次 CameraCapture 命令的操作。 其他类似。
其他属性		CameraCapture、CameraCaptureBase64、CameraCaptureBook、RecogBarCode需要的其他属性，参考相应的命令。比如，CameraCapture还需要指定devNum、mode等属性。

返回格式

```
{
  "func": "ExternalButton",
  "reqId": 1801,
  "enable": true,
  "result": 0
}
```

在启用ExternalButton功能并且指定callbackFunc为Notify时，当按键按下时，服务端会发回一条以下格式的消息。

```
{
  "func": "Notify",
  "event": "OnUsbKeyPress",
  "reqId": <ExternalButton开启时的reqId>,
  "time": "<按键按下的系统时间的时间戳>",
  "result": 0
}
```

其中OnUsbKeyPress表示是按键按下事件。

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
enable	布尔型	布尔型。复制的请求命令中的enable属性。

人脸识别

人像比对 FaceMatch

人像比对可对比两张头像照片；或者头像照片与指定的摄像头实时人像进比对。

请求格式

两张头像比对时，直接指定头像文件全路径：

```
{
  "func": "FaceMatch",
  "reqId": "223396",
  "image1Path": "D:\\\\Damon.jpg",
  "image2Path": "D:\\\\chj.jpg"
}
```

也可以指定头像的Base64串，头像1与头像2，即可指定文件路径，也可指定Base64串，两种方式指定一种即可：

```
{
  "func": "FaceMatch",
  "reqId": "223397",
  "image1Base64": ".....",
  "image2Base64": "....."
}
```

头像照片与指定的摄像头实时人像进比对：


```
{
  "func": "FaceMatch",
  "reqId": "223398",
  "image1Path": "D:\\Jean.jpg",
  "image2DevNum": 2,
  "sendMatchedImage": 1,
  "playWav": "recognitionPass.wav",
  "liveFace": 1,
  "matchedSimilarity": 80,
  "stopMode": 1,
  "runSecond": 0
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
image1Path	字符串	比对的头像1的全路径。与image1Base64需至少指定一个，image1Path优先级高。
image1Base64	字符串	比对的头像1的Base64串。与image1Path需至少指定一个，image1Path优先级高。
image2Path	字符串	比对的头像2的全路径。与image2Base64、image2DevNum需至少指定一个，image2Path优先级最高。
image2Base64	字符串	比对的头像2的Base64串。与image2Base64、image2DevNum需至少指定一个，image2Path优先级最高。
image2DevNum	数字	从指定的摄像头获取头像2进行比对。与image2Base64、image2DevNum需至少指定一个，image2Path优先级最高。
sendMatchedImage	数字	在检测匹配时，在回应的json中包含匹配的视频图像(1/0)，1表示包含。
matchedSimilarity	数字	比对人脸相似度>=这个值(0-100)表示匹配成功, 设置liveFace为1时,匹配成功才会进行活体检测
liveFace	数字	是否开启活体检测(1/0)，1表示启活体检测。
playWav	字符串	在检测匹配时，要播放的wav文件路径。相对路径(安装目录下的media子目录)或全路径。
stopMode	数字	停止方式： 0-手动停止，要停止时发送一条stop属性为1的FaceMatch命令； 1-匹配成功活检通过(若设置) 或 超过runSecond指定的秒数自动停止； 2-定时运行runSecond指定的秒数后停止。可省略，默认值为1。
runSecond	数字	检测运行的时长(秒数, 停止方式为1或2时有效)。
stop	数字	停止之前的人像比对时，指定此属性为1。其他情况可省略此参数。

返回格式

两张头像直接比对时，返回：

```
{
  "func": "FaceMatch",
  "reqId": "223396",
  "similarity": 26
}
```

启动或停止人像比对时，会立即返回以下消息，表示启动或停止人像比对正常执行：

```
{
  "func": "FaceMatch",
  "reqId": "223397",
  "result": 0
}
```

头像照片与指定的摄像头实时人像进比对时，会每隔2秒，发一次结果：

```
{
  "func": "FaceMatch",
  "reqId": "223398",
  "result": 0,
  "checkResult": 7,
  "similarity": 62,
  "matchedSimilarity": 80,
  "isLiveFace": 0,
  "liveScore": 0,
  "liveFace": 1
}
```

头像照片与指定的摄像头实时人像对时成功时，返回：

```
{
  "func": "FaceMatch",
  "reqId": "223398",
  "result": 0,
  "matched": 1,
  "checkResult": 15,
  "similarity": 86,
  "matchedSimilarity": 80,
  "isLiveFace": 1,
  "liveScore": 92,
  "liveFace": 1,
  "stoped": 0,
  "stopMode": 1,
  "imageBase64": "....."
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
matched	数字	是否匹配通过(1/0)。
checkResult	数字	比对时的检测结果，从低到高按位表示：是否检测到彩色人脸，彩色人脸特征比对是否通过，是否检测到红外人脸，是否活体。
similarity	数字	人脸比对结果的相似度，0-100。
isLiveFace	数字	检测到是否是活体。1表示是活体。
liveScore	数字	活体检测的分数，0-100，数字越大表示越准确。
stoped	数字	检测是否已停止(1/0)。
liveFace	数字	请求参数中指定的liveFace参数。复制到返回结果中方便客户端处理。
matchedSimilarity	数字	请求参数中指定的matchedSimilarity参数。复制到返回结果中方便客户端处理。
stopMode	数字	请求参数中指定的stopMode参数。复制到返回结果中方便客户端处理。
imageBase64	字符串	在检测匹配时的视频图像。请求参数中指定sendMatchedImage为1才有。

其他功能

获取系统版本号：GetVersion

获取CamSDK的系统版本号。CamSDK每次发布，都会有不同的版本号。版本号的格式为：yyyyMMdd。

请求格式

```
{
  "func": "GetVersion",
  "reqId": "223500"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "GetVersion",
  "reqId": "223500",
  "result": 0,
  "sysVersion": 20210930
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
sysVersion	数字	CamSDK的系统版本号。CamSDK每次发布，都会有不同的版本号。版本号的格式为：yyyyMMdd。

获取支持的命令列表：GetCommandList

获取CamSDK支持的命令列表。

请求格式

```
{
  "func": "GetCommandList",
  "reqId": "223501"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

```
{
  "func": "GetCommandList",
  "reqId": "223501",
  "result": 0,
  "commandList": [
    "CameraAutoCapture",
    "CameraCapture",
    "CameraCaptureBase64",
    "CameraCaptureBook",
    "CloseCamera",
    "FaceMatch",
    "FileToBase64",
    "FingerMatch",
    "GetCameraInfo",
    "GetCameraVideoBuff",
    "GetCommandList",
    "GetOcrSupportInfo",
    "GetVersion",
    "MergeFile",
    "OpenCamera",
    "ReadFingerInfo",
    "ReadIDCardInfo",
    "RecogBarCode",
    "RemoveFile",
    "SetCameraImageInfo",
    "SetCameraInfo",
    "UploadHttpFile"
  ]
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
commandList	字符串数组	CamSDK支持的命令的列表。

获取OCR支持信息：GetOcrSupportInfo

获取OCR支持信息，包括OCR支持的语言列表、OCR支持的输出文件格式。

请求格式

```
{
  "func": "GetOcrSupportInfo",
  "reqId": "223502"
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号， 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

正常返回格式：

```
{
  "func": "GetOcrSupportInfo",
  "reqId": "223502",
  "fileTypes": [
    ".docx",
    ".txt",
    ".xml",
    ".html",
    ".odt",
  ]
}
```

```
".rtf",
".pdf",
".xps",
".xlsx",
".epub",
".pptx"
],
"languages": [
"Afaan Oromo",
"Afrikaans",
"Albanian",
"Asturian",
"Aymara",
"Azeri Latin",
"Balinese",
"Basque",
"Bemba",
"Bikol",
"Bislama",
"Bosnian Cyrillic",
"Bosnian Latin",
"Brazilian",
"Breton",
"Bulgarian",
"Catalan",
"Cebuano",
"Chamorro",
"Corsican",
"Croatian",
"Czech",
"Danish",
"Dutch",
"English",
"Esperanto",
"Estonian",
"Faroeese",
"Farsi",
"Fijian",
"Finnish",
"French",
"Frisian",
"Friulian",
"Galician",
"Ganda",
"German",
"Greek",
"Greenlandic",
"Haitian Creole",
"Hani",
"Hiligaynon",
"Hungarian",
"Icelandic",
"Ido",
"Ilocano",
"Indonesian",
"Interlingua",
"Irish Gaelic",
"Italian",
```


"Japanese",
"Japanese+English",
"Javanese",
"Kapampangan",
"Kazakh Cyrillic",
"Kikongo",
"Kinyarwanda",
"Korean",
"Kurdish",
"Latin",
"Latvian",
"Lithuanian",
"Luba",
"Luxembourgish",
"Macedonian",
"Madurese",
"Malagasy",
"Malay",
"Maltese",
"Manx",
"Maori",
"Mayan",
"Mexican Spanish",
"Minangkabau",
"Moldovan",
"Mongolian Cyrillic",
"Nahuatl",
"Nigerian Pidgin",
"Norwegian",
"Norwegian Nynorsk",
"Numeric",
"Nyanja",
"Occitan",
"Papiamentu",
"Polish",
"Portuguese",
"Quechua",
"Rhaeto Romance",
"Romanian",
"Rundi",
"Russian",
"Samoan",
"Sardinian",
"Scottish Gaelic",
"Serbian",
"Serbian Latin",
"Simplified chinese",
"Simplified chinese+English",
"Slovak",
"Slovenian",
"Sotho",
"Spanish",
"Sundanese",
"Swahili",
"Swedish",
"Tagalog",
"Tahitian",
"Tatar Latin",

```
    "Tetum",
    "Tokpisin",
    "Tongan",
    "Traditional chinese",
    "Traditional chinese+English",
    "Tswana",
    "Turkmen Latin",
    "Turkish",
    "Ukrainian",
    "Uzbek Latin",
    "Waray",
    "Welsh",
    "Wolof",
    "Xhosa",
    "Zapotec",
    "Zulu"
  ],
  "result": 0
}
```

没有安装OCR库时返回：

```
{
  "func": "GetOcrSupportInfo",
  "reqId": "223502",
  "result": 14,
  "errorMsg": "IRIS\\CmOCRSDK_IRIS.dll load fail, GetLastError()=183"
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
fileTypes	字符串数组	OCR所支持的输出文件类型的列表。
languages	字符串数组	OCR所支持的语言列表。

获取错误代码信息对照表：GetErrorCodeList

获取系统的全部错误代码信息对照表。

请求格式

```
{
  "func": "GetErrorCodeList",
  "reqId": 16578
}
```

请求参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串或数字	发出请求的编号，服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。

返回格式

正常返回格式：

```
{
  "func": "GetErrorCodeList",
  "reqId": 16578,
  "result": 0,
  "errorCodes": {
    "-99": "没有找到DLL中的函数",
    "0": "成功",
    "1": "未知错误",
    "10": "没有找到任何(或指定)摄像头",
    "10000": "没有人脸授权",
    "10001": "没有DETECTFACE",
    "10002": "服务没开启",
    "10003": "加载model失败",
    "10004": "监测人脸失败",
    "10005": "没有初始化",
    "10006": "加载摄像机DLL失败",
    "10007": "设备没连接",
    "10008": "打开摄像头失败",
    "10009": "超时",
    "10010": "加载CMFACE3失败",
    "10011": "配置文件格式错误",
    "10012": "加载配置文件失败",
    "10013": "保存图片时失败",
    "10014": "非活体",
    "1025": "FTP连接失败",
    "1026": "FTP断开失败",
    "1027": "FTP路径有误",
    "1028": "FTP上传失败",
  }
}
```

"1029": "HTTP连接失败",
"1030": "文件找不到",
"1031": "认证失败",
"1032": "删除文件失败",
"1033": "文件打开失败",
"1034": "EXE程序启动失败",
"1035": "代理执行DLL函数失败",
"1036": "执行超时",
"11": "设备未开启",
"12": "未初始化",
"1281": "加载证件识别失败",
"1297": "加载条码识别失败",
"13": "已存在",
"14": "DLL加载失败",
"15": "没有找到",
"1537": "识别参数错误",
"1538": "不支持智能连拍",
"1539": "没检测到手指",
"1540": "CLR识别失败",
"16": "加载图像失败",
"17": "功能未配置开启",
"1793": "命令不支持",
"1794": "JSON格式无效",
"1795": "websocket连接失败",
"1796": "websocket发送失败",
"1797": "回调命令不支持",
"18": "APP启动失败",
"19": "设备未连接.",
"1900": "转换没有输入文件",
"1901": "转换没有输出文件",
"2": "属性设置错误",
"20": "社保卡片复位失败",
"2000": "文件保存类型错误",
"21": "无身份证背景图",
"2100": "不支持的Media_Type",
"2101": "Codec not found AV_CODEC_ID_H264",
"2102": "Could not allocate codec context",
"2103": "Could not open codec",
"2104": "av_packet_alloc失败",
"2105": "av_packet_from_data error",
"2106": "avcodec_send_packet error",
"2107": "av_frame_get_buffer failed",
"2108": "sws_getContext failed",
"2109": "sws_scale failed",
"22": "空信息(没读取或读取失败)",
"2200": "没有人脸授权",
"2201": "算法初始化失败",
"2202": "读取头像文件时图像解析失败",
"2203": "没有检测到人脸",
"2204": "检测人脸失败",
"2205": "提取人脸特征码失败",
"23": "指纹不匹配",
"24": "人脸不匹配",
"25": "没有找到支持的ID卡设备",
"257": "开启设备失败",
"258": "开启视频失败",
"259": "设置分辨率失败",
"260": "自动对焦失败",

```
"261": "拍照空指针",  
"262": "保存图像失败",  
"263": "没有授权",  
"264": "写设备数据失败",  
"265": "无授权已反初始化SDK",  
"266": "不支持灯光",  
"267": "设备不支持拍书",  
"268": "权限受限",  
"269": "打开设备时存储空间不足",  
"3": "初始化错误",  
"4": "数组越界",  
"5": "空指针",  
"6": "参数错误",  
"65": "未读取到证件信息",  
"7": "无效的路径",  
"8": "不支持",  
"9": "使用中,忙"  
}  
}
```

返回参数说明

属性名称	类型	含义
func	字符串	要执行的功能的名称
reqId	字符串 或数字	发出请求的编号。 服务端在返回结果时会附带这个reqId属性，这样客户端可通过这个属性来区分不同的请求。
result	数字	返回结果，0表示成功；其他表示 错误码。 成功后没有errorMsg属性，其他错误码时，会有errorMsg属性表示错误信息。
errorCodes	对象	系统的全部错误代码信息对照表。其中key为错误代码，value为对应的错误信息。