



武汉视跃科技有限公司

gb/t28181 设备端 SDK(android 版)

接
口
说
明

2019 年 9 月

目录

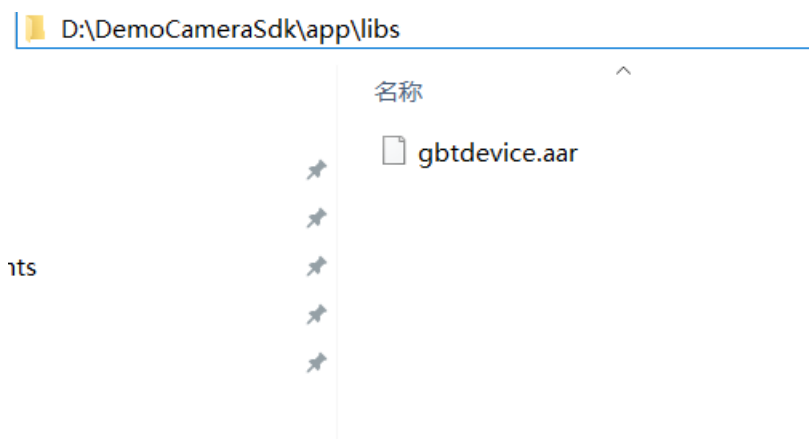
1 sdk 使用说明.....	2
2 GB/T 28181 设备类接口.....	3
1.1 设置本设备端的多媒体格式.....	4
1.2 设置字幕叠加.....	5
1.3 清除字幕叠加.....	6
1.4 设置相机图像预览的方向.....	7
1.5 将设备注册到 gb/t28181 服务器.....	7
1.6 将设备从 gb/t28181 服务器上下线（反注册）.....	7
1.7 设备开启音视频编码与转发.....	7
1.8 设备停止音视频编码与转发.....	8
1.9 发送（更新）设备的位置.....	8
1.10 上报报警信息.....	8
3 接口调用流程.....	10
4 代码示例.....	10

1 sdk 使用说明

本 sdk 底层采用 native c++编写，由应用层（java 层）和 jni 层（native c++）组成，jni 层提供的是基础的功能 api 函数。应用层对其进行封装，并结合 android 原生设备相关的 sdk 功能对其进行了功能的组合与协作，实现了功能全面的 gbt 设备端应用场景，本文档提供的是应用层 sdk。

Sdk 以 module 形式提供，为文件名为 gbtdevice.aar 的 aar 包，使用方法如下：

1. 将 gbtdevice.aar 文件拷贝在 app 下的 libs 文件夹里，如下图：



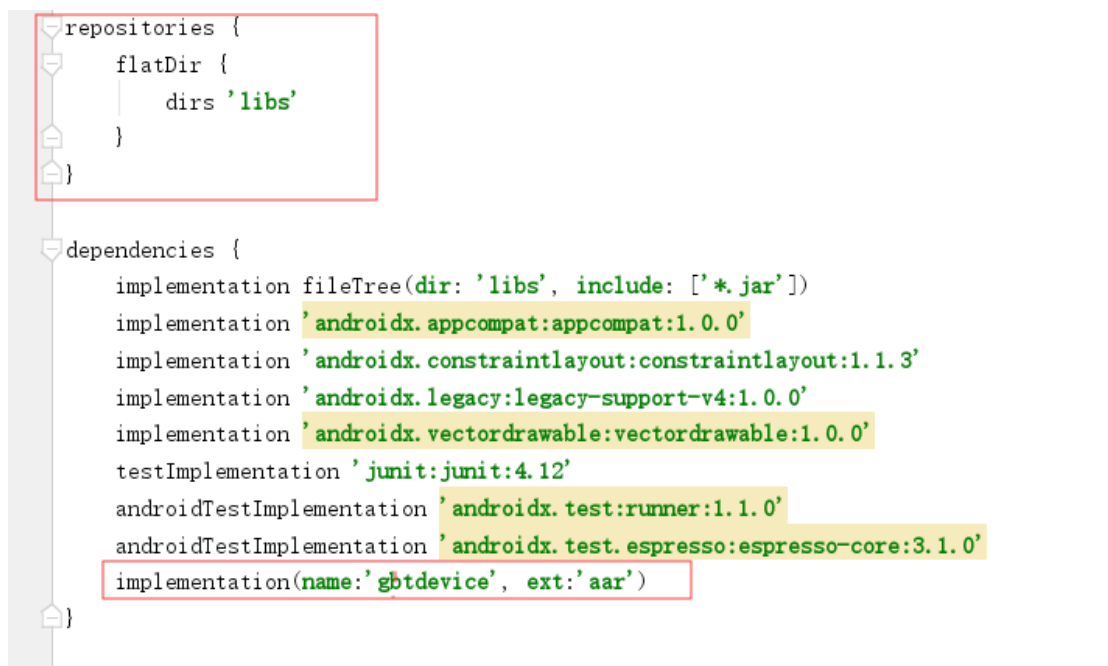
2. 在 app 的 build.gradle 里添加对本 sdk 依赖，需要先在 build.gradle 里增加：

```
repositories {
    flatDir {
        dirs 'libs'
    }
}
```

同时，在 dependencies 里增加如下一行：

```
implementation(name:'gbtdevice', ext:'aar')
```

最终 build.gradle 如下图（红色框为增加部分）：



这样即完成了 app 对本 sdk 的引入。

2 GB/T 28181 设备类接口

类 CGbtMedia 是核心的应用类，实现了设备端 gb28181 的所有功能。

1.1 设置本设备端的多媒体格式

接口原型: `public boolean setFormat(MFormat fmt)`

接口参数:

fmt (in)	设置的多媒体格式, 关于类型 <code>MFormat</code> 说明见本节下面
返回参数	<code>False/true</code> -- 失败/成功

接口说明: 本函数设置设备的音视频输出格式。`MFormat` 为多媒体格式结构体, 其定义如下:

```
public class MFormat {

    public int codec;           // 视频编码
    public int width;          // 视频高
    public int height;         // 视频宽
    public int fps;            // 帧率
    public int bitrate;        // 视频编码码率
    public int gop;            // 关键帧间隔, 不设置默认为 1s

    public int audioCodec;     // 音频编码
    public int channels;       // 通道数
    public int sampleRate;     // 采样率
}
```

其中 `codec` 对应的视频编码, 其取值范围为:

属性名称	类型	说明
<code>MCODEC_H264</code>	<code>int</code>	视频采用 H264 编码
<code>MCODEC_HEVC</code>	<code>int</code>	视频采用 H265 编码

`audioCodec` 对应的音频编码, 其取值范围为:

属性名称	类型	说明
<code>MCODEC_G711A</code>	<code>int</code>	音频采用 G711A 编码
<code>MCODEC_G711U</code>	<code>int</code>	音频采用 G711U 编码
<code>MCODEC_AAC</code>	<code>int</code>	音频采用 AAC 编码

创建多媒体结构对象示例代码:

```
MFormat fmt = new MFormat();
fmt.width = 640;
fmt.height = 480;
fmt.codec = MediaType.MCODEC_H264;
fmt.fps = 25;
fmt.gop = 1;
fmt.bitrate = 1024*1024;

fmt.audioCodec = MediaType.MCODEC_G711U;
fmt.channels = 1;
fmt.sampleRate = 8000;
```

1.2 设置字幕叠加

接口原型: `public boolean setOsd(int index, OsdOverlay overlay)`

接口参数:

index (in)	叠加字幕的序号 (sdk 支持多层字幕叠加, 通过该参数确定那层字幕)
overlay (in)	叠加字幕的结构体, 关于 OsdOverlay 说明见本节后面
返回参数	False/true -- 失败/成功

接口说明: 本函数的功能是对采集的视频进行字幕, 时间戳或图片叠加输出, 字幕叠加类为 OsdOverlay, 其主要属性和说明如下:

属性名称	类型	说明
<code>font</code>	OsdFont	字体参数类
<code>color</code>	<code>int</code>	字体颜色
<code>flags</code>	<code>int</code>	设置 <code>ANTI_ALIAS_FLAG</code> 表示框锯齿
<code>background</code>	<code>int</code>	背景颜色
<code>alpha</code>	<code>int</code>	字体透明度, 取值范围 0-255, 0 表示完全透明, 即不可见
<code>transparent</code>	<code>boolean</code>	背景是否透明
<code>type</code>	<code>int</code>	叠加类型 <code>TYPE_TEXT</code> 表示静态 osd 文本 <code>TYPE_DATETIME</code> 表示时间戳叠加 <code>TYPE_IMAGE</code> 图片叠加
<code>text</code>	String	当为文本时, 该属性表示文本内容, 当为时间戳时, 此属性确定时间戳显示的格式, 比如 yy-MM-dd HH:mm:ss
<code>top</code>	<code>int</code>	显示位置起点的 y 值, 应该小于屏幕分辨率的高度
<code>left</code>	<code>int</code>	显示位置起点的 x 值, 应该小于屏幕分辨率的宽度

关于字体结构 OsdFont 的定位和说明如下:

```
public static class OsdFont
{
    public int size;           // 字体大小, 取值范围: [12, 80]
    public boolean italic;    // > 0 表示 斜体
    public boolean bold;     // >0 表示粗体
    public String name;      // 字体名称, 为空或者 null 表示默认字体

    public OsdFont ()
    {
        this.size = 16;
    }
}
```

```

        this.italic = false;
        this.bold = false;
        this.name = "";
    }
}

```

创建文字叠加示例代码:

```
String osdText = "武汉视跃科技-4G 智能联网监控";
```

```
OsdOverlay overlay = new OsdOverlay();
```

```
overlay.font.size = 36;
```

```
overlay.font.italic = true;
```

```
overlay.color = Color.rgb(0, 255, 0);
```

```
overlay.flags = OsdOverlay.ANTI_ALIAS_FLAG;
```

```
overlay.background = Color.rgb(127, 127, 0);
```

```
overlay.alpha = 127;
```

```
overlay.transparent = false;
```

```
overlay.text = osdText;
```

```
overlay.left = 0;
```

```
overlay.top = 0;
```

```
gbtdevice.setOsd(0, overlay);
```

创建时间戳叠加示例代码:

```
OsdOverlay dateOverlay = new OsdOverlay();
```

```
dateOverlay.setDateTime("yyy-MM-dd HH:mm:ss");
```

```
dateOverlay.top = 100;
```

```
dateOverlay.left = 20;
```

```
dateOverlay.font.size = 36;
```

```
dateOverlay.color = Color.rgb(0, 0, 255);
```

```
gbtdevice.setOsd(1, dateOverlay);
```

1.3 清除字幕叠加

接口原型: `public void clearOsd()`

接口参数: 无

接口说明: 清除对视频中叠加的所有字幕。

1.4 设置相机图像预览的方向

接口原型: `public void setDisplayOrientation(int degree)`

接口参数:

degree (in)	设备的相机采集图像的预览旋转的角度, 设置 0, 90,180,270 时才有效
-------------	--

接口说明: 该接口用来设置设备端采集的视频本地预览的方向, 在 `open ()` 之前调用。

1.5 将设备注册到 gb/t28181 服务器

接口原型: `public int open(int cameraId, SurfaceHolder surfaceHolder, String serverip, String localip, int serverport, String serverid, String userid, String userpwd)`

接口参数:

cameraId (in)	摄像机的 id, 0/1 - 后置/前置
surfaceHolder (in)	预览 surface 句柄
Serverip (in)	Gb/t 28181 服务器 IP 地址
Localip (in)	本地 IP 地址 (有些设备会存在多网卡现象, 故需指定本地 IP)
Serverport (in)	Gb/t28181 服务器端口
Serverid (in)	Gb/t 28181 服务 ID
Userid (in)	设备 ID
Userpwd (in)	设备密码
返回参数	0 表示成功;-1 表示 camera 打开失败;-2 表示注册失败

接口说明: 该接口用来将设备注册到 Gb/t 28181 服务器。

1.6 将设备从 gb/t28181 服务器上下线 (反注册)

接口原型: `public void close()`

接口参数: 无

接口说明: 该接口功能是将设备从 gb/t28181 服务器上下线, 与 `open ()` 相对。

1.7 设备开启音视频编码与转发

接口原型: `public boolean start()`

接口参数:

返回参数	False/true -- 失败/成功
------	---------------------

接口说明: 启动音视频编码与转发, 在 `open ()` 成功后调用。

1.8 设备停止音视频编码与转发

接口原型: `public void stop()`

接口参数: 无

接口说明: 停止音视频编码与转发, 与 `start()` 相对应。

1.9 发送（更新）设备的位置

接口原型: `public void setPosition(double longitude, double latitude)`

接口参数:

<code>longitude (in)</code>	设备的精度
<code>latitude (in)</code>	设备的维度

接口说明: 向 `gb/t28181` 服务器发送自己的位置, 注册成功后该接口可调用。

1.10 上报报警信息

接口原型: `public void sendOutAlarm(int alarmlevel, int alarmMethod, int alarmType, String salarmDesc)`

<code>alarmlevel (in)</code>	报警级别, 取值范围 1-4, 分别表示 1-4 级警情
<code>alarmMethod (in)</code>	报警方式, 取值范围 1-7, 详见本节后面
<code>alarmType (in)</code>	报警类型, 详见本节后面
<code>salarmDesc (in)</code>	报警的文字描述信息

接口说明: 向 `gb/t28181` 服务器上报警信息, 注册成功后该接口可调用。

报警方式的取值对应的含义如下:

取值	含义
1	电话报警
2	设备报警
3	短信报警
4	Gps 报警
5	视频报警
6	设备故障报警
7	其他报警

当报警方式 `alarmMethod` 为设备报警 (值为 2) 时, 报警类型的取值对应的含义如下:

取值	含义
1	视频丢失报警
2	设备防拆报警
3	存储设备磁盘满报警

4	设备高温报警
5	设备低温报警

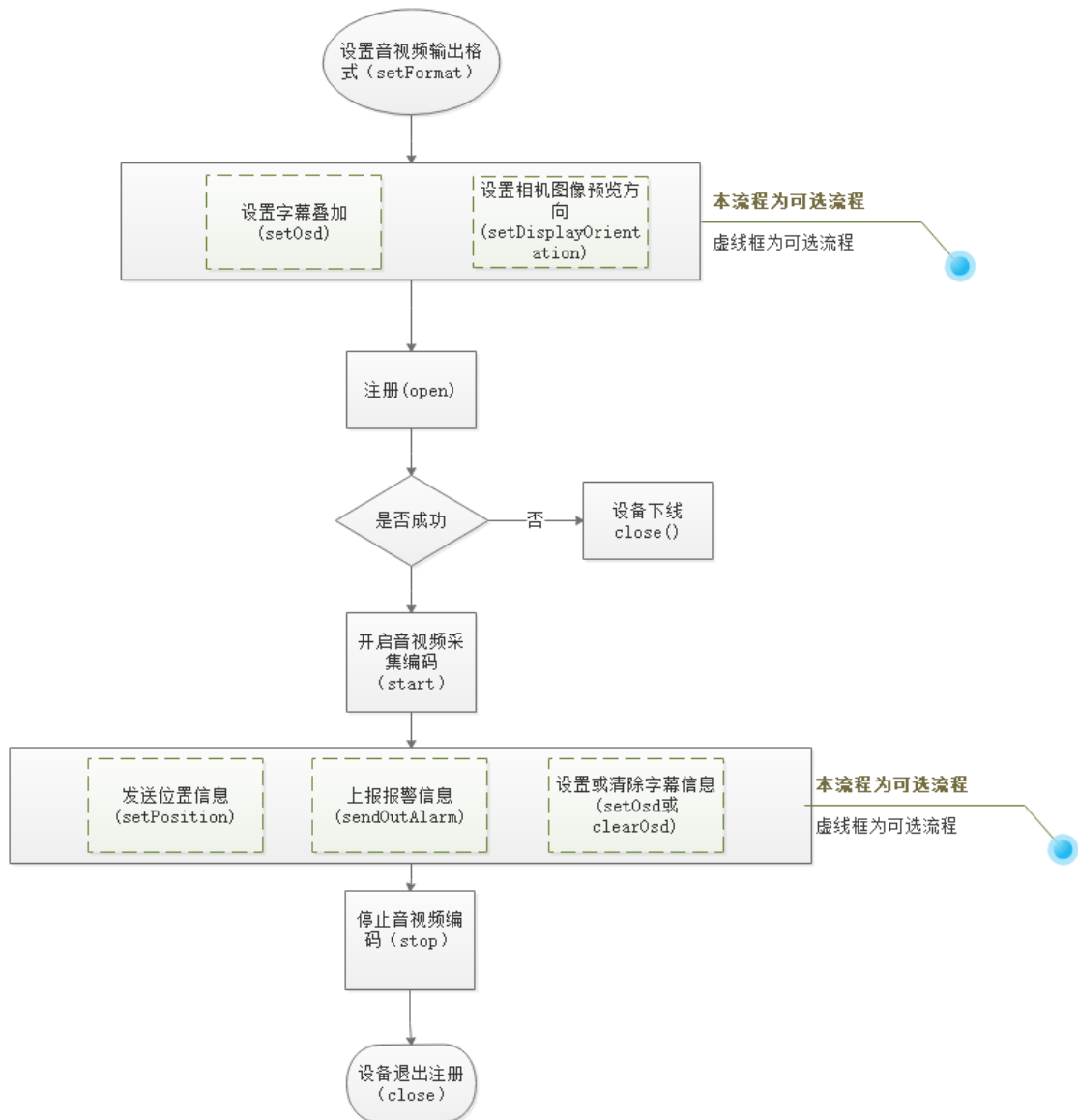
当报警方式 alarmMethod 为视频报警（值为 5）时，报警类型的取值对应的含义如下：

取值	含义
1	人工视频报警
2	运动目标检测报警
3	遗留物检测报警
4	物体移除检测报警
5	拌线检测报警
6	入侵检测报警
7	逆行检测报警
8	徘徊检测报警
9	流量统计报警
10	密度检测报警
11	视频异常检测报警
12	快速移动报警

当报警方式 alarmMethod 为设备故障报警（值为 6）时，报警类型的取值对应的含义如下：

取值	含义
1	存储设备磁盘故障报警
2	存储设备风扇故障报警

3 接口调用流程



4 代码示例

详见 demo 代码示例。