

SDK简介

- SDK的主要功能包括：监控点预览、回放功能。
- 开发者拿到的SDK压缩文件，解压后包括《移动端视频SDK编程指南》文档一份、HikVideoPlayerDemo 示例Demo一份。Demo仅作为示例，旨在直观的展示SDK的功能，界面较为简单，界面的实现不在本文档解释，请开发者参考demo中的示例代码及注释开发自己的界面和程序。请开发者认真阅读本文档，以便于顺利进行SDK的接入和开发工作。
- 环境要求：**AndroidStudio 3.0.0**版本及以上，**Android 4.3**及以上

更新说明

Version 1.3.0(build20200106)

- 支持软硬解码、智能信息展示设置。
- 增加电子放大功能支持。
- 增加arm64-v8a 64位库支持。

Version 1.2.1(build20191128)

- 修复播放视频概率性卡死问题。
- 修复录像定位跳跃问题。

Version 1.2.0(build20190517)

- 修复语音对讲啸叫问题。

Version 1.1.0(build20181213)

- 新增语音对讲功能。

Version 1.0.0(build20180928)

- 首个版本提供监控点的预览、回放相关功能。
- 在预览、回放过程中，提供抓图、声音控制、录像等功能。

快速集成

创建工程

用户需要自行通过Android Studio创建新项目，并配置应用的名称、软件包名称和项目的位置。如果项目已创建，请忽略此步骤。

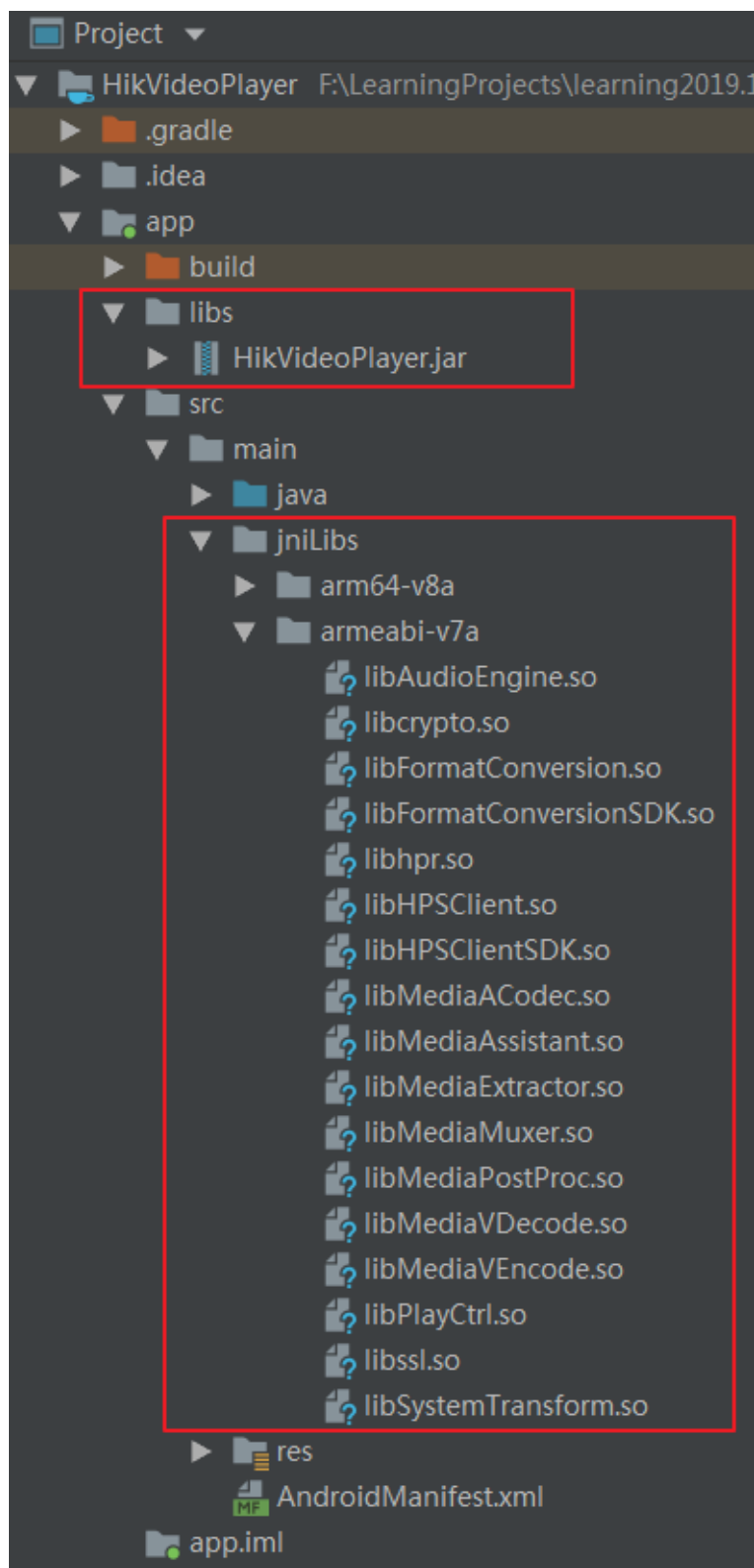
SDK导入

在 Android Studio 完成新项目的创建后，请从 IDE 左侧打开 **Project** 窗格并选择 **Project** 视图。
然后请严格按照以下步骤执行：

1. 首先将 **HikVideoPlayer.jar** 添加到 **Module** 的 **libs** 文件夹中，如果没有则先创建再添加。
2. 然后将SDK包括的全部so 库 添加到 **jniLibs** 文件夹中，如果没有则先创建再添加。

注意：由于**Android**官方已不再支持生成**ABI**为:**'armeabi'**的**so**库，因此海康威视不再提供**'armeabi'**的**so**库。如果您的项目中引用了其他三方**'armeabi'**的**so**库，可以将**'armeabi'**的**so**库直接置于**'armeabi-v7a'**中。如果您的项目中存在多种**ABI**文件夹，例如同时存在**'armeabi'**和**'armeabi-v7a'**文件夹，您必须保证项目中的各个**ABI**文件夹都有一份相应**ABI**格式的**so**库。建议您只需要保留**ABI**为**'armeabi-v7a'**的**so**即可。**arm64-v8a**同理。

如下图所示：



3. 接着在 Module 的 `build.gradle` 文件的 `dependencies` 中添加如下代码，并同步项目：

```
dependencies {  
    implementation fileTree(include: ['*.jar'], dir: 'libs')  
}
```

4. 最后打开 Module 的 `AndroidManifest.xml`，添加网络权限；如需使用SDK本地录像功能，需申请Android系统读写存储权限；如需使用SDK语音对讲功能，还需申请Android系统麦克风权限。示例代码

如下:

```
<uses-permission android:name="android.permission.INTERNET" />
<!-- 允许程序写入录像文件和抓图 -->
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<!-- 允许程序录制音频 -->
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
```

对于目标**Android API**在**23 (Android 6.0)** 以上的, 还需在**Java**代码中动态申请系统权限。
配置完项目后, 我们可以使用HikVideoPlayer了。

SDK初始化

初始化方法必须在工程的自定义 **Application** 类的 **onCreate()** 方法中调用。

```
HikVideoPlayerFactory.initLib(String appKey, boolean enableLog);
```

appKey 为保留字段, 目前只需传入null或空字符串即可。 **enableLog** 为日志开关选项, 传入 **true** 可以在 **debug** 模式下打开SDK的日志打印。

预览功能使用

```
//创建HikVideoPlayer实例
HikVideoPlayer mPlayer = HikVideoPlayerFactory.provideHikVideoPlayer();

//设置播放画面显示surface, 仅支持TextureView
TextureView textureView = findViewById(R.id.textureView);
mPlayer.setSurfaceTexture(textureView.getSurfaceTexture());

//开启预览
//realPlayUrl为预览短链接, 需要通过调用openApi获取
//此方法需要在子线程中执行
mPlayer.startRealPlay(realPlayUrl, new HikVideoPlayerCallback(){

    //播放结果回调
    //status 共三种状态: SUCCESS (播放成功)、FAILED (播放失败)、EXCEPTION (取流异常)
    //错误码: 只有 FAILED 和 EXCEPTION 才有值
    //注意: 回调仍为子线程
    @Override
    public void onPlayerStatus(@NonNull Status status, int errorCode) {

    }

});
```

回放功能使用

```

//创建HikVideoPlayer实例
HikVideoPlayer mPlayer = HikVideoPlayerFactory.provideHikVideoPlayer();

//设置播放画面显示surface,仅支持TextureView
TextureView textureView = findViewById(R.id.textureView);
mPlayer.setSurfaceTexture(textureView.getSurfaceTexture());

//开启回放
//playbackUrl为回放短链接,需要通过调用openApi获取
//回放方法需要传入播放的录像片段的开始时间和结束时间
//时间格式为 "yyyy-MM-dd'T'HH:mm:ss.SSSZ",例如: "2018-05-07T14:41:57.819+03:00"
//此方法需要在子线程中执行
mPlayer.startPlayback(playbackUrl,startTime, stopTime, new HikVideoPlayerCallback(){

    //播放结果回调
    //status 共四种状态: SUCCESS (播放成功)、FAILED (播放失败)、EXCEPTION (取流异常)、FINISH (回放结束)
    //错误码,只有 FAILED 和 EXCEPTION 才有值
    //注意:回调仍为子线程
    @Override
    public void onPlayerStatus(@NonNull Status status, int errorCode) {

    }

});

```

语音对讲功能使用

```

//创建HikVideoPlayer实例
HikVideoPlayer mPlayer = HikVideoPlayerFactory.provideHikVideoPlayer();

//开启对讲
//talkUrl为对讲短链接,需要通过调用openApi获取
//此方法需要在子线程中执行
mPlayer.startVoiceTalk(talkUrl, new VoiceTalkCallback(){

    //播放结果回调
    //status 共三种状态: SUCCESS (开启对讲成功)、FAILED (开启对讲失败)、EXCEPTION (取流异常)
    //错误码,只有 FAILED 和 EXCEPTION 才有值
    //注意:回调仍为子线程
    @Override
    public void onTalkStatus(@NonNull HikVideoPlayerCallback.Status status, int errorCode) {

    }

});

```

混淆配置

如果App需要使用proguard进行混淆打包,请添加以下混淆代码:

```
-dontwarn android.support.annotation.Keep
# 保留类内部使用@keep注解的成员变量
-keep @android.support.annotation.Keep class **{
    @android.support.annotation.Keep <fields>;
}

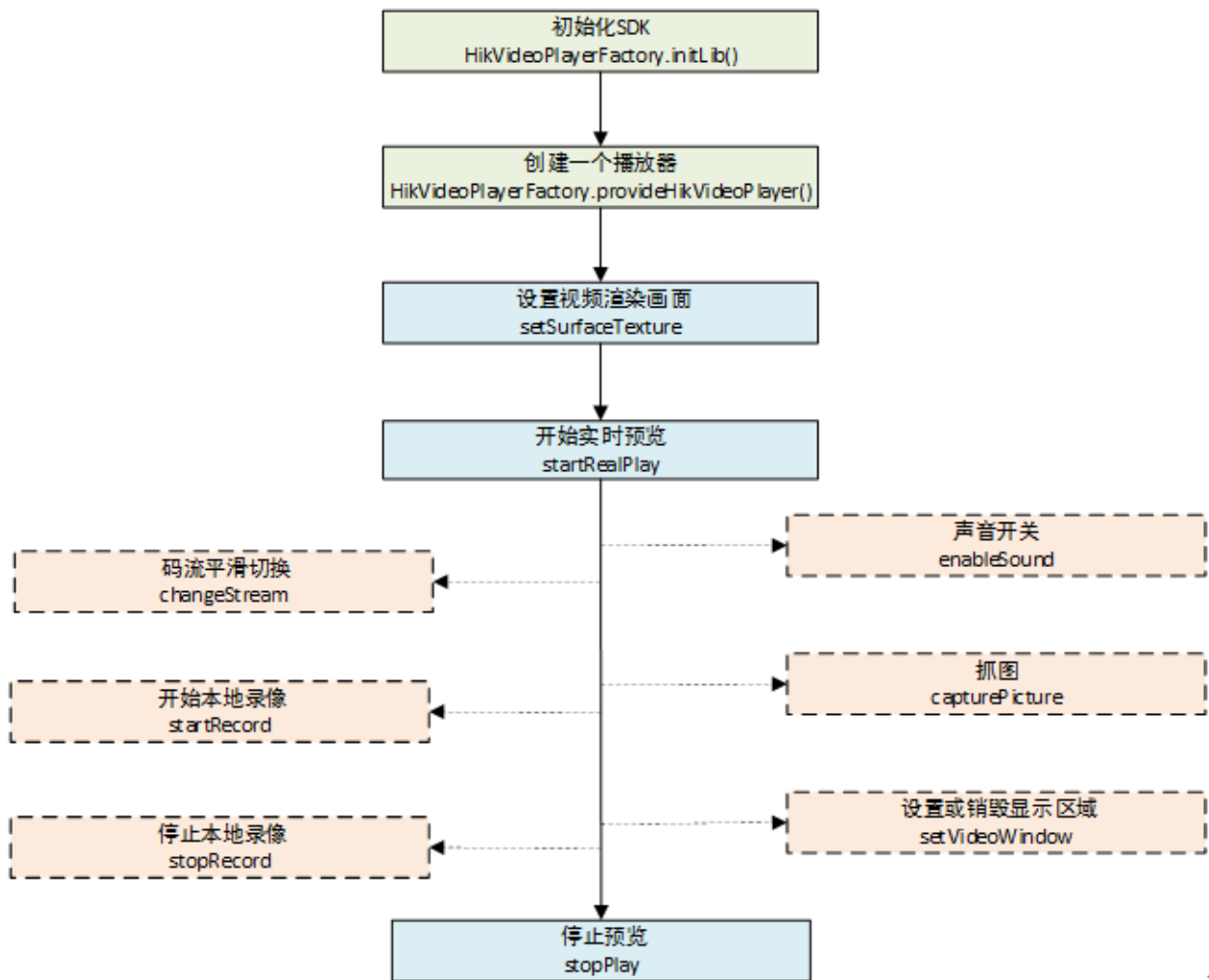
# natvie 方法不混淆
-keepclasseswithmembernames class * {
    native <methods>;
}

# 海康威视视频取流播放相关库的混淆配置
-keep class org.MediaPlayer.PlayM4.** {*; }
-keep class com.hikvision.netsdk.** {*; }
-keep class com.hikvision.audio.** {*; }
-keep class hik.common.isms.hpsclient.** {*; }
-keep class com.hikvision.open.hikvideoplayer.** {*; }
```

编程引导

实时预览

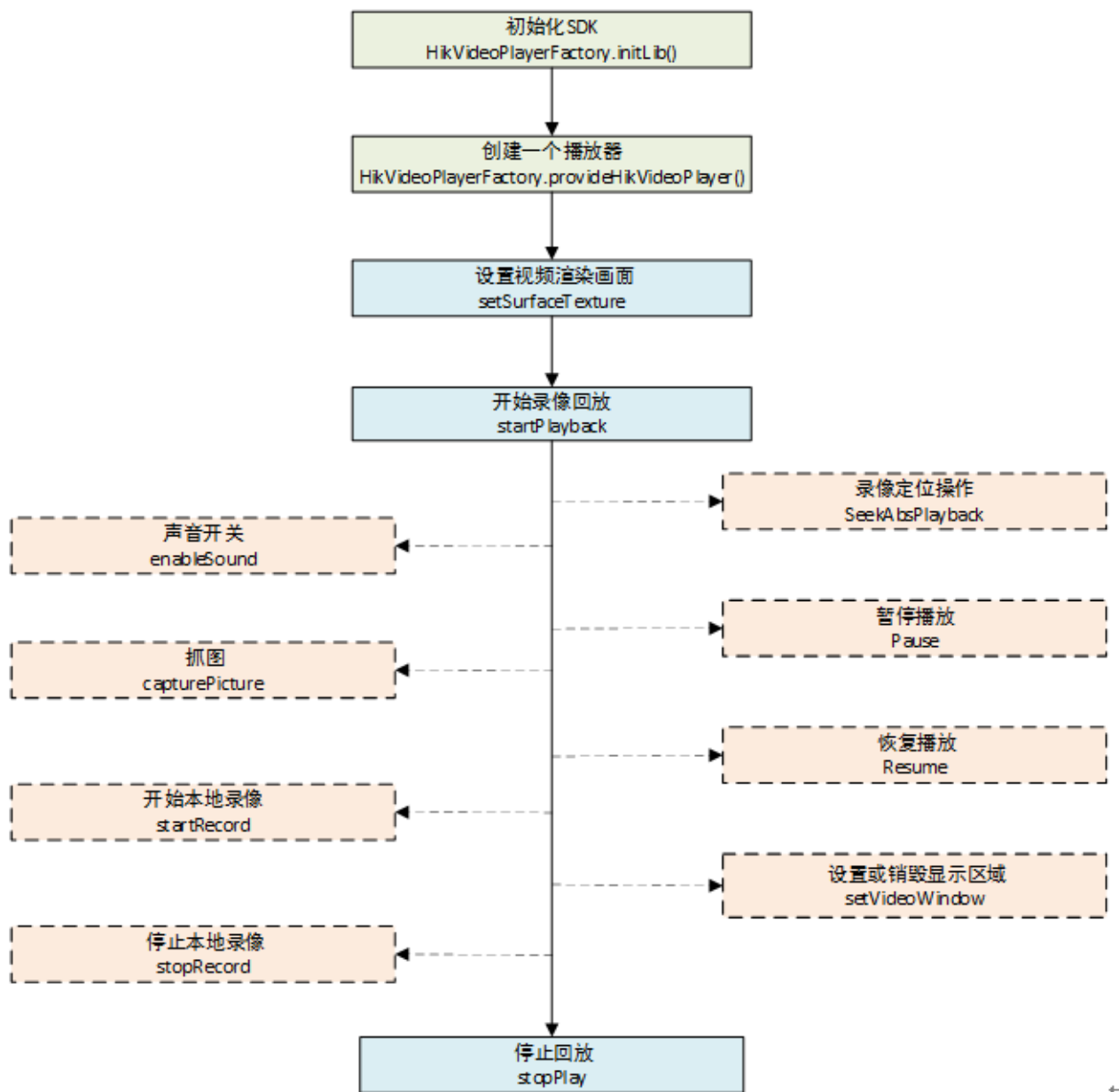
实时预览功能可以实时查看摄像机的监控画面，从而进行实时监控。本节对实时预览的接入流程进行详细说明。接口调用流程如下：



图中虚线框部分的模块是与预览模块相关，这些模块之间是并列的关系，各自完成相应的功能。

录像回放

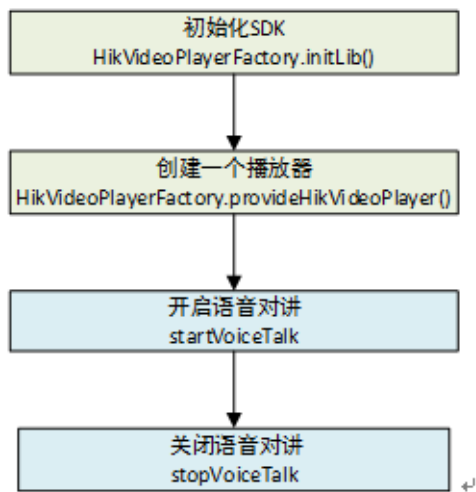
该流程介绍录像回放功能接入，接口调用流程如下：



图中虚线框部分的模块是与回放模块相关，这些模块之间是并列的关系，各自完成相应的功能。

语音对讲

该流程介绍语音对讲功能接入，接口调用流程如下：



函数说明

1.初始化

1.1 SDK初始化

接口名称：

```
public static boolean initLib(@Nullable String appKey, boolean enableLog)
```

接口描述：

SDK初始化，调SDK其它接口的前提。

参数说明：

成员名	数据类型	必填	说明
appKey	String	否	保留参数，直接传null即可
enableLog	boolean	是	日志开关控制，ture-打开 fasle-关闭

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

1. 只需在APP启动时调用一次即可；
2. 建议在debug模式下打开日志，release关闭日志打印；

1.2 获取视频播放器

接口名称：

```
public static HikVideoPlayer provideHikVideoPlayer()
```

接口描述：

用来获取一个海康播放器实例，可以用于实时预览播放或录像回放播放。

参数说明：

无

返回值：

海康视频播放器实例

备注：

每一路视频须获取一个HikVideoPlayer。

1.3 设置视频显示窗口

接口名称：

```
void setSurfaceTexture(@NonNull SurfaceTexture surfaceTexture);
```

接口描述：

设置视频显示画面。

参数说明：

成员名	数据类型	必填	说明
surfaceTexture	SurfaceTexture	是	视频流渲染画面， 通过TextureView.getSurfaceTexture获得。

返回值：

无

备注：

目前只支持TextureView，不支持SurfaceView。

关联接口：

无

2.视频预览

2.1 开始预览

接口名称：

```
boolean startRealPlay(@NonNull String liveRtspUrl, @NonNull HikVideoPlayerCallback callback);
```

接口描述：

开启视频预览。

参数说明：

成员名	数据类型	必填	说明
liveRtspUrl	String	是	预览取流URL，不能为空
callback	HikVideoPlayerCallback	是	播放结果回调函数,详见 HikVideoPlayerCallback

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

使用场景：

主子码流切换、视频实时画面监控。

备注：

- 1. 有些第三方设备可能不支持子码流，只能以主码流形式预览，请求预览URL时注意取主码流的URL。
- 2. 对该接口功能，开放平台OpenAPI 自V1.1版本支持在获取监控点取流URL的请求体 "expand" 字段中指定扩展字段为: "expand": " transcode=1&videotype=h264" 。否则SDK无法解码大华协议的码流，导致画面无法显示。如果是非大华设备，即在获取监控点取流URL的请求体"expand"字段中必须指定封装格式为: "expand": "streamform=ps" 。
- 3. 调用开始预览接口若返回false代表开启预览失败，但是预览成功需要根据HikVideoPlayerCallback的回调结果确定，详情请查看Demo。
- 4. 预览中可能因网络波动导致取流发生异常，需要在 异常回调中主动重新开启或者停止播放 。

关联接口：

详见 `stopPlay` 。

2.2 码流平滑切换

接口名称：

```
boolean changeStream(@NonNull String liveRtspUrl, @NonNull HikVideoPlayerCallback liveCallback);
```

接口描述：

切换码流清晰度接口，用于码流无缝切换高清-标清，不会出现黑屏。

参数说明：

成员名	数据类型	必填	说明
liveRtspUrl	String	是	要切换的码流URL，不能为空
callback	HikVideoPlayerCallback	是	播放结果回调函数,详见 HikVideoPlayerCallback

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

使用场景：

主子码流平滑无缝切换，提升用户体验。

备注：

- 1.使用码流平滑切换接口之前，必须先调用 开始预览 接口；
- 2.使用方式与 开始预览 接口一致；

关联接口：

详见 开始预览、stopPlay。

3.网络录像回放

3.1 开始回放

接口名称：

```
boolean startPlayback(@NonNull String playbackUrl, @NonNull String start, @NonNull String end, @NonN
```

接口描述：

开始回放。

参数说明：

成员名	数据类型	必填	说明
playbackUrl	String	是	回放URL，不能为空
start	String	是	回放录像文件的开始时间，时间格式为'yyyy-MM-dd'T'HH:mm:ss.SSSZ'
end	String	是	回放录像文件的结束时间，时间格式为'yyyy-MM-dd'T'HH:mm:ss.SSSZ'
callback	HikVideoPlayerCallback	是	播放结果回调函数,详见 HikVideoPlayerCallback

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

1. 在开始回放之前，需要先从录像存储服务器中查询录像片段列表，录像回放的开始时间是录像片段列表中第一个片段的开始时间，结束时间是录像片段列表中最后一个片段的结束时间。
2. 对该接口功能，开放平台OpenAPI 自V1.1版本支持在获取监控点取流URL的请求体 "expand" 字段中指定扩展字段为: "expand": " transcode=1&videotype=h264" 。否则SDK无法解码大华协议的码流，导致画面无法显示。如果是非大华设备，即在获取监控点取流URL的请求体"expand"字段中必须指定封装格式为: "expand": "streamform=ps" 。
3. 调用开始回放接口若返回false代表开启回放失败，但是回放成功需要根据HikVideoPlayerCallback的回调结果确定，详情请查看Demo。
4. 回放中可能因网络波动导致取流发生异常，需要在 异常回调中主动重新开启或者停止播放 。
5. 开始时间和结束时间为 **AbsTime** 的 **startPlayback()** 方法已不推荐使用。

关联接口：

详见 `stopPlay` 。

3.2 按绝对时间回放定位

接口名称：

```
boolean seekAbsPlayback(@NonNull String seekTime, @NonNull HikVideoPlayerCallback callback);
```

接口描述：

在录像回放的开始时间和结束时间之间进行播放定位控制。

参数说明：

成员名	数据类型	必填	说明
seekTime	String	是	定位录像文件的开始时间，时间格式为'yyyy-MM-dd'T'HH:mm:ss.SSSZ'
callback	HikVideoPlayerCallback	是	播放结果回调函数,详见 HikVideoPlayerCallback

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

- 1. 使用按绝对时间回放定位接口之前，必须先调用开始回放接口；
- 2. 定位播放时，时间戳单位为秒。
- 3. 调用录像回放定位操作返回false代表录像定位失败，但是定位成功需要根据HikVideoPlayerCallback的回调结果确定，详情请查看Demo。
- 4. 定位时间为 **AbsTime** 的 `seekAbsPlayback()` 方法已不推荐使用。

关联接口：

详见 `startPlayback`。

3.3 查询当前播放时间戳接口

接口名称：

```
long getOSDTime();
```

接口描述：

查询当前播放时间戳。

参数说明：

无

返回值：

-1表示失败，大于等于0时表示当前回放的时间戳，单位：秒。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

无。

关联接口：

无。

3.4 暂停回放

接口名称：

```
boolean pause();
```

接口描述：

暂停录像回放取流及解码播放。

参数说明：

无。

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

无。

关联接口：

详见 `resume`。

3.5 恢复回放

接口名称：

```
boolean resume();
```

接口描述：

恢复录像回放取流及解码播放。

参数说明：

无。

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

无。

关联接口：

详见 `pause`。

4. 停止播放

4.1 停止播放

接口名称：

```
boolean stopPlay();
```

接口描述：

停止实时预览或录像回放。

参数说明：

无。

返回值：

`true`表示成功，`false`表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

停止播放失败不影响下次开启。

关联接口：

详见 `startRealPlay`、`startPlayback`。

5. 语音对讲

5.1 开启语音对讲

接口名称：

```
boolean startVoiceTalk(@NonNull String talkUrl, HikVideoPlayerCallback.VoiceTalkCallback voiceTalkCa
```

接口描述：

开启语音对讲。

参数说明：

成员名	数据类型	必填	说明
talkUrl	String	是	语音对讲取流URL，不能为空
voiceTalkCallback	HikVideoPlayerCallback.VoiceTalkCallback	是	开启对讲结果回调函数，详见 HikVideoPlayerCallback

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

使用场景：

移动端与设备端进行实时语音对讲。

备注：

- 1. 调用开始预览接口若返回false代表开启对讲失败，但是对讲成功需要根据 HikVideoPlayerCallback.VoiceTalkCallback的回调结果确定，详情请查看Demo。
- 2. 对讲过程中可能因网络波动或对讲通道被占用导致发生异常，需要在异常回调中主动关闭语音对讲。

关联接口：

详见 `stopVoiceTalk`。

5.2 关闭语音对讲

接口名称：

```
boolean stopVoiceTalk();
```

接口描述：

关闭语音对讲。

参数说明：

无。

返回值：

true表示成功，false表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

关闭失败不影响下次开启。

关联接口：

详见 `startVoiceTalk`。

6.播放抓图

6.1 预览回放抓图

接口名称：

```
boolean capturePicture(@NonNull String bitmapPath);
```

接口描述：

在视频预览或回放时，从视频中抓取一帧，并保存至指定的本地文件路径。

参数说明：

成员名	数据类型	必填	说明
bitmapPath	String	是	要保存的图片绝对路径（含文件名），根据文件名后缀确定保存的图片格式（支持jpg），例：“/SnapShot/测试监控点.jpg”。

返回值：

`true`表示成功，`false`表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

本地抓图文件可以通过手机自带图片应用查看。

关联接口：

无。

7.本地录像

7.1 开启本地录像

接口名称：

```
boolean startRecord(@NonNull String mediaFilePath);
```

接口描述：

在视频预览或回放时，保存录像数据流，并保存至指定的本地文件路径。

参数说明：

成员名	数据类型	必填	说明
mediaFilePath	String	是	要保存的录像文件绝对路径（含文件名），例：“/SnapShot/测试监控点.mp4”。

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

- 1. 因缓存原因录像会有 2~3 秒延时。
- 2. 本地录像文件可以通过手机自带视频播放器查看。

关联接口：

详见 `stopRecord`。

7.2 关闭本地录像

接口名称：

```
boolean stopRecord();
```

接口描述：

在视频预览或回放时，关闭录像流。

参数说明：

无

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

- 1. 因缓存原因录像会有 2~3 秒延时。
- 2. 本地录像文件可以通过手机自带视频播放器查看。

关联接口：

详见 startRecord 。

8.声音控制

8.1 声音控制

接口名称：

```
boolean enableSound(boolean enable);
```

接口描述：

打开预览或回放中的声音，只管一路预览或回放。

参数说明：

成员名	数据类型	必填	说明
enable	boolean	是	打开预览或回放中的声音，只管一路预览或回放。

返回值：

true表示成功，false表示失败。接口返回失败请调用 getLastError 获取错误码，通过错误码判断出错原因。

备注：

- 1. 打开声音，同一时刻只能有一路声音。如果现在已经有声音打开，则自动关闭原来已经打开的声音。
- 2. 默认情况下是不开启声音的。

关联接口：

无。

9.电子放大

9.1 开启电子放大

接口名称：

```
boolean openDigitalZoom(@NonNull CustomRect original, @NonNull CustomRect target)
```

接口描述：

执行电子放大功能。

参数说明：

成员名	数据类型	必填	说明
original	CustomRect	是	最初的显示区域的坐标矩形
target	CustomRect	是	目标显示区域的坐标矩形

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

- 1. 电子放大支持倍率为**1-10**倍。
- 2. 默认情况下不开启电子放大的。

关联接口：

详见 `closeDigitalZoom`

9.2 关闭电子放大

接口名称：

```
boolean closeDigitalZoom();
```

接口描述：

关闭电子放大功能。

参数说明：

无。

返回值：

true表示成功，**false**表示失败。接口返回失败请调用 `getLastError` 获取错误码，通过错误码判断出错原因。

备注：

- 1. 电子放大支持倍率为**1-10**倍。
- 2. 默认情况下是不开启电子放大的。

关联接口：

详见 `openDigitalZoom`

10.播放配置

10.1 解码设置

接口名称：

```
void setHardDecodePlay(boolean isHardDecode);
```

接口描述：

开启预览或回放前，设置码流解码是否使用硬件解码；

参数说明：

成员名	数据类型	必填	说明
isHardDecode	boolean	是	开启或关闭硬件解码，只管一路预览或回放。

返回值：

无。

备注：

- 1. 开启硬件解码后，智能信息不会显示。
- 2. 此方法必须在预览或回放开启前设置才会应用到本次播放，否则本次播放不起作用。
- 3. 默认使用软件解码；

关联接口：

无。

10.2 智能信息显示设置

接口名称：

```
void setSmartDetect(boolean isShow);
```

接口描述：

开启预览或回放前，设置智能信息是否展示；

参数说明：

成员名	数据类型	必填	说明
-----	------	----	----

成员名	数据类型	必填	说明
isShow	boolean	是	开启或关闭智能信息显示，只管一路预览或回放。

返回值：

无。

备注：

- 1. 开启硬件解码后，智能信息不会显示。
- 2. 此方法必须在预览或回放开启前设置才会应用到本次播放，否则本次播放不起作用。
- 3. 默认不展示智能信息。

关联接口：

无。

11.错误码

11.1 获取错误码

接口名称：

```
int getLastError();
```

接口描述：

接口调用失败时，调用此函数获取错误码；

参数说明：

无

返回值：

错误码。

备注：

此接口返回的错误码为10进制int值，需要转换成16进制错误码，才能对照 取流库错误码 中的取流库错误码查看出错原因。

关联接口：

无。

12.获取版本号

12.1 获取版本号

接口名称：

```
String getVersion();
```

接口描述：

获取版本号。

参数说明：

无

返回值：

成功表示版本号，如“V1.1.0”，null表示失败。

备注：

无。

关联接口：

无。

回调函数定义

1. 播放状态回调

回调接口：

```
public interface HikVideoPlayerCallback {
    @WorkerThread
    void onPlayerStatus(@NonNull Status status, int errorCode);
}
```

回调接口描述：

用于开始实时预览或开始录像回放，注册监听播放器状态回调。

参数说明：

成员名	数据类型	必填	说明
-----	------	----	----

成员名	数据类型	必填	说明
status	Status	是	Status为枚举，表示播放器状态。有四种类型：SUCCESS：解码播放成功，视频画面显示；FAILED：开启实时预览或录像回放失败，取流失败；EXCEPTION：播放过程中发生异常，请从新取流；FINISH：录像回放取流结束，录像播完了。
errorCode	int	否	错误码，只有在 status 状态为： FAILED 、 EXCEPTION 才有值 ,其他 status 值为 -1。

备注：

无。

关联接口：

详见 startRealPlay 、 startPlayback

错误码及说明

取流库错误码说明

|0x017是媒体网关或媒体取流SDK的错误，0x018是视频点播服务的错误码，0x019是设备接入框架的错误码。

下表是部分媒体取流客户端的错误码：

错误代码数值	宏定义	说明
0x0173ea60	HPS_PUBLIC__SOCKET_CREATE_FAILED	创建socket失败
0x0173ea61	HPS_PUBLIC__SOCKET_SETREUSEADDR_FAILED	设置socket地址重用失败
0x0173ea62	HPS_PUBLIC__SOCKET_MAKEADDR_FAILED	生成socket地址结构失败
0x0173ea63	HPS_PUBLIC__SOCKET_SETBUFFERSIZE_FAILED	设置socket缓冲区失败
0x0173ea64	HPS_PUBLIC__SOCKET_BIND_FAILED	绑定socket端口失败
0x0173ea65	HPS_PUBLIC__SOCKET_LISTEN_FAILED	监听socket失败
0x0173ea66	HPS_PUBLIC__SOCKET_CONNECT_FAILED	连接socket失败
0x0173ea67	HPS_PUBLIC__SOCKET_HANDLE_INVALID	socket句柄无效
0x0173ea68	HPS_PUBLIC__ASYNCIO_BINDIOQUENE_FAILED	绑定io完成端口队列失败

错误代码数值	宏定义	说明
0x0173ea69	HPS_PUBLIC__ASYNCIO_SENDDATA_FAILED	iocp发送数据失败
0x0173ea6a	HPS_PUBLIC__ASYNCIO_RECVDATA_FAILED	iocp接收数据失败
0x0173ea6b	HPS_PUBLIC__ASYNCIO_POSTIOSTATUS_FAILED	投递iocp完成状态失败
0x0173ea6c	HPS_PUBLIC__ASYNCIO_ACCEPT_FAILED	iocp接收连接失败
0x0173ea6d	HPS_PUBLIC__ASYNCIO_BINDIOHANDLE_FAILED	绑定io完成端口句柄失败
0x0173ea6e	HPS_PUBLIC__MELLOC_MEM_FAILED	内存申请失败
0x0173ea6f	HPS_PUBLIC__FUNC_PARAMS_INVALID	函数参数无效
0x0173ea70	HPS_PUBLIC__FUNC_NOT_SUPPORT	功能不支持或未实现
0x0173ea71	HPS_PUBLIC__IDENTIFY_TOKEN_INVALID	身份认证token无效
0x0173ea72	HPS_PUBLIC__SESSIONHANDLE_INVALID	会话handle无效
0x0173ea73	HPS_PUBLIC__URL_FORMAT_INVALID	url格式错误
0x0173ea74	HPS_PUBLIC__LENGTH_OUTLIMIT	数据长度超出限制范围
0x0173ea75	HPS_PUBLIC__RTSP_RSP_ERROR	rtsp协议报文异常
0x0173ea76	HPS_PUBLIC__TRANSMETHOD_INVALID	传输方式无效或不支持
0x0173ea77	HPS_PUBLIC__HTRACE_FUNC_FAIL	埋点库调用异常
0x0173ea78	HPS_TRANS__REPEAT_OPEN_FAILED	重复打开传输连接
0x0173ea79	HPS_TRANS__SOCKET_SETMULTITTL_FAILED	设置socket多播ttl失败
0x0173ea7b	HPS_RTSP__RSAKEY_INIT_FAILED	rsa公钥初始化失败
0x0173ea7c	HPS_RTSP__RSA_ENCRYPT_FAILED	rsa公钥加密失败
0x0173ea7d	HPS_RTSP__AES_ENCRYPT_FAILED	aes加密失败
0x0173ea7e	HPS_RTSP__BASIC_ENCODE_FAILED	base64编码失败
0x0173ea7f	HPS_RTSP__GET_RAND_NUM_FAIL	获取随机数失败
0x0173ea80	HPS_RTSP__ASYNC_CB_EXCEPTION	异步消息回调异常
0x0173ea81	HPS_RTSP__SESSION_STATE_INVALID	rtsp会话状态无效
0x0173ea82	HPS_RTSP__ASYNCINFO_INVALID	rtsp异步会话信息无效

错误代码数值	宏定义	说明
0x0173ea83	HPS_RTSP__CONFIG_SESSION_INVALID	会话配置信息无效
0x0173ea85	HPS_RTSP__IPCONVERT_FAILED	ip/域名转换ip失败
0x0173ea86	HPS_RTSP__SEND_DESCRIBE_ERROR	发送describe失败
0x0173ea87	HPS_RTSP__RECV_DESCRIBE_TIMEOUT	接收describe响应超时
0x0173ea88	HPS_RTSP__SEND_SETUP_ERROR	发送setup失败
0x0173ea89	HPS_RTSP__RECV_SETUP_TIMEOUT	接收setup响应超时
0x0173ea8a	HPS_RTSP__SEND_PLAY_ERROR	发送play失败
0x0173ea8b	HPS_RTSP__RECV_PLAY_TIMEOUT	接收play响应超时
0x0173ea8c	HPS_RTSP__SEND_TREARDOWN_ERROR	发送teardown失败
0x0173ea8d	HPS_RTSP__RECV_TREARDWN_TIMEOUT	接收teardown响应超时
0x0173ea8e	HPS_RTSP__SEND_OPTIONS_ERROR	发送options失败
0x0173ea8f	HPS_RTSP__RECV_OPTIONS_TIMEOUT	接收options响应超时
0x0173ea90	HPS_RTSP__SEND_PAUSE_ERROR	发送pause失败
0x0173ea91	HPS_RTSP__RECV_PAUSE_TIMEOUT	接收pause响应超时
0x0173ea92	HPS_RTSP__SEND_FORCEIFRAME_ERROR	发送forceiframe失败
0x0173ea93	HPS_RTSP__RECV_FORCEIFRAME_TIMEOUT	接收forceiframe响应超时
0x0173ea94	HPS_RTSP__SEND_SETPARAMETER_ERROR	发送setparameter失败
0x0173ea95	HPS_RTSP__RECV_SETPARAMETER_TIMEOUT	接收setparameter响应超时
0x0173ea96	HPS_RTSP__ASYNC_RECV_TIMEOUT	异步接收超时
0x0173ea97	HPS_RTSP__RECV_NOT_FULL	数据接收不完整
0x0173ea98	HPS_RTSP__PARSE_RTSP_FAILED	解析rtsp报文失败
0x0173ea99	HPS_RTSP__HEARTBEAT_TIMEOUT	心跳超时(client与server间)
0x0173ea9a	HPS_RTSP__PROC_RECVDATA_EXCEPTION	处理接收到的数据异常
0x0173ea9b	HPS_RTSP__GET_SERVER_UDPPORT_FAIL	获取server端udp端口失败
0x0173ea9c	HPS_RTSP__CREATE_UDPTRANS_FAIL	创建udp传输失败

错误代码数值	宏定义	说明
0x0173ea9d	HPS_RTSP__CREATE_TCPTRANS_FAIL	创建tcp传输失败
0x0173ea9e	HPS_RTSP__OPEN_UDPTRANS_FAIL	开启udp传输失败
0x0173ea9f	HPS_RTSP__OPEN_TCPTRANS_FAIL	开启tcp传输失败
0x0173eaa0	HPS_RTSP__SOCKET_SETOPT_FAIL	socket设置失败
0x0173eaa1	HPS_RTSP__SERVER_NOT_VTM	请求端不是集群调度节点
0x0173eaa2	HPS_RTSP__THREAD_HANDLE_INVALID	线程句柄无效
0x0173eaa3	HPS_RTSP__NO_RTSP_SESSION	无可会话句柄
0x0173eaa4	HPS_RTSP__HANDLE_ALREADY_IN_QUENE	会话句柄已经在队列中
0x0173eaa5	HPS_RTSP__CREATE_ASYNC_QUENE_FAIL	创建异步io队列失败
0x0173eaa6	HPS_STREAM__CB_BLOCK	回调线程出现阻塞
0x0173eaa7	HPS_STREAM__SYSTRANS_FAIL	转封装库接口调用失败
0x0173eaa8	HPS_STREAM__GET_EXEPATH_FAIL	获取/ 设置当前程序运行路径失败
0x0173eaa9	HPS_STREAM__FILE_OPEN_FAIL	文件打开失败
0x0173eaaa	HPS_STREAM__JSON_PARSE_FAIL	json解析失败
0x0173eaab	HPS_STREAM__PARSE_SDP_FAIL	sdp解析失败
0x0173eaac	HPS_STREAM__NOT_INIT	sdk未初始化
0x0173eaad	HPS_STREAM__RTSP_CLIENT_INIT_FAIL	rtsp协议栈未初始化
0x0173eaae	HPS_STREAM__MEDIACOUNT_LESS_ZERO	sdp媒体信息少于等于0
0x0173eaaf	HPS_STREAM__ABSTIME_ERROR	绝对时间转换失败
0x0173eab0	HPS_STREAM__BUFFER_TOO_SHORT	buffer长度不足
0x0173eab1	HPS_STREAM__TRY_TIMES_FAIL	多次尝试取流后依旧失败

常见问题排查

1.SDK初始化的appKey从哪获取？

目前V1.0和V1.1版本的SDK无需appKey，直接填null即可。

2.按照文档将so库拷贝到项目中，运行后崩溃，提示某些so库找不到？

```
java.lang.UnsatisfiedLinkError: dlopen failed: library "/system/lib64/libcrypto.so" needed or dlopen
HikVideoPlayerFactory.initLib(null, true);
```

由于Android官方已不再支持生成ABI为:'armeabi'的so库，因此海康威视不再提供'armeabi'的so库。如果您的项目中引用了其他三方'armeabi'的so库，可以将'armeabi'的so库直接置于'armeabi-v7a'中。如果您的项目中存在多种ABI文件夹，例如同时存在'armeabi'和'armeabi-v7a'文件夹，您必须保证项目中的各个ABI文件夹都有一份相应ABI格式的so库。建议您只需要保留ABI为'armeabi-v7a'的so即可。目前的Android应用已无需支持'armeabi'格式的ABI。

有关海康SDK或三方SDK中的so库导入请查看 [SDK导入](#) 小节。

3.开启实时预览或录像回放后，始终不显示画面？

首先请确认要查看的监控设备是否是大华设备，开放平台OpenAPI 自V1.1版本支持在获取监控点取流URL的请求体"expand"字段中指定扩展字段为: "expand": " transcode=1&videotype=h264" 。否则SDK无法解码大华协议的码流，导致画面无法显示。

如果是非大华设备，即在获取监控点取流URL的请求体"expand"字段中必须指定封装格式为:

"expand": "streamform=ps" 。

4.开启语音对讲失败？

开放平台OpenAPI目前仅支持海康、国标、Ehome协议接入的监控设备语音对讲，大华和onvif协议接入的监控设备不支持语音对讲。

5.进行本地录像后发现没有录像文件生成？

使用移动端SDK在预览或回放时进行本地录像，如果是非大华设备，必须在获取时URL中指定PS封装格式，即在获取监控点取流URL的请求体"expand"字段中指定封装格式为: "expand": "streamform=ps" 。否则会出现录像无法生成的问题。

对于大华设备，开放平台OpenAPI 自V1.1版本支持在获取监控点取流URL的请求体"expand"字段中指定扩展字段为: "expand": " transcode=1&videotype=h264" 。否则移动端SDK无法解码大华协议的码流，导致画面无法显示。