短视频 SDK

常见问题

产品文档

腾讯云

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

常见问题
- 上传常见问题
- Android 编译问题
- Android 常见问题
- iOS 常见问题
- License 常见问题

常见问题

上传常见问题

最近更新时间：2022-08-16 17:09:23

什么是客户端视频上传？

客户端视频上传，是指 App 用户将本地视频直接上传到云点播。

视频上传功能 TXUGCPublish 找不到？

视频上传模块已经从 SDK 中独立出来，并开源到 Demo 中，需要用户自己集成短视频上传，步骤如下：

1. 下载 Demo。

2. 将 `app\libs\upload` 目录下上传的 jar 包拷贝到您的项目 `..\app\libs\upload` 目录下。

3. 将短视频上传源码目录 `Demo\app\src\main\java\com\tencent\liteav\demo\videoupload` 拷贝到您自己的工程目录下，并修改源码里的 package 名称。

4. 在工程 App 目录下的 build.gradle 中，添加引用 jar 包的代码。

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

5. 在 `AndroidManifest.xml` 中配置 App 的权限。

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
```

上传失败，报1000的内部错误？

请检查是否开启云点播服务。

短视频上传参数类错误？

请检查视频文件地址和图片地址是否正确，该路径下是否能找到相应文件。

短视频上传签名错误？

客户端在发起上传前，需要向 App 服务器请求上传签名，如果 App 服务器允许客户端上传，则会按照 签名规则 为客户端生成一个上传签名，客户端必须携带该签名，让云点播验证客户端上传是否被授权。

客户端上传签名的步骤如下：

1. 获取 API 密钥。
2. 拼接明文串。
3. 将明文串转为最终签名。
4. 服务搭建完毕之后，开发者可以通过云点播提供的工具来校验签名的正确性：
   - 签名生成工具：根据参数和密钥，快速生成签名。
   - 签名校验工具：对签名进行解析，得到生成签名时所使用的各项参数。

更多请参见 客户端上传签名。

视频上传时，最大允许上传的时长和大小有限制吗？

短视频 SDK 对上传视频的时长和大小没有限制。

是否可以上传图片？

暂不支持单独上传图片功能，但上传视频时可附带封面图，相关说明请参见 视频上传。

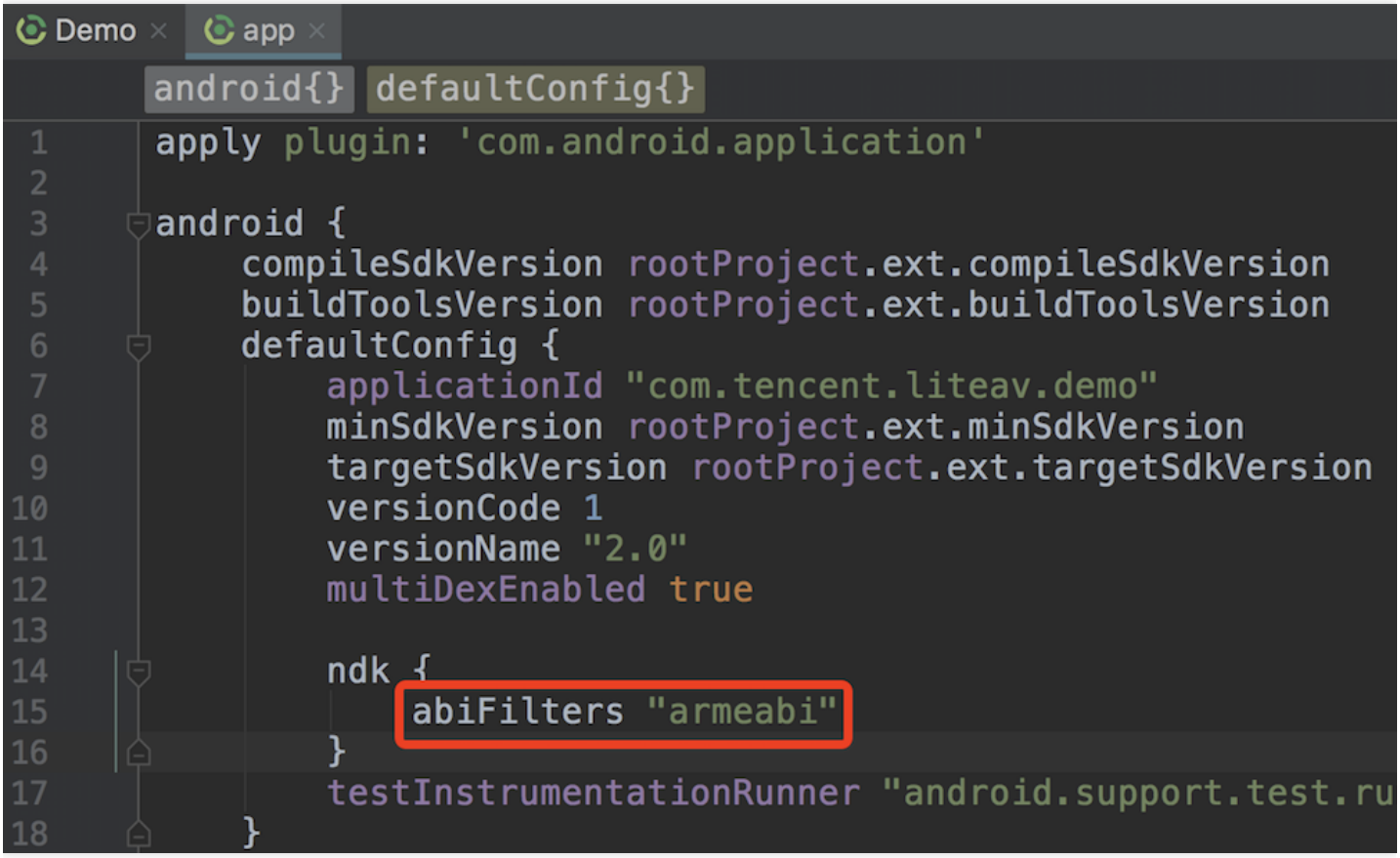
Android 编译问题

最近更新时间：2022-06-24 14:56:48

集成遇到异常怎么办？

```
java.lang.UnsatisfiedLinkError: No implementation found for void com.tencent.liteav.basic.log.TXCLog.nativeLogInit() (tried
om_tencent_liteav_basic_log_TXCLog_nativeLogInit__)
    at com.tencent.liteav.basic.log.TXCLog.nativeLogInit(Native Method)
    at com.tencent.liteav.basic.log.TXCLog.init(TXCLog.java:31)
    at com.tencent.liteav.basic.log.TXCLog.setConsoleEnabled(TXCLog.java:52)
    at com.tencent.rtmp.TXLiveBase.setConsoleEnabled(TXLiveBase.java:73)
```

您可以使用 armeabi 和 armeabi-v7a 架构。



如上图所示，请在 `app` 的 build.gradle 中指定 abiFilters 为“armeabi”。

同时集成两款以上 LiteAV 体系的 SDK 出现冲突怎么办？

如果您的项目中同时集成了两款以上的 LiteAV 体系的 SDK，就会出现符号冲突（symbol duplicate）的问题，这是由于 LiteAV 体系的 SDK 都使用了相同的基础模块。

要避免符号冲突问题，正确的做法是不要同时集成两个 SDK，而是集成全功能版 SDK：

所属平台	ZIP 包	Github	64位支持	安装包增量	安装包瘦身

iOS	DOWNLOAD	Github	支持	4.08M（arm64）	DOC
Android	DOWNLOAD	Github	支持	jar：1.5M so(armeabi)：6.5M so(armv7)：6.1M so(arm64)：7.3M	DOC

SDK 升级后，短视频的功能不能使用？

1. 如果使用的是 androidstudio，在替换新的 aar 后，请修改 `app` 的 build.gradle 中的 aar 引用，是否与您放入工程下 /libs 目录下的 aar **文件名称是否一致**。然后重新 clean 并且 build 一下您的工程。
2. 确认 SDK 版本，短视频 SDK 4.5 版本之后需要 License 支持。

请先申请 License，SDK 有**精简版**和**基础版**两种版本 License。

Android 端短视频如何设置暂停和进度条？

短视频播放是基于短视频的播放器进行实现的，因此进度条功能需要您**自行研发**。

Android 常见问题

最近更新时间：2022-11-14 18:18:58

功能相关

目前短视频录制生成的分辨率支持自定义吗？有哪些可定制化的输出？

短视频录制可定制参数包括 fps（每秒钟有多少帧画面），GOP（多少秒编出一个关键I帧）大小，视频码率（每秒钟编码器产生的音视频数据的多少），录制最大/最小时长，录制的分辨率以常量方式提供了四种分辨率供您选择：360 × 640、540 × 960、720 × 1280、1080 × 1920。

录制为什么以常量方式而不是用户自定义大小，原因如下：

- 以上四种是主流的录制分辨率。
- Android 手机兼容问题，不支持一些非主流的分辨率，会产生一些花屏、绿屏、马赛克。

通过调用 TXUGCRecord 类的 startCameraCustomPreview 接口，将自定义录制的参数传入，代码如下所示：

```
// 自定义配置
TXRecordCommon.TXUGCCustomConfig customConfig = new TXRecordCommon.TXUGCCustomCon
fig();
customConfig.videoResolution = TXRecordCommon.VIDEO_RESOLUTION_540_960;
customConfig.minDuration = mMinDuration; // 最小时长
customConfig.maxDuration = mMaxDuration; // 最大时长
customConfig.videoBitrate = mBiteRate; // 视频码率
customConfig.videoGop = mGop; // GOP 大小
customConfig.videoFps = mFps; // FPS
customConfig.isFront = mFront; // 是否前置摄像头
mTXCameraRecord.startCameraCustomPreview(customConfig, mVideoView);
```

Android 短视频录制结束，为什么没有收到 onRecordComplete 回调？

开始录制短视频前，请先通过调用 TXUGCRecord 类的 setVideoRecordListener() 接口设置录制回调的监听器
结束时，需要调用 TXUGCRecord 类的 stopRecord() 接口结束录制。

```
// 录制前
mTXCameraRecord = TXUGCRecord.getInstance(this.getApplicationContext());
mTXCameraRecord.setVideoRecordListener(this);

...
```

```
// 结束录制
mTXCameraRecord.stopRecord();
```

短视频退出录制，开启第二次录制，如何继续接着上一次内容录制？

Demo 在 onRecordComplete 回调之后，调用了 mTXCameraRecord.getPartsManager().deleteAllParts()，用于清除分片文件，因为 stopRecord 已经将分片文件合成完成。

如果录制退出，继续上次录制，不需要删除分片，不要调用 mTXCameraRecord.getPartsManager().deleteAllParts()。

```
@Override
public void onRecordComplete(TXRecordCommon.TXRecordResult result) {
TXCLog.i(TAG, "onRecordComplete, result retCode = " + result.retCode + ", descMsg = " + result.descMsg + ", videoPath + " + result.videoPath + ", coverPath = " + result.coverPath);
if (mTXRecordResult.retCode < 0) {
Toast.makeText(TCVideoRecordActivity.this.getApplicationContext(), "录制失败，原因：" + mTXRecordResult.descMsg, Toast.LENGTH_SHORT).show();
} else {
mDuration = mTXCameraRecord.getPartsManager().getDuration(); //录制的总时长
if (mTXCameraRecord != null) {
mTXCameraRecord.getPartsManager().deleteAllParts(); //删除多次录制的分片文件
}
startPreview(); //进去预览界面
}
```

为什么短视频录制设置背景音没有生效？

设置背景音乐一定要在启动录制（TXUGCRecord 的 startRecord）接口之前设置才能生效。代码调用顺序参照下面示例：

```
TXRecordCommon.TXUGCSimpleConfig simpleConfig = new TXRecordCommon.TXUGCSimpleConfig();
simpleConfig.videoQuality = TXRecordCommon.VIDEO_QUALITY_MEDIUM;
simpleConfig.minDuration = mMinDuration;
simpleConfig.maxDuration = mMaxDuration;
// 1、首先开启预览
mTXCameraRecord.startCameraSimplePreview(simpleConfig, mVideoView);
// 2、再设置背景音乐的路径并播放背景音乐
mBGMDuration = mTXCameraRecord.setBGM(mBGMPath);
mTXCameraRecord.playBGMFromTime(0, mBGMDuration);
// 3、启动录制(customVideoPath:录制后视频路径, customPartFolder：录制视频的文件夹, customCoverPath：录制后视频的封面路径)
```

```
int result = mTXCameraRecord.startRecord(customVideoPath, customPartFolder, custo
mCoverPath);
```

录制是否有拍照功能？

短视频 SDK 有拍照功能，调用 TXUGCRecord 类的 snapshot 接口，以 TXRecordCommon.ITXSnapshotListener 回调异步返回拍照的图片，代码示例如下：

```
private void snapshot() {
if (mTXCameraRecord != null) {
mTXCameraRecord.snapshot(new TXRecordCommon.ITXSnapshotListener() {
@Override
public void onSnapshot(Bitmap bmp) {
// 拍照的图片
saveBitmap(bmp);
}
});
}
}
```

变速录制速度的倍数是多少？

变速录制不支持自定义速度

定义	TXRecordCommon 中对应常量	倍数
极慢速	RECORD_SPEED_SLOWEST	0.5倍
慢速	RECORD_SPEED_SLOW	0.8倍
标准	RECORD_SPEED_NORMAL	1倍
快速	RECORD_SPEED_FAST	1.25倍
极快速	RECORD_SPEED_FASTEST	1.5倍

变速录制通过调用 TXUGCRecord 的 setRecordSpeed(record)，设置不同的录制速度

```
mTXCameraRecord.setRecordSpeed(TXRecordCommon.RECORD_SPEED_FAST);
```

导入视频的格式要求？是否支持导入分辨率大于720P（例如2K，4K）的视频？导入文件限制有具体大小吗？

导入视频目前 Android 端仅支持 MP4，分辨率没有做限制，导入文件不限制大小。

版权所有：腾讯云计算（北京）有限责任公司

- 导入视频的分辨率没有做限制，不管原视频多大，经过预处理后最大是720P。
- 为了快速的导入视频，SDK4.7后可以不经过预处理（一些功能收到限制，倒放，单针预览等），对于分辨率大于720P的视频，建议需要加上预处理，因为预览是将每一帧解码，一些手机的性能不好，导致解码一帧并渲染到界面的时间过长，导致卡顿。

目前短视频编辑支持哪种格式的背景音乐？

目前仅支持 MP3 和 M4A 类型。

目前短视频编辑有哪些可定制化的输出？

短视频编辑可定制视频码率（SDK4.5及以上）、音频码率（SDK4.7及以上）、分辨率以常量方式提供了几种分辨率供您选择：360 x 640、480 x 640、540 x 960、720 x 1280、1080 × 1920。

分辨率	TXVideoEditConstants 中对应常量
360x640	VIDEO_COMPRESSED_360P
480x640	VIDEO_COMPRESSED_480P
540x960	VIDEO_COMPRESSED_540P
720x1280	VIDEO_COMPRESSED_720P
1080x1920	VIDEO_COMPRESSED_1080P

```
//设置输出视频码率
mTXVideoEditer.setVideoBitrate(3600);
//设置输出分辨率
mTXVideoEditer.generateVideo(TXVideoEditConstants.VIDEO_COMPRESSED_720P, mVideoOutputPath);
```

通过短视频录制功能录制的视频，其中的音频是可以剥离出来的吗？

目前短视频录制不支持同时录制 BGM 和人声，所以进入编辑后，重新设置 BGM，可以将原声音量设为0，达到替换 BGM 的目的，代码如下所示：

```
// 设置视频原声音量大小（设为0，去掉录制的 BGM）
mTXVideoEditer.setVideoVolume(0.0f);
// 设置本地的背景音乐路径
String bgmPath = getBGMPath();
mTXVideoEditer.setBGM(bgmPath);
// 设置背景音乐音量大小，范围0.0f-1.0f
mTXVideoEditer.setBGMVolume(1.0f);
```

版权所有：腾讯云计算（北京）有限责任公司

预览画面在同一个 Activity 窗口和全屏模式如何切换？

动态修改传入 SDK 视频预览 View 父布局的大小，SDK 内部会根据父布局的大小，根据视频宽高动态调整视频的大小。

SDK 接口的调用顺序：

先进行 stopPlay，在修改传入 SDK 的 FrameLayout 的宽高，调用 initWithPreview(parm)，将新的 FrameLayout 承载播放组建的 layout 传入，再次 startPlay。

```
// 停止播放
mTXVideoEditer.stopPlay();
if (isFullScreen) {
// 如果是全屏模式，则下面切换成窗口模式
FrameLayout.LayoutParams params = new FrameLayout.LayoutParams(ViewGroup.LayoutPa
rams.MATCH_PARENT, 1500);
mVideoPlayerLayout.setLayoutParams(params);
initPlayerLayout(false);
isFullScreen = false;
} else {
// 如果是窗口模式，则下面切换成全屏模式
FrameLayout.LayoutParams params = new FrameLayout.LayoutParams(ViewGroup.LayoutPa
rams.MATCH_PARENT, ViewGroup.LayoutParams.MATCH_PARENT);
mVideoPlayerLayout.setLayoutParams(params);
initPlayerLayout(false);
isFullScreen = true;
}
// 开始播放
mTXVideoEditer.startPlayFromTime(startTime, endTime);
// 重新设置预览View
private void initPlayerLayout(boolean isFullScreen) {
TXVideoEditConstants.TXPreviewParam param = new TXVideoEditConstants.TXPreviewPar
am();
param.videoView = mVideoPlayerLayout;
if (isFullScreen) {
param.renderMode = TXVideoEditConstants.PREVIEW_RENDER_MODE_FILL_SCREEN;
} else {
param.renderMode = TXVideoEditConstants.PREVIEW_RENDER_MODE_FILL_EDGE;
}
mTXVideoEditer.initWithPreview(param);
}
```

短视频编辑时，腾讯云短视频 Demo 是把“剪辑”和“滤镜”等功能放在一个页面处理。不过，我们公司产品是把“剪辑”功能和“滤镜”分成两个页面？

可以先进行裁剪（setCutTimeFrom）+ 预处理（processVideo）同时执行，结果生成一个裁剪后的视频预处理完的视频，再进行各种编辑的操作，将裁剪设置成整个时长（setCutTimeFrom），最后调用 generateVideo 生成视频，

防止压缩两次导致画质降低。

> 注意：
> 在预处理进行裁剪了，生成完的预处理视频，在最后生成前，一定要将裁剪时长设置为整个视频时长，不然还会再次进行裁剪。

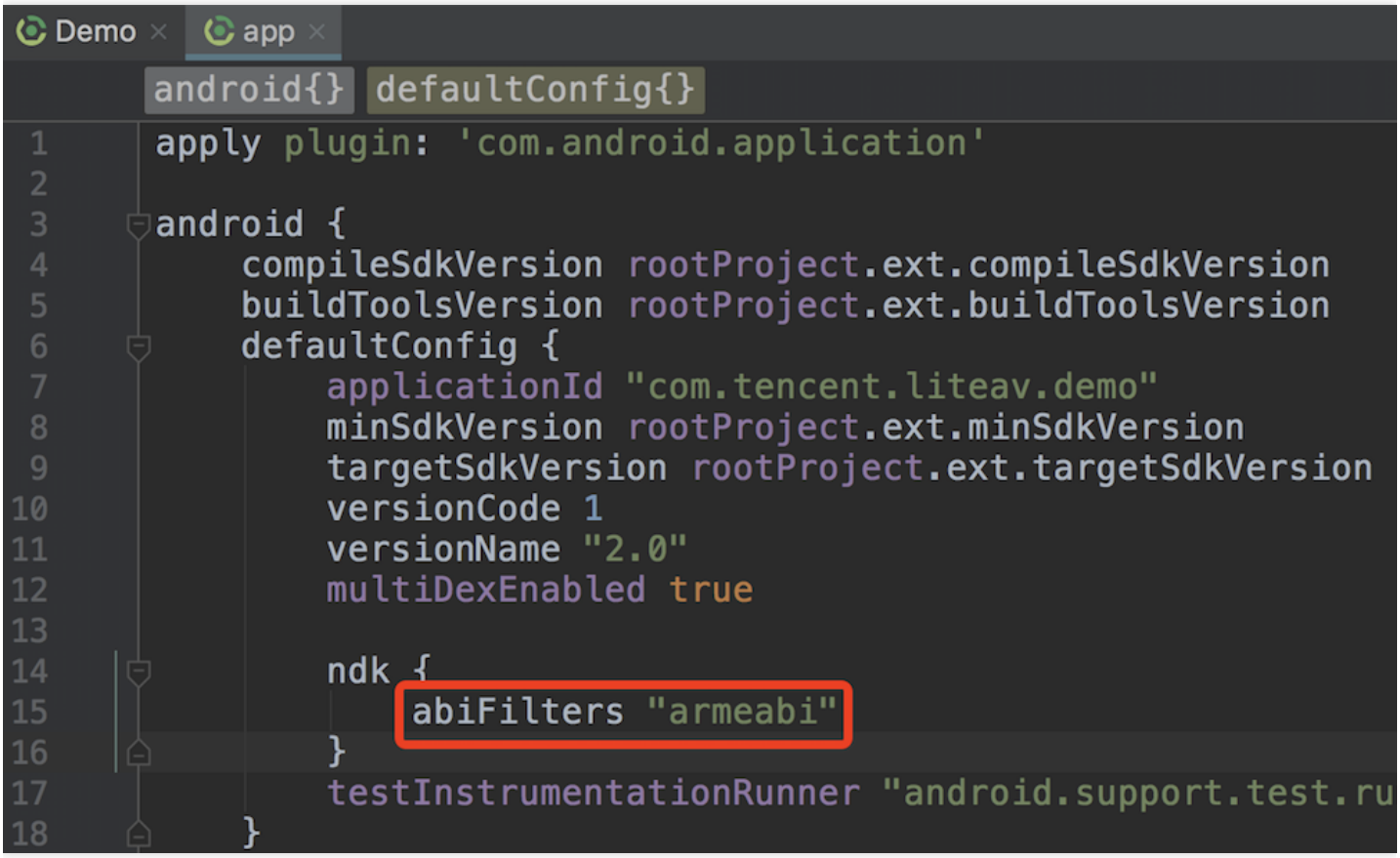
```
//裁剪页面
mTXVideoEditer = new TXVideoEditer(mContext);
mTXVideoEditer.setCutFromTime(mTCVideoEditView.getSegmentFrom(), mTCVideoEditVie
w.getSegmentTo());
mTXVideoEditer.processVideo();

// 将裁剪设置成整个时长 (setCutTimeFrom)
mTXVideoEditer.setCutFromTime(0, mVideoDuration);
//跳转到特效页面，进行生成
mTXVideoEditer.generateVideo(TXVideoEditConstants.VIDEO_COMPRESSED_720P, mVideoO
utputPath);
```

编译相关

集成遇到异常怎么办？

```
java.lang.UnsatisfiedLinkError: No implementation found for void com.tencent.liteav.basic.log.TXCLog.nativeLogInit() (tried
om_tencent_liteav_basic_log_TXCLog_nativeLogInit__)
    at com.tencent.liteav.basic.log.TXCLog.nativeLogInit(Native Method)
    at com.tencent.liteav.basic.log.TXCLog.init(TXCLog.java:31)
    at com.tencent.liteav.basic.log.TXCLog.setConsoleEnabled(TXCLog.java:52)
    at com.tencent.rtmp.TXLiveBase.setConsoleEnabled(TXLiveBase.java:73)
```

您可以使用 armeabi 和 armeabi-v7a 架构。



如上图所示，请在 `app` 的 build.gradle 中指定 abiFilters 为“armeabi”。

同时集成两款以上 LiteAV 体系的 SDK 出现冲突怎么办？

如果您的项目中同时集成了两款以上的 LiteAV 体系的 SDK，就会出现符号冲突（symbol duplicate）的问题，这是由于 LiteAV 体系的 SDK 都使用了相同的基础模块。

要避免符号冲突问题，正确的做法是不要同时集成两个 SDK，而是集成全功能版 SDK：

所属平台	ZIP 包	Github	64位支持	安装包增量	安装包瘦身
iOS	DOWNLOAD	Github	支持	4.08M（arm64）	DOC
Android	DOWNLOAD	Github	支持	jar：1.5M so(armeabi)：6.5M so(armv7)：6.1M so(arm64)：7.3M	DOC

SDK 升级后，短视频的功能不能使用？

1. 如果使用的是 androidstudio，在替换新的 aar 后，请修改 `app` 的 build.gradle 中的 aar 引用，是否与您放入工程下 /libs 目录下的 aar **文件名称是否一致**。然后重新 clean 并且 build 一下您的工程。

2. 确认 SDK 版本，短视频 SDK 4.5 版本之后需要 License 支持。

请先申请 License，SDK 有**精简版**和**基础版**两种版本 License：

- 详细价格请参见 价格文档
- 如需使用美颜特效等高级功能，请参见 腾讯特效 SDK。

Android 端短视频如何设置暂停和进度条？

短视频播放是基于短视频的播放器进行实现的，因此进度条功能需要您**自行研发**，相关功能实现说明可参见 播放器 SDK—进度展示。

iOS 常见问题

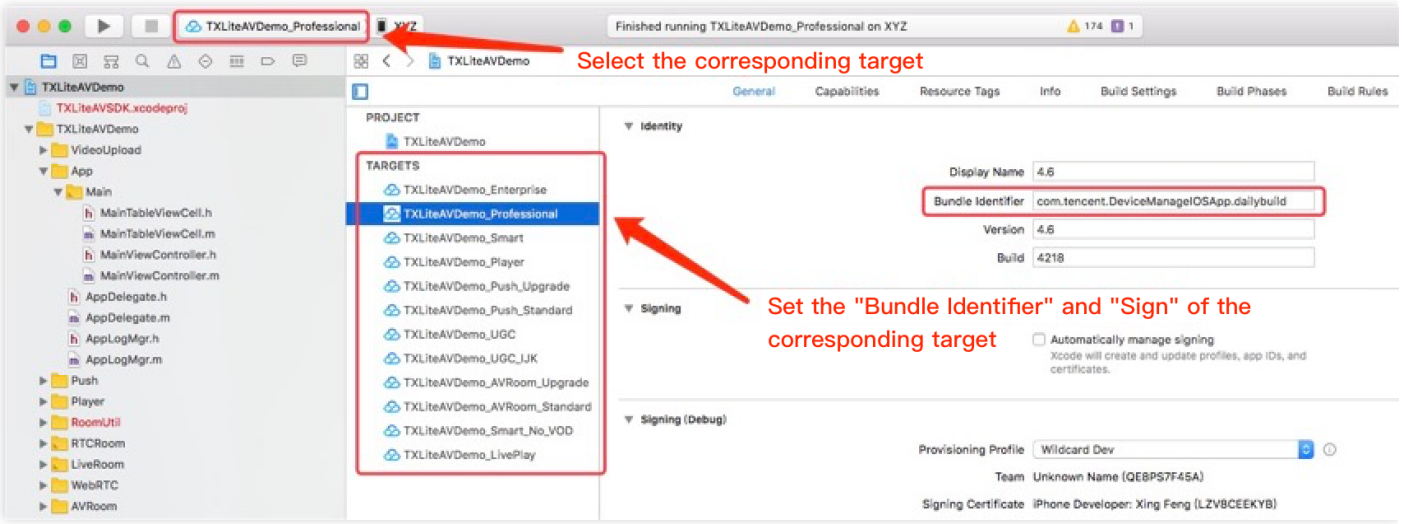
最近更新时间：2022-08-16 17:09:23

关于 TXUGCPublish.h 的问题？

从4.5版本开始，`TXUGCPublish` 相关的类从 SDK 上移到了 Demo 层，开发者如果需要使用，使用时直接把 `VideoUpload` 目录整个拖入自己的工程即可。

Xcode 中直接运行 Demo 报错？

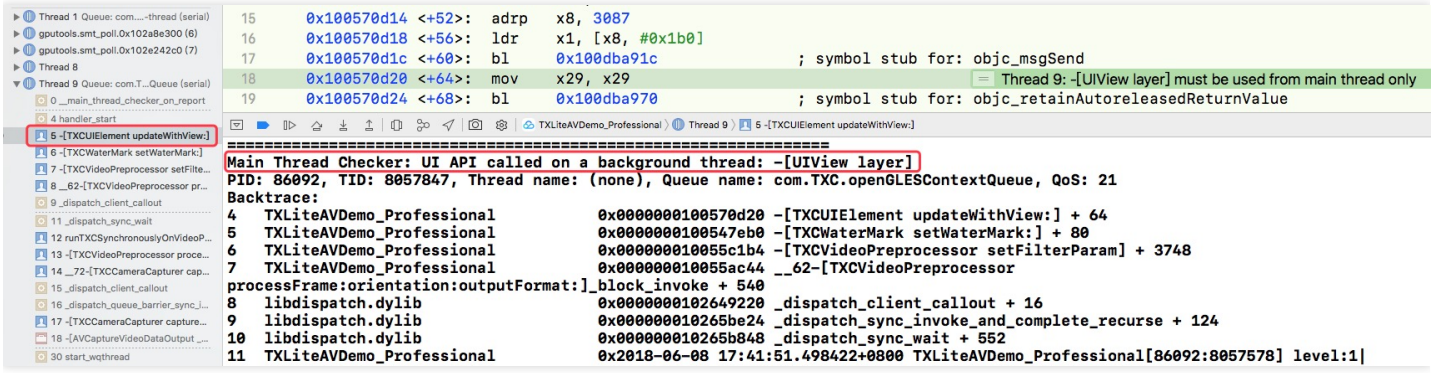
- 报错信息：



- 解决方法：执行 pod install。

连接 Xcode 调试，短视频录制报错？

连接 Xcode 调试，短视频录制报错，报错信息：`Main Thread Checker: UI API called on a background thread`



- 原因：某些 API（一般是 UI 相关的）需要在主线程调用，如果在非主线程调用，同时有勾选 `Main Thread Checker` 的话，就会报错。
- 解决办法： `Product` > `Scheme` > `Edit Scheme` > `Run` > `Diagnostics` ，取消勾选 `Main Thread Checker` 。

> 说明：
> 该问题在4.9版本已经修复。

使用 SDK 时报找不到头文件？

执行 pod install 即可。

运行工程时报找不到类别方法或者 crash？

SDK 用到了一些类别的方法，加载类别方法需要在工程配置： `Build Settings` > `Linking` > `Other Linker Flags` 添加 `-ObjC` 。

录制短视频时设置背景音乐无效？

1. 确定传的 BGM path 下有没有文件，以及是否可以正常播放。
2. 确定接口的调用顺序： `startCameraSimple:preview:` > `setBGM:` > `startRecord` 。

> 注意：
> 很多接口调用有时序要求，不然会无效。一般在注释上会有说明。
> 例如短视频录制的 `setVideoResolution:` 、 `setVideoBitrate:` 、 `setAspectRatio:` 等接口都需要在 `startRecord` 之前设置才有效。

录制设置 BGM 不能循环播放？

目前逻辑暂未支持循环播放。

录制设置 BGM， `endTime` 时没有完成回调？

如果设置的 `endTime` 小于音乐文件总时长，4.6及之前的版本是在 BGM 播放完后才会触发完成回调，从4.7版本开始在 `endTime` 时触发完成回调。

为什么录制时第一次打开摄像头比较慢？

苹果手机摄像头第一次打开时（冷启动）耗时相对较长，通过系统接口打开摄像头也是如此。

因为摄像头打开的操作不适合放在子线程去做，经过测试在子线程中进行打开摄像头操作耗时会更大，并且在主线程连续打开/关闭摄像头的时候，子线程的响应延迟会更高，体验不好。

返回继续录制怎么实现？

在第一次录制完成的时候，不要调用 `stopRecord` 和 `stopCameraPreview` （调用之后不能再继续录制，只能重新录制），可以调用 `pauseRecord` ，然后通过 `TXUGCPartsManager.getVideoPathList` 获取已经录取的视频片段，通过 `TXVideoJoiner.joinVideo` 合成最终视频（4.5之前版本），还可以直接调用 `TXUGCPartsManager.joinAllParts` 合成最终视频，这个方法合成速度更快（4.5以后版本支持），这样当返回继续录制的时候，所有的录制状态都在，可以继续录制。

短视频录制完成时收不到完成回调？

- 确定有没有调用 stopRecord，只有调用 stopRecord 后才会有完成回调。
- 确定函数的调用是否都在主线程。

录制过程中用其他播放器播放视频，返回继续录制，声音录制不了？

iOS 中的 AudioSession 是所有音视频应用共用的，使用其他播放器播放的时候，AudioSession 会被占用，播放结束时如果 AudioSession 没有让出或者让出不及时，会导致录制模块的 AudioSession 失效，SDK 提供了 `-(void) pauseAudioSession` 和 `-(void) resumeAudioSession` 两个接口，在去其他播放器预览的时候先调用 `pauseAudioSession` ，返回继续录制前调用 `resumeAudioSession` 。

为什么录制出来的视频不清晰？

码率和分辨率不匹配，录制出来的视频就会不清晰。可以通过适当增大码率、开启B帧来提升画质。

视频编辑时退后台再回到前台，视频生成失败？

生成视频默认采用的是硬编码（编码效率高，编码出来的图像效果好），硬编码器在程序进后台后会停止工作，从而导致视频生成失败。SDK 提供了两个接口 `pauseGenerate` 和 `resumeGenerate` ，App 进后台时可以调用 `pauseGenerate` 暂停视频生成，App 回到前台后再调用 `resumeGenerate` 继续视频生成。

> 注意：
> 调用 `resumeGenerate` ，SDK 将重启硬编码器，有一定的概率重启失败，或重启后前几帧数据编码失败。此时，SDK 内部会在 `TXVideoGenerateListener` 抛出错误事件，收到错误事件后需要重新生成视频。

文件上传失败？

文件上传状态码：

```
typedef NS_ENUM(NSInteger, TXPublishResultCode)
{
    PUBLISH_RESULT_OK                           = 0,    // Published successfully
    PUBLISH_RESULT_UPLOAD_REQUEST_FAILED        = 1001, // Step 1. The "file upload request" failed to be sent
    PUBLISH_RESULT_UPLOAD_RESPONSE_ERROR        = 1002, // Step 1. An error response is received for the "file upload request"
    PUBLISH_RESULT_UPLOAD_VIDEO_FAILED          = 1003, // Step 2. The "video file" failed to be uploaded
    PUBLISH_RESULT_UPLOAD_COVER_FAILED          = 1004, // Step 2. The "cover file" failed to be uploaded
    PUBLISH_RESULT_PUBLISH_REQUEST_FAILED       = 1005, // Step 3. The "short video release request" failed to be sent
    PUBLISH_RESULT_PUBLISH_RESPONSE_ERROR       = 1006, // Step 3. An error response is received for the "short video release request"
};
```

1. 确定上传的文件是否在本地沙盒，如果上传媒体库的文件，需要先 copy 到本地沙盒。
2. 返回错误码1002：签名有问题、时间戳过期、点播服务问题（未开通或停服）。
3. 返回错误码1003：请求参数问题、上传文件格式不支持。

短视频录制是否有拍照功能？

短视频 SDK 可以实现拍照功能，开始预览后调用 `TXUGCRecord` 类的 `snapshot` 接口获取图片即可。

集成时一直报错“Use of undeclared identifier 'TXVideoInfo'”？

该错误是编译器没有检测到 TXVideoInfo 类，建议检查 SDK（framework）是否正确，可依照 文档步骤 重新导入工程。

调用视频合成报错“-1，Failed to enable encoder” 怎么办？

1. 请确认问题是否为必现问题，建议更换机型测试。
2. 可下载最新版的 Demo 中复现一下问题。若问题是必现的，请提供 完整的日志信息 并 提工单 解决。

License 常见问题

最近更新时间：2022-06-29 14:41:17

关于 License 版本问题？

短视频 SDK 分精简版和基础版，从4.5版本开始需要 License，基础版需要短视频的 License（TXUgcSDK.licence），把 License 放到工程目录，并修改为对应的名字即可。

4.9版本开始 License 使用方式有改变，可以选择是否把 License 打包到项目中。使用时需要调用 `setLicenceURL：` 和 `key：` 接口设置 License 的 URL 和 key。

> 注意：
> 4.5 - 4.7版本的 SDK 不支持 License 自动续期，4.9版本开始才支持自动续期。4.9的 SDK 可以兼容之前的 Licence（url 和 key 不能传 null，可以随便传个字符串），但是新的 License 无法在4.9之前的 SDK 上使用。

测试 License 到期后是否可以延期？

您可以免费申请测试版 License（免费测试有效期为14天，可续期1次，共28天）体验测试。到期后请尽快购买正式 license，详情可提交工单咨询或联系商务。

> 注意：
> 试用期内申请测试续期，则续期到期时间以申请测试时刻为准；若试用期结束后申请测试续期，则续期到期时间以申请测试续期时刻为准。
>
> - 当申请测试开始时间为 `2022-05-25 11:34:55` ，则14天后到期时间为 `2022-06-09 00:00:00` 。
> - 免费续期一次时，若在试用期14天内申请续期，则到期时间为 `2022-06-23 00:00:00` ；若在试用期14天结束后申请续期，申请续期的时间为 `2022-07-03 22:26:20` ，则续期的到期时间为 `2022-07-18 00:00:00` 。

测试 License 能否更改 Android 的 PackageName 和 iOS 的 BundleID?

测试 License 支持更改，在 云点播控制台 选择测试 License 信息右上角，单击**编辑**即可进行修改。

正式 License 能否更改 Android 的 PackageName 和 iOS 的 BundleID?

正式版 License 不能更改 Package Name 和 Bundle ID。请您在添加正式版 License 先核对包名在应用商店里是否被占用，提交后不支持修改和替换。

版权所有：腾讯云计算（北京）有限责任公司

License 可以同时支持多个 App 吗？

一个 License 只能对应一个 PackageName 和 BundleID，不支持多个 App。

License 应该如何确认绑定关系（Android 的 PackageName 和 iOS 的 BundleID）？

在填写时用户需要确认绑定正式上架 App Store 的 iOS 所对应的 Bundle ID 和正式上架应用市场 Android 的 Package Name。

续期 License 时出现“license not exist”问题，如何解决？

您可登录 **云点播控制台** > **License 管理** > **SDK License** 根据以下方式排查：

1. 请确认是否在**管理员**页面进行 License 绑定续期。
2. 如果您是在**非管理员**页面下进行操作，请联系**管理员**协助您进行 License 变更操作。

License 无法添加/新增，如何解决？

查看绑定页面是否为**管理员**页面，请选择管理员页面进行绑定。

License 校验失败，如何排查？

建议您可以参考以下几点进行排查：

- 确认您的 License 是否在有效期内。
- 确认 License 信息里的 Package Name 是否与项目里面的包名一致。
- 确认 License 中的 LicenseUrl 协议是否为 HTTPS。

> 说明：
> 若上述方法无法解决您的问题，建议**卸载应用重新安**装或 提工单 解决。

没有 bundleid，Android 端是不是无法使用 License？

bundleid 类似于 Android 端的 package name，若您不集成 iOS 端，可随意填写，不使用即可。

个人购买的短视频 SDK License 可以用于企业吗？

短视频 SDK 暂仅支持购买所在账号进行使用，暂无个人实名认证以及企业实名认证的限制。

为什么接收不到 License 到期等相关消息通知？

您可以通过在 消息订阅 中订阅音视频终端 SDK，配置**站内信/邮件/短信/企微**等消息接收渠道，接收正式版 License 到期提醒。短视频正式版 License 将在到期时间距离当前时间为30天、15天、7天、1天时各向您发送一次到期提醒，提示您及时续费以免影响正常业务运行。

版权所有：腾讯云计算（北京）有限责任公司