

短视频处理 LanSoEditor-SDK 之 Android 平台集成步骤

一，集成方式（Android Studio 环境下）：

说明 1，为了方便您在实际编程中查看调用相关的方法，我们开放了一些 native 的接口方法的各种源代码，因此需要拷贝 lansosdk 源代码文件夹，如后期你们混淆代码，JAVA 的 JNI 机制是一定要保留 native 字符串方法名字的，故请在脚本中增加：

```
-keep public class com.lansosdk.videoeditor.** {  
    <fields>;  
    <methods>;  
}  
-keep public class com.lansosdk.videoplayer.** {  
    <fields>;  
    <methods>;  
}  
-keep class com.lansosdk.box.**{*;  
}
```

我们其他的 jar 文件，已经在 sdk 提供给您的时候，做过混淆操作，请勿再次混淆。

说明 2，当前 SDK 提供 armeabi-v7a 指令集的库文件，如果您实际项目需要 armeabi 和 arm64-v8a 的指令集，请联系我们，我们另外编译提供给您。

说明 3，此集成方式为大部分开发者常用方式，如果您清晰 Android Studio 的 module 机制，也可以整理成一个单独的 module，以方便后期更新升级。

说明 4，如果您在未授权的情况下，集成我们 SDK 到您项目中，请修改您 app 的名字为我们 demo 的名字方可运行，不然底层无法运行。授权后将不再限制。

SDK 由 4 部分组成：

1. Armeabi-v7a： native 层的库文件。

拷贝《Demo 工程》中 libs 下的 armeabi-v7a 文件夹到您工程中的 app/src/main 下的 jniLibs。

2. Lansosdk 文件夹： 开放源代码的各种方法和类。

此源代码开放目的是为了让您在开发中，只需要点击一下，就可以看到每个方法的功能和详细的注释。请不要修改我们 lansosdk 文件夹中的内容，以方便后期的更新，

您可以在外部继承我们的类，然后在里面写相关的方法。

请拷贝《Demo 工程》中 app\src\main\java\com 文件夹下的 lansosdk 文件夹 到您工程中的 app\src\main\java\com 文件夹里。如果没有 com 文件夹，则新建，
里面的源代码与 java 的 JNI native 相关联，包名和路径不许更改，请注意！！
里面的源代码与 java 的 JNI native 相关联，包名和路径不许更改，请注意！！
里面的源代码与 java 的 JNI native 相关联，包名和路径不许更改，请注意！！
【请务必确认我们的 lansosdk 文件是在 app/src/main/java/com 下】。

3. lansongeditor_hx.jar: java 层代码。

拷贝《Demo 工程》中 libs 下的 jar 文件到您工程的 app/libs 下，需要在 build.gradle 中声明编译此 jar。

4. LSResource 文件夹（Android 专业版本使用）

拷贝 assets 文件夹下的整个 LSResource 文件夹到您工程的 assets 文件夹下。

5. 在您的主启动加载库的代码里，一般是 MainActivity 或 Application.java 中，初始化我们的 SDK：

```
LanSoEditor.initSDK(getApplicationContext(),null);
```

6. 在您的 Manifest.xml 文件里增加读写，摄像机，录音等权限：

7. 刷新您的工程后， 可以按照我们《Demo 工程》里的调用方法，来实现您的各种功能。

8. 遇到问题，请首先执行

```
Log.i(TAG,"当前音视频文件信息是:"+MediaInfo.checkFile("你的文件路径"));
```

看打印什么信息，此方法可检查出 60%以上的错误信息。

9. 强烈建议您在先看明白我们的 SDK 演示调用流程后，再集成到您项目中， 这样可事半功倍，并且我们的 SDK 每个 Activity 的演示是尽量简单的，一般 3 个步骤就可完成。

建议前期直接在我们 demo 中开发，以减少您的不必要的时间。

10. 如有开发中有其它问题， 请直接联系我们， 不要自己在那里闷头猜测，或许我们 SDK 的一点小改动， 就可解决您视频功能的大问题。