

短视频处理 LanSoEditor-SDK 之下载地址和集成说明

1. Android 短视频编辑《基本版本》SDK:

下载地址:

https://github.com/LanSoSdk/LanSoEditor_common (主地址)

https://code.csdn.net/LanSongSDK/lansoeditor_common (备用地址)

主要功能:

音视频的: 裁剪,剪切,缩放,分离,合并,转换,拼接,水印,叠加,混合,转码,旋转等场合。

核心优势是: 我们在 ffmpeg 的底层实现了 Android 平台的硬件加速解码, 和硬件加速编码, 让你在 Android 平台上使用 ffmpeg 更快的执行您的命令。

兼容完整的 FFMPEG 命令。

我们在项目中提供了大约 40 个常用的方法并写了详细的说明注释,基本满足一般视频编辑的需求。

我们另外提供了扩展接口,您完全可以根据强大的 FFMPEG 命令来扩展您需要的功能。

2. Android 短视频编辑《高级版本》SDK:

下载地址:

https://github.com/LanSoSdk/LanSoEditor_advance (主地址)

https://code.csdn.net/LanSongSDK/lansoeditor_advance (备用地址)

主要功能:

类似画家作画一样,我们设计了多种画笔和画板,用不同的画笔和画板的组合,您可以任意的设计出您要的视频效果,这样的画笔有 VideoPen/BitmapPen/ViewPen/CanvasPen 等等。每种画笔就支持滤镜操作。

比如你要在视频上增删图,把"视频画笔"VideoPen 和"图片画笔"BitmapPen 增加到 DrawPad 画板上就可以了

比如增加文字,动画,线条,炫酷的 UI 等,那用到视频画笔和 ViewPen "UI 画笔".

每种画笔都支持旋转,缩放,平移,闪烁等操作;

您可以实现任意的媒体转换为视频;

实现任意的叠加, 实现你定义各种炫酷个性化如涂鸦, 标记, 覆盖, 整理, 图片 3D, 拖动, 炫酷文字等效果;

接口调用及其简单, 逻辑清晰易懂, 即使您是刚开始做 Android 开发, 也可以直接使用;
视频处理从未如此简单;

3. IOS 短视频编辑 SDK

下载地址

https://github.com/LanSoSdk/LanSongEditor_ios (未发布)

主要功能

当前还没有完整的发布, 链接中的代码仅做参考, 可关注我们的 QQ 空间的变动, 即将推出, 谢谢。

4. 安卓平台集成到项目中的方式是:

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>;  
}
```

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。

1. 拷贝《Demo 工程》中 libs 下的 armeabi-v7a 文件夹到 您工程中的 libs 下 (eclipse 环境) 或 app/src/main 下的 jniLibs (Android Studio 环境)。
2. 拷贝《Demo 工程》中 libs 下的 jar 文件到 您工程的 libs 下 (eclipse 环境) 或 app/libs 下 (Android Studio 环境, 需要在 build.gradle 中声明编译此 jar)。

3. 拷贝《Demo 工程》中 `src/com` 文件夹下的 `lansosdk` 文件夹 到您工程中的 `src/com` 文件夹里(Eclipse 环境)或 `app/src/main/java/com` 文件夹里(Andriod Studio 环境), 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。
4. 在您的主启动加载库的代码里, 一般是 `MainActivity` 或 `Application.java` 中, 加载和初始化我们的 `sdk` 底层 `so` 库文件:

```
LoadLanSongSdk.loadLibraries();
LanSoEditor.initSo(getApplicationContext(),null);
```

5. 在您的 `Manifest.xml` 文件里增加读写权限:

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

6. 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。
7. 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。

5. 下载评估

我们的合作方式:

您可先下载后, 大概浏览下我们的工程结构, 熟悉后, 可以直接增加到您项目中去, 实际使用评估后,再和我们合作, 取得授权 SDK 文件, 永久使用。

我们提供的服务:

一个授权文件支持一款 APP, 不限制使用时间, 永久使用, 并可以在后续的版本更新中永久更新。

提供一年的技术支持, 并支持一些小功能的增加, FFMPEG 方法的封装, 各种小方法的修改。

在实际使用中, 遇到的普通问题(可重现, 可调试的问题), 在三个工作日内解决。其他各种资料, 文档说明可联系我们索取。

我们 SDK 已经应用在许多上线 APP 中, 可联系我们, 索取相关的上线 APP 信息。