信号与系统大作业之“B 站，我来了!”

作业报告

学号：2019011008

班级：无 92

姓名：刘雪枫

目录

一、实验目的

根据给定的背景音乐和视频片段进行分析，生成混剪视频，以使得生成视频给人的视觉与听觉感受的激烈程度尽可能大。

二、开发环境

本次作业所用开发工具为 Windows10 系统上的 MATLAB R2021a (64 位)、视频合成工具为 Windows Subsystem for Linux (Ubuntu 20.04 LTS) 系统上的 ffmpeg v4.2.4。

使用的开源工具地址为：https://github.com/zhangzw16/Project-for-Signals-and-Systems-2021。

三、实验内容

（一）提取节奏点

1. 半波整流

1) 使用 `audioread` 函数读取背景音乐的向量。如果读取的为单声道则不做处理，如果为双声道则将两个声道的向量相加为一个向量。
2) 将得到的向量 x 进行平方得到向量 y_1，在加窗得到 y_2，然后差分提取变化点得到 y_3，然后进行半波整流，即可得到 y_4。即：

$$y_1(n) = x^2(n)$$

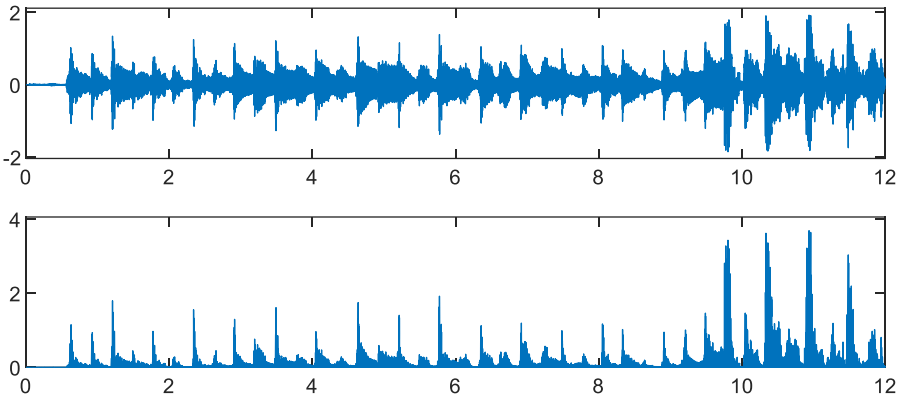
$$y_2(n) = \sum_{j=0}^{M-1} w_j y_1(n-j)$$

$$y_3(n) = y_2(n) - y_2(n-1)$$

$$y_4(n) = \max\{y_3(n), 0\}$$

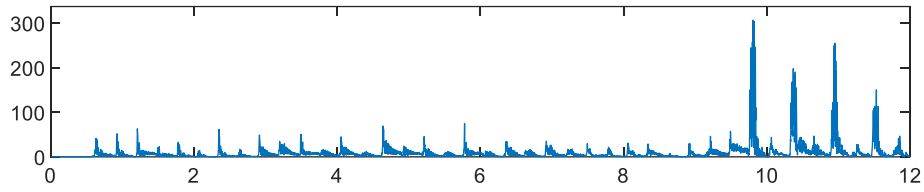
其中，$\{w_0, w_1, \ldots, w_{M-1}\}$表示窗函数。

得到的 x 和 y_1 的前 12 秒的图象如下：

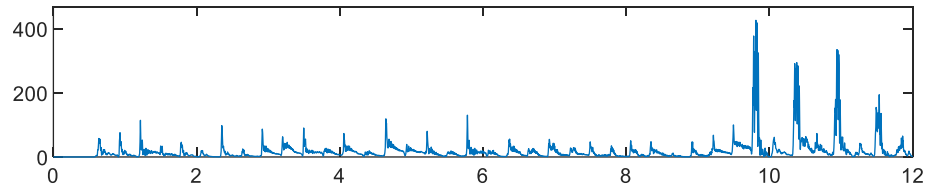


本次作业计划使用 MATLAB 函数 `hanning` 生成汉宁窗。设音频的采样率为 Fs。当窗的长度分别取不同值时，生成的能量包络图分别如下：

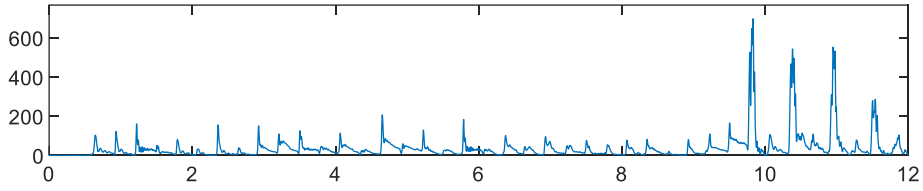
窗的长度为 Fs / 160：



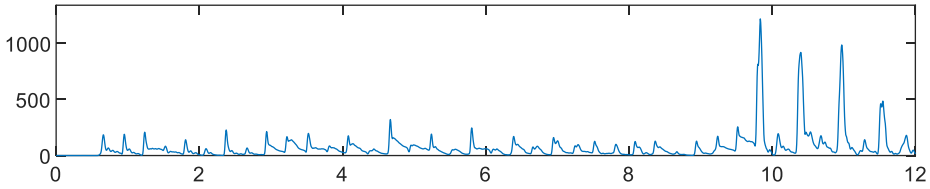
窗的长度为 Fs / 80：



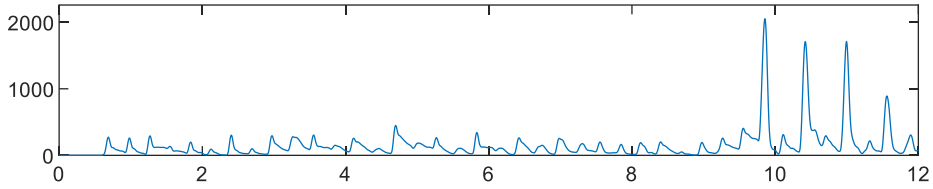
窗的长度为 Fs / 40：



窗的长度为 Fs / 20：

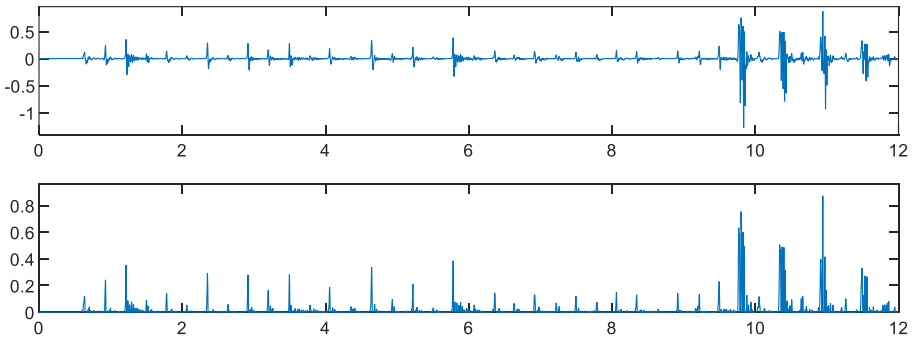


窗的长度为 Fs / 10：



可以看到，使用的窗的长度越长，得到的能量包络越平滑；而使用的窗的长度越短，得到的能量包络越粗糙。因此窗的长度不能过长也不能过短：若窗的长度过长，则得到的包络过于平滑，难以精确分辨出每个时间点的能量信息；而若窗的长度过短，则包络的效果过差，没有完全滤除高频分量，因此得到的能量信息不准确。通过观察，取窗的长度为 Fs / 40 可能效果较好。

3) 取长度为 Fs / 40 的汉宁窗，进行差分和半波整流，结果分别如下：



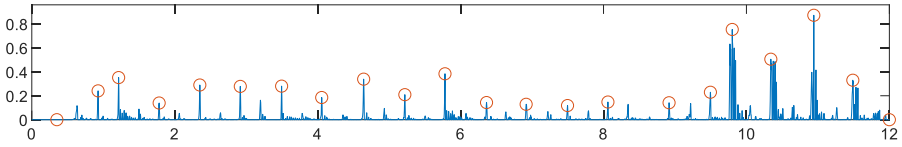
2. 节奏点提取

下面进行提取节奏点。根据日常的经验可以认为，大部分影视音乐的重节奏点之间的间隔大致大于 0.5 秒。因此对半波整流后的波形每隔 0.5 秒取一个最大值作为选点。

观察到对于本音乐来说，观察图谱基本确定为按强、弱、次强、弱的 4/4 拍，节奏点之间间隔大于 0.5 秒，因此每隔 0.5 秒取最大值是合理的。

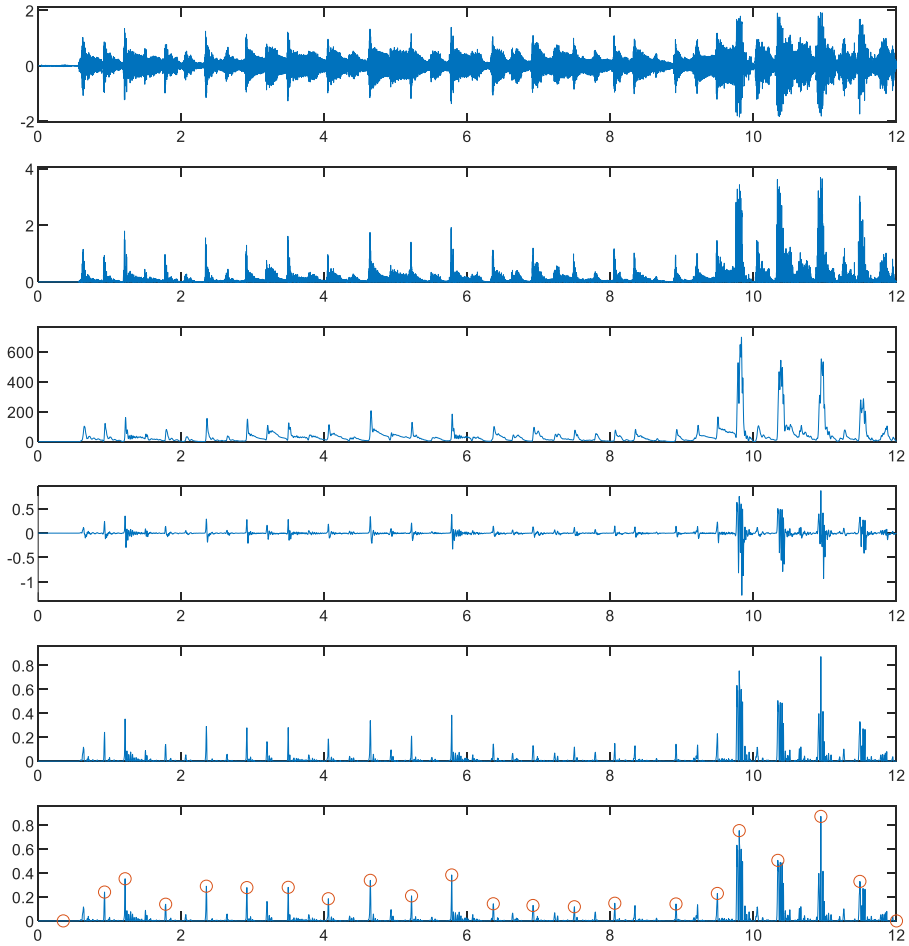
此外，为了防止一个波峰恰好跨越 0.5 秒的分界点导致一个节奏点被选了两次，因此对每两个相邻的选点进行查看，若两个相邻的选点之间的距离小于 0.2 秒，则保留它们中较大的那个而去除较小者。资料显示人耳能分辨的最小声音间隔是 0.1 秒，但是对于较强的节奏点来说间隔一定会比 0.1 秒大，因此选 0.2 秒为筛除时间间隔是合理的。

经过上述筛选，前 12 秒得到的节奏点选点如下：



可以看到确实选取到了大部分的节奏点。

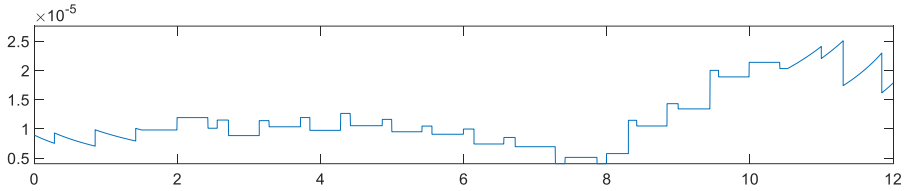
最后，前 12 秒的所有波形绘制结果为（从上到下依次为 x、y_1、y_2、y_3、y_4、节奏点选取）：



（二）度量激烈度

1. “激烈度”序列

为了生成激烈度序列，先将 y_4 进行处理：保留节奏点的函数值，非节奏点的幅度值均置为零，得到节奏点强度序列。对于节奏点强度序列，为了衡量每个时间点附近的“激烈度”，对该时间点周围的强度进行取平均操作，作为该时间点的“激烈度”。因此该操作相当于给节奏点强度序列加上矩形窗，这个操作可以用 MATLAB 的 `movmean` 函数完成。根据经验，计划采用每个点附近 3 秒的强度的均值作为该点的“激烈度”，因此所加矩形窗的长度为 Fs * 3（Fs 为采样率）。得到的“激烈度”序列如下：



2. 视频“激烈度”排序

采用与上述描述相似的方法对 1.mp4~5.mp4 的音频进行处理，得到该五个视频的“激烈度”序列，分别求得“激烈度”序列的 L-2 范数，并对其进行降序排列，结果如下：

```
>> p_mp4(1:5)

ans =

    0.0098    0.0042    0.0103    0.0017    0.0025

>> [~, order_of_first_5] = sort(p_mp4(1:5), 'descend')

order_of_first_5 =

     3     1     2     5     4
```

因此得到“激烈度”排序的顺序由激烈到平缓分别为：3、1、2、5、4。

（三）生成混剪视频

1) 总体思路

通过生活经验可知，要给观看者足够的视觉感受，最好的方案是在背景音乐高潮处使用较为激烈的视频片段，而在背景音乐较为平缓处采用较为平缓的视频片段。在数学上体现为：对于背景音乐“激烈度”序列向量的范数序列与选取的视频“激烈度”序列向量的范数序列做归一化内积时，由排序不等式，顺序和最大，即范数最大的背景音乐片段与范数最大的视频片段相配，此时得到的归一化内积最大。因此，考虑近似算法：

假定所有视频序列的时间长度相差不是很大，对所有视频的时间长度取一个平均值，按照这个平均的时间，将背景音乐划分为时长相等的 N 段：$\{q_1, q_2, q_3, \dots, q_N\}$，然后按照每段的范数从大到小进行排序，得到：$\{q_{S_1}, q_{S_2}, q_{S_3}, \dots, q_{S_N}\}$，然后在视频中选取 N 个视频$\{p_{C_1}, p_{C_2}, p_{C_3}, \dots, p_{C_N}\}$，也按照范数从大到小进行排序，得到：$\{p_{\tilde{C}_1}, p_{\tilde{C}_2}, p_{\tilde{C}_3}, \dots, p_{\tilde{C}_N}\}$，建立一一映射$q_{S_i} \leftrightarrow p_{\tilde{C}_i}$，则将$q_{S_i}$恢复排序前的顺序，得到的$p$的顺序即为拼成混剪视频的视频片段的顺序。上述算法概括为，对于同一种视频选择来说，把视频激烈的片段对应于背景音乐更激烈的片段。

2) 视频片段的选择

接下来的问题是如何进行视频片段的选择，即如何选择这 N 个视频。最初的思路是采取具有一定随机性的方法，用该方法选取 m 次，每次选取 N 个视频，如果视频的时长小于 80 秒则重新选取，然后按照上述算法计算出的顺序拼接视频，再评估每次选取的视频的相关程度，找到相关程度最大的那次作为最终结果。

相关程度的判别方法为：按照本次选出的顺序依次排列各个视频片段$\{p_1, p_2, p_3, \dots, p_N\}$，每一段视频的时间分别记为$\{t_1, t_2, t_3, \dots, t_N\}$，然后根据该时间序列，把背景音乐依次切分成 N 段$\{r_1, r_2, r_3, \dots, r_N\}$，其中每段序列$r_i$的时长为$t_i$。之后再对各段激烈度的范数做归一化内积，得到相关程度：

$$cov = \sum_{i=1}^{n} \frac{p_i r_i}{|p_i|}$$

最后选取 *cov* 的值最大的方案作为最终结果。

3) 随机性方法的制定

最后的问题是“具有一定随机性的方法”如何制定。

a) 方法一——完全随机

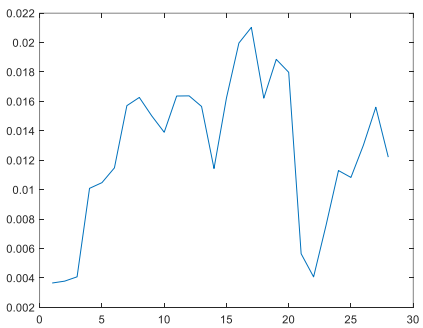
最简单的方法是完全随机，从 53 个视频片段中完全随机选取 N 个视频片段。这种方法最简单直接，但是莽撞性比较大。

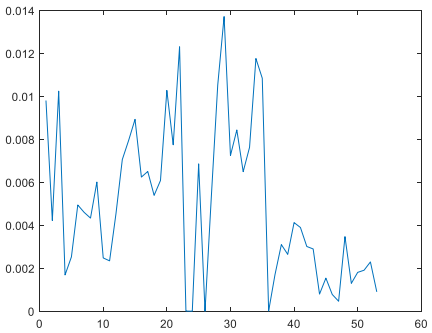
b) 方法二——对随机性加以限制

完全随机的莽撞行比较大，可能较难在少数的尝试中获得较佳的结果。因此考虑对随机性进行限制。

注意到，视频的激烈与否与背景音乐的高潮与否应该是大致匹配的，因此可以在一定范围内随机以增大找到较优解的概率。因此采取的方法是：在把背景音乐的片段$\{q_1, q_2, q_3, \dots, q_N\}$按激烈程度分成三个类别：激烈、中等和平缓（设在三个类别中的片段数分别为 N_1、N_2、N_3，显然有 $N_1+N_2+N_3$=N），同样把视频片段$\{p_{I_1}, p_{I_2}, p_{I_3}, \dots, p_{I_M}\}$（$M$ 为视频片段的总数）也分成激烈、中等、平缓三个类别。这样在进行随机抽取时，在视频片段的三个类别中分别抽取 N_1、N_2、N_3 个视频组成 N 个视频，即可。这样做可能让背景音乐和视频片段的激烈程度进行匹配，可能会较快找到较优解。

用 MATLAB 绘制两个序列的图象如下（左为 q 的图象，右为 p 的图象）：





通过观察可以看出，q 的分界线大概为 0.008 和 0.015，即高于 0.015 的看做是高潮部分，低于 0.008 的看做是平缓部分；而 p 的分界线大概是 0.002 和 0.005。按照该分界线进行定界。

该方法的优点是可以较快得到较优的解，但是也有一些缺点。例如，本方法对选取的范围进行了手动的限制，这可能导致一些更优或最优解没有落在可选的

范围内，因此无法得到这些更优和最优解。

为了让这两种方法互相补充，决定分别使用这两种方法进行若干次随机性选取，并将每次得到的结果汇总，选取相关性最高的一次结果作为输出结果。且由于完全随机的不确定性更大，因此完全随机的尝试次数设置为另一种方法的两倍。

至于选取的时间，考虑到本人电脑的运算速度，经过多次的尝试，尝试 300000 次不会明显增加程序运行的时间，因此完全随机尝试次数取 200000 次，限制随机尝试次数取 100000 次。

最后按照输出顺序拼接视频即可。

四、程序运行方法

使用 MATLAB R2021a 打开 vedio_maker.m 文件，对前两行的 `music_path` 和 `mp4_path` 变量的值进行设置，其中 `music_path` 为背景音乐的**文件路径及文件名**，而 `mp4_path` 为视频片段的**文件路径**，且必须以目录分隔符 `\` 或 `/` 结尾（Windows 下为 `\` ，UNIX 下为 `/`）。然后运行 vedio_maker.m 文件，等待程序运行完毕后，便会生成文件 input.txt，然后利用 ffmpeg 工具将视频和背景音乐按照 input.txt 内的顺序拼接即可。

可以根据自己电脑的计算能力调节参数 `try_time_input`，为进行随机选取视频的总次数。

五、实验结果

本次作业中，程序运行后生成的 input.txt 文件内顺序如下：

52、11、5、38、48、6、30、31、17、16、15、1、13、7、14、34、29、33、35、20、39、10、42、12、8、19、25、27

生成的 input.txt 文件和完成的视频均已放在 result 文件夹中。

六、结果分析

1. 总体评价

从实验结果上看，基本上本次实验实现了影片高潮部分与背景音乐高潮部分的匹配功能，总体上还是比较成功的。

2. 不足与可能的改进方案

从实验的过程上看，出于作者的能力、精力与知识水平所限，本次实验还存在很多设计上和结果上的不足，其中最主要的是程序的普适性和鲁棒性不强，可能很难应对各种各样的视频片段和背景音乐。主要的问题与可能的未来改进方案如下：

1) 参数调节未实现为自动化。

本次实验中，仍然存在着需要手工调节的参数：在限制随机的算法中，需要根据中间结果绘制的 p 序列与 q 序列人工为不同类别进行定界。这可能存在着大量的主观因素，并且也增大了程序使用的难度。

一个可能的改进方案是，设计算法使得程序能够自动为它们进行分类。在这里，一个可能的建议是引入机器学习进行多分类任务。一旦实现了这个功能，本程序可以适用于更加广泛的视频片段和背景音乐。

2) 对音乐节奏点的分析未实现为智能化

在本次实验的中间过程中，我们根据生活经验，选取了 0.5 秒、0.2 秒等参数作为对节奏点的间隔的估计。但是，输入的音乐与视频很可能与这个估计值有较大的偏差，尤其是视频的音频可能会存在很多环境噪声，并且不像普通音乐那样均匀规律，因而导致遗漏某些 节奏点或提取了多余的节奏点，等等。

一个可能的解决方案是设计新的算法来自动检测各个波形的局部极大值点而非最大值点来作为节奏点，这样可以很好地避免上述问题，增加程序的普适性。

3) 随机算法的缺陷

本次实验所选取的算法偶然性太大。虽然引入了限制随机性的算法对全随机进行了一定程度的限制，但是限制的效果可能并不尽人意，甚至存在过限制的倾向。

为了探究限制算法与全随机算法的效果好坏，我设计了如下实验：

分别让全随机和限制随机的算法随机选取 *m* 次，分别取最好结果，并将两个最好结果的相关度 *cov* 进行比较。重复这个实验 10 轮，观察结果好坏。

a) 当 *m*=300 时，在 10 轮重复实验中，每轮全随机的最好结果基本介于 0.0015 与 0.0023 之间，每一轮实验之间差别较大，并且大部分位于 0.0015~0.0022 之间；而对于限制随机的算法，每轮的最好结果介于 0.0019~0.0021 之间。因此可以看到，限制随机的算法在这个条件下确实可以使结果变得更加稳定，并且平均效果好于全随机

b) 当 *m*=300000 时，在 10 轮重复实验中，每轮全随机的最好结果基本介于 0.0038~0.0047 之间，而每轮限制随机算法的最好结果基本介于 0.0029~0.0032 之间。由此可见，当 *m* 值较大时，限制随机的算法确实把很多较优解排除在了选择之外，存在过限制现象。

而本次实验中，全随机进行 200000 次，限制随机进行 100000 次，推测限制随机并无法起到太大的作用，因此对随机性的限制需要进行改进。

一个可能的实现方案是在进行分界时，不确定明确的界限，而是采取非均匀随机的方案：高潮部分的背景音乐也有几率匹配到中等或平缓的视频片段，但是概率比匹配到高潮视频片段的要小，并且视频片段越平缓，概率越小；对于中等或平缓的背景音乐进行相同的处理。具体新方案能否得到预期效果还需要进一步探究。

4) 对双声道的处理

本程序未考虑双声道的问题。在人的左右耳接收到不同的声音时，对人脑产生的刺激是非常复杂的。本次实验尚未对双声道对人脑冲击力的影响进行建模，这需要未来进行更多的科学调研来解决。

七、致谢声明

在本次实验中，老师和助教提供的作业说明对我实验的完成有莫大的帮助，再次对他们的辛勤付出表示由衷的感谢。此外，在课下我也和许多同学就本次作业进行了一些思路方面的讨论，互相交流一些自己的见解和看法，也对我完成本次实验有一定的促进作用，在此也一并表示感谢。