



数字信号处理

专题实践 钢琴音频识别(1)

华东理工大学 信息科学与工程学院 万永菁





1、实验目的

通过钢琴乐音识别技术的研究，掌握满足**复杂工程问题需求**的离散时间系统的基本设计方法与**分析技术**，了解影响设计目标和技术方案的各种因素，并得出有效结论。

2、实验内容

钢琴乐音识别技术对钢琴乐音信号进行基频估计，然后根据基频大小来区分音高，从而实现对乐曲的识别。

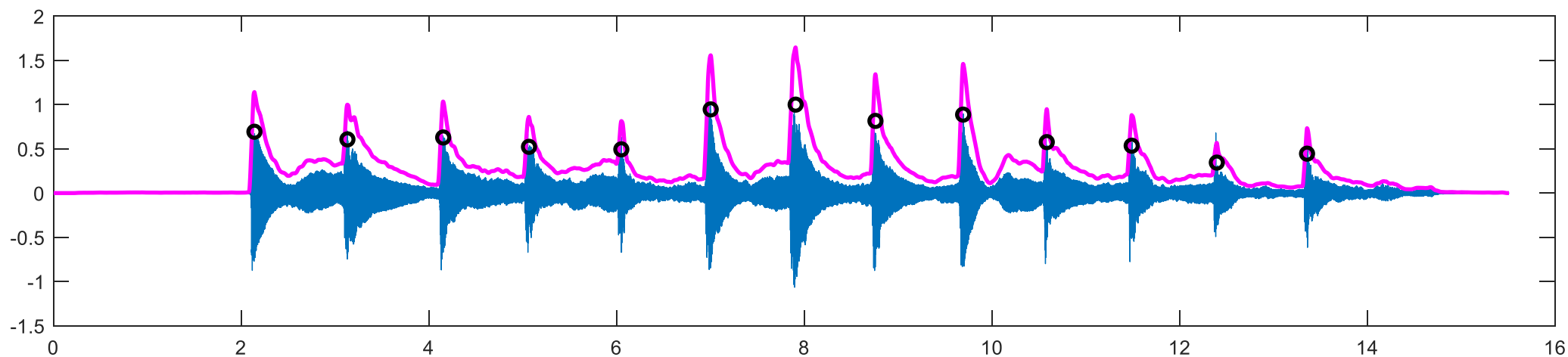
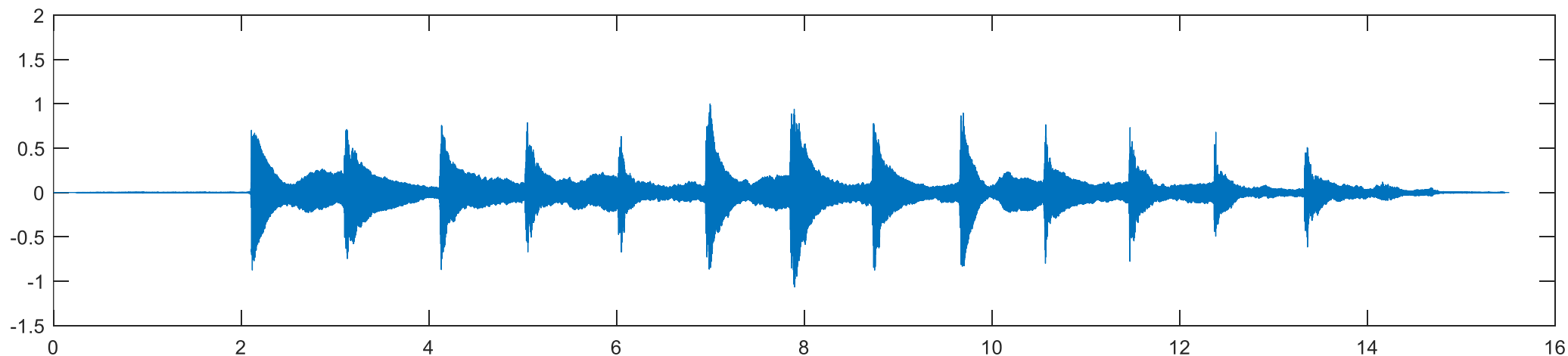
课题针对钢琴音频信号进行乐音识别技术的研究。在对音频信号其进行**分帧、分音程检测**后，对各**音程段信号进行离散傅里叶变换**，识别频谱中所蕴含的**音符信息**，并**与乐谱进行比对，并得出结论**。

请查阅提供的文献资料，并结合自己查阅的资料，完成以下实验内容：

- (1) **基础要求（必做）：实现不同组别的钢琴音阶识别；**
- (2) **进阶要求（选做）：实现钢琴乐曲《小星星》的整曲识别。**
- (3) **拓展要求（选做）：对钢琴乐曲《小星星》的整曲节奏进行分析评价。**



音符切分

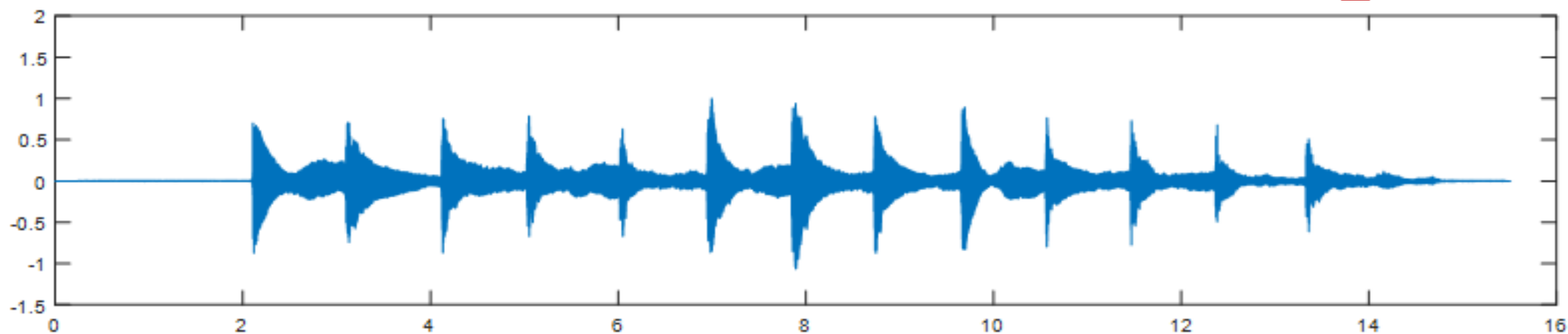




Step1. 读入音频文件

使用audioread函数，注意提取单声道信号，归一化，存Fs参数，画图。

音阶\M1_i1.wav



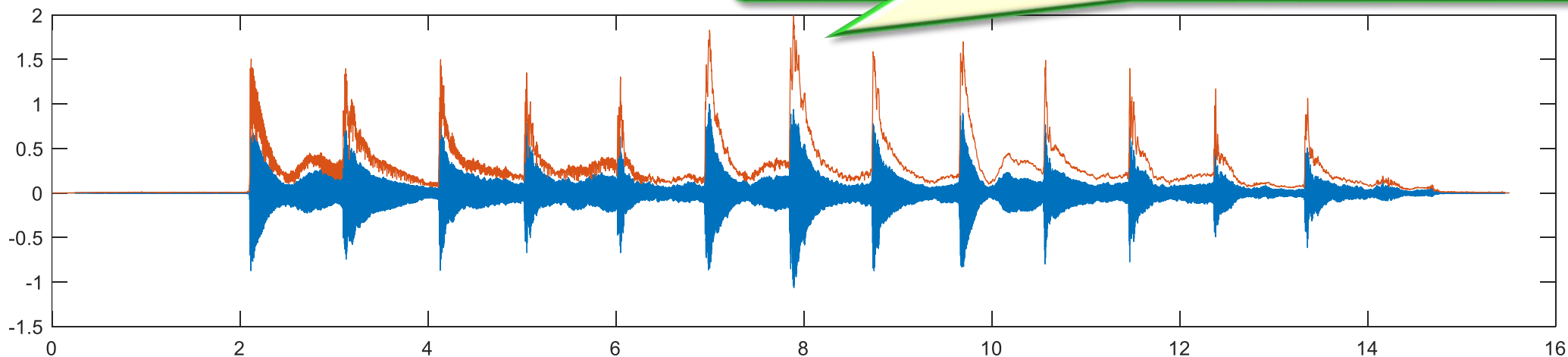
信号存在music变量中。



Step2. 使用帧峰检测法提取音频的包络信号

- (1) 对音频信号进行分帧，帧长设为`Frame_Num` (此参数请研究，比如取100)；
- (2) 记录每一帧内的峰峰值，即“最大值-最小值”；
- (3) 将每一帧的峰峰值记录在包络数组`envelope1`中；
- (4) 在原音频图上画出包络线，用`hold on`。
- (5) 注意包络信号的横坐标与原始信号的横坐标的间隔不一样。

包络信号噪声较大，后续寻找音符起点困难





```
[x,Fs]=audioread('D:\课程\3DSP\讲义\2023大混合讲义\实验2023\...  
专题实验资料-学生\音阶\M1_i1.wav' );
```

```
music = x(:,1)./max(x(:,1)); %归一化  
Frame_Num = 100; %帧长参数如何选择请自行研究对比
```

% ① 使用帧峰检测法提取音频的包络

```
envelope1=[];  
for i = 1 : floor(length(music)/ Frame_Num )-1 % i是以Frame_Num个点分帧后的帧号  
    %每Frame_Num点是一个临时帧  
    temp = music( (i-1)* Frame_Num +1 : i* Frame_Num );  
    %取出当前帧的峰峰值作为该帧的帧峰值  
    %并保存在envelope包络中  
    envelope1=[envelope1 max(temp)-min(temp)];  
end
```

如何将音频信号和envelope1包络信号
画在一个图上请自行研究



Step3. 尝试使用滤波算法，对包络信号进行滤波

Step3(法1). 中值滤波

方法1：使用 **medfilt1()** 函数，查阅帮助了解函数使用方法；

方法2： **自己编写** 中值滤波函数（请尝试）。

Step3(法2). 均值滤波

方法1：可使用 **smooth()** 函数，查阅帮助了解函数使用方法；

方法2： **自己编写** 均值滤波函数（请尝试）。

注：包络滤波后，可以做一次归一化，以便后续设定音符起点的阈值

3、尝试使用滤波算法，对包络信号进行滤波

请自行调整参数，
记录实验结果，并分析

envelope1

帧峰包络

envelope2

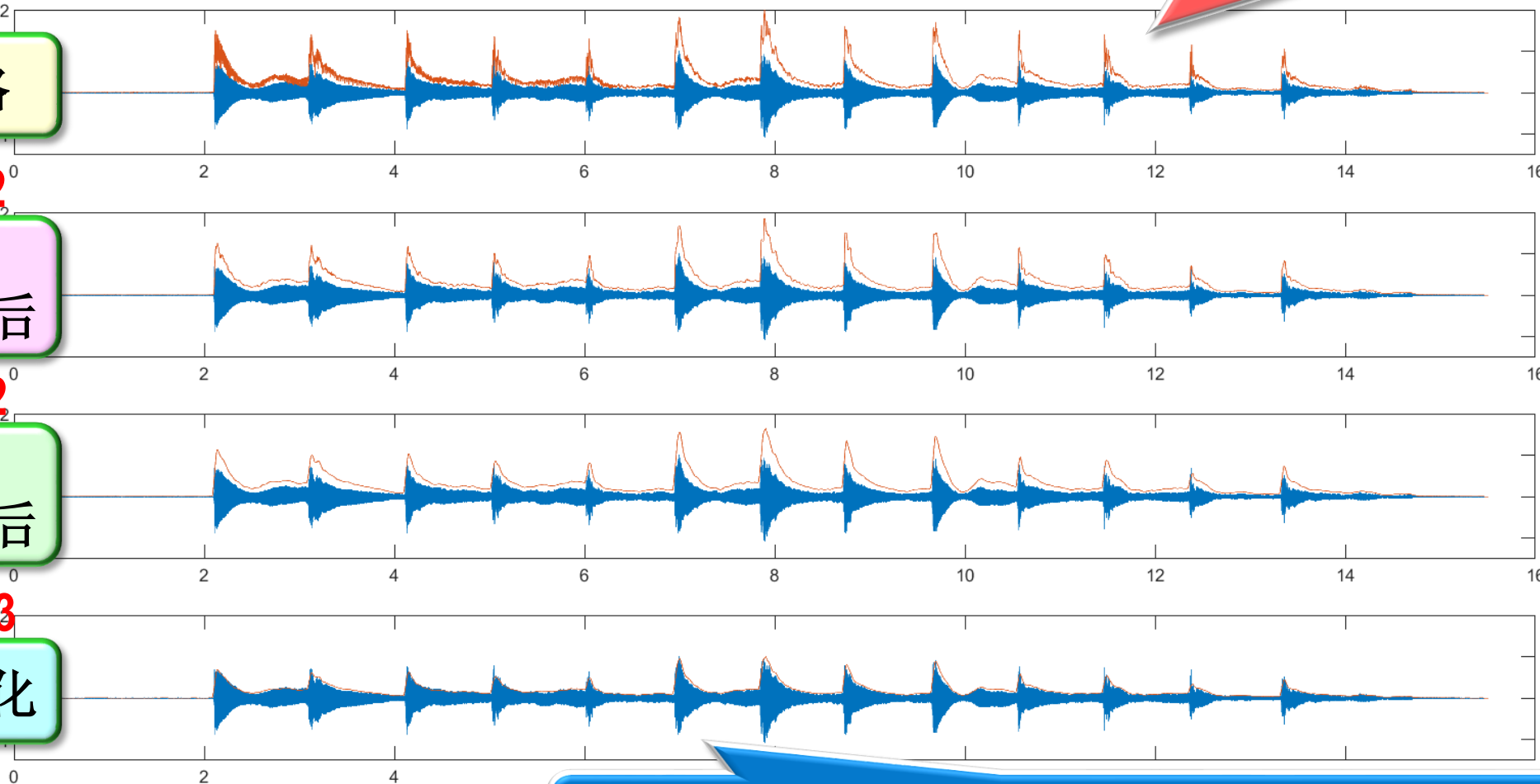
11点
中值滤波后

envelope2

21点
均值滤波后

envelope3

包络归一化

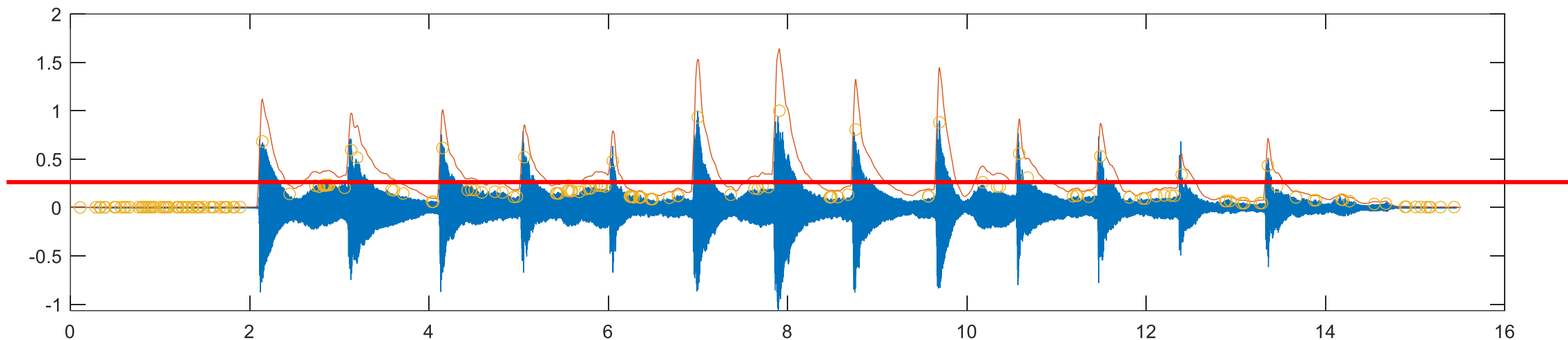


可对帧峰包络进行多次滤波，注意包络最后的归一化，
假设滤波后的归一化包络存在envelope3中；



Step4. 根据包络信号寻找音符的起始点

Step4-1. 寻找包络信号的峰值，可以用`findpeaks()`函数

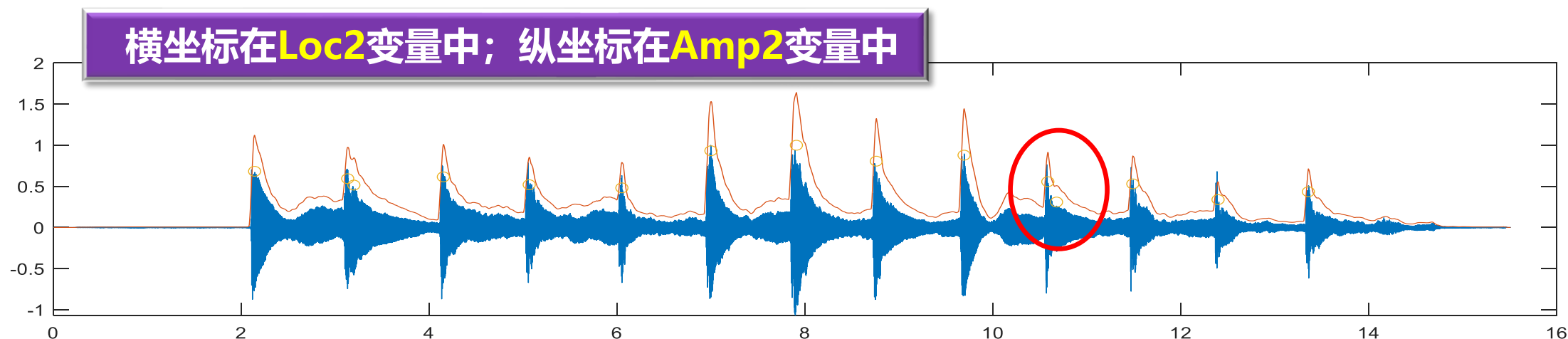
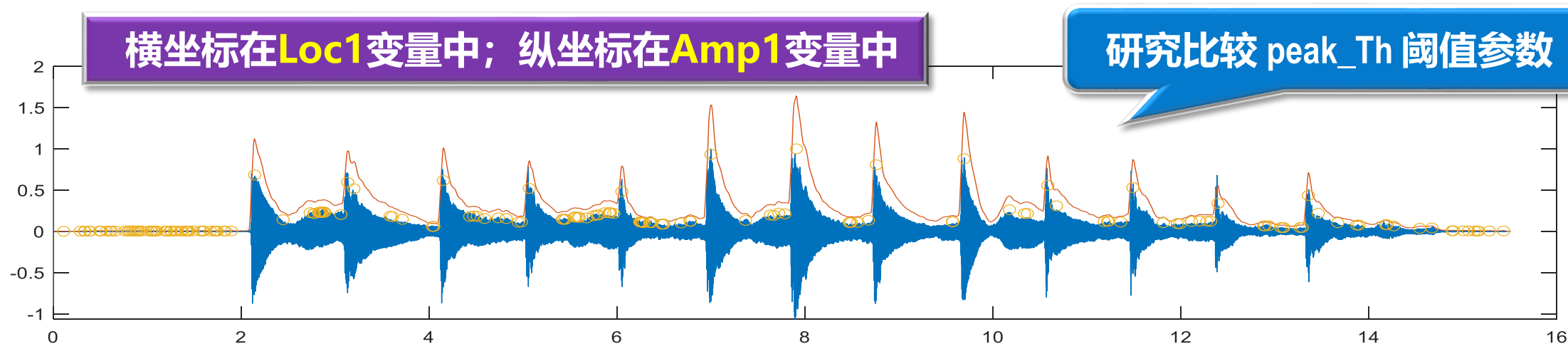


Step4-2. 可以设定一个阈值，去除较小的伪峰点（伪音符起始点）

注：阈值选取过大则会漏掉起始点；过小则后面去伪峰点数目增多

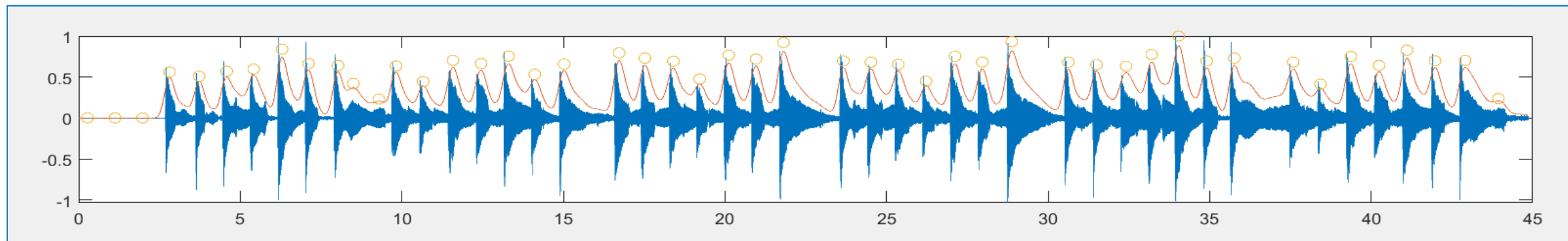
Step4. 根据包络信号寻找音符的起始点

- 设阈值 peak_Th 取0.3，小于 peak_Th 的峰直接去除

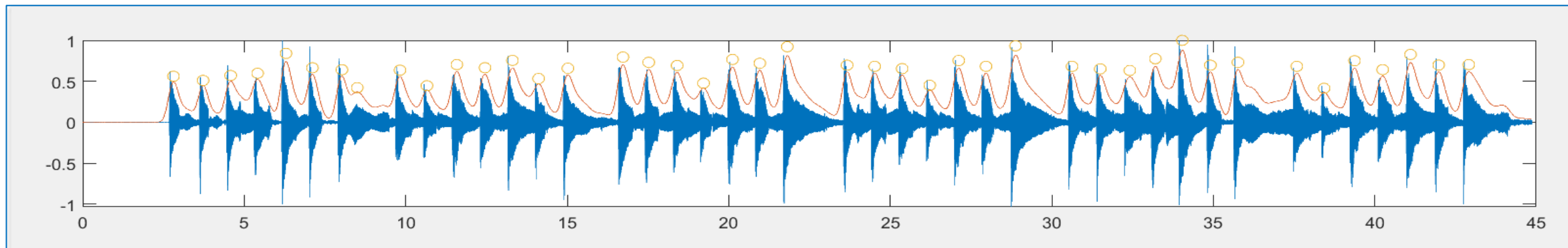


Step4-1. 用findpeaks函数寻找包络峰值;

横坐标在Loc1变量中; 纵坐标在Amp1变量中



Step4-2. 峰的幅度限制: 小于peak_Th(比如设0.3)的峰直接去除, peak_Th参数请研究;



横坐标在Loc2变量中; 纵坐标在Amp2变量中



Step4. 根据包络信号寻找音符的起始点

Step4-3. 去冗余峰值点

思路：若峰点距离很近，则判为伪峰。

最小峰间隔为：最小节奏时长所对应的帧间隔点数。

$$\text{峰间隔}_{\min} = \frac{[\text{每拍时长}(s)][\text{最小拍数}]}{[\text{采样间隔}T][\text{帧长}]} = \frac{\left[\frac{60(\text{秒})}{\text{每分钟节拍数}}\right]\left[\frac{1}{\text{最大分拍数}}\right]}{[\text{采样间隔}T][\text{帧长}]}$$

举例：若 $\text{J}=100$ ， $\text{fs}=44100$ ，最快 $1/4$ 拍，帧长=100

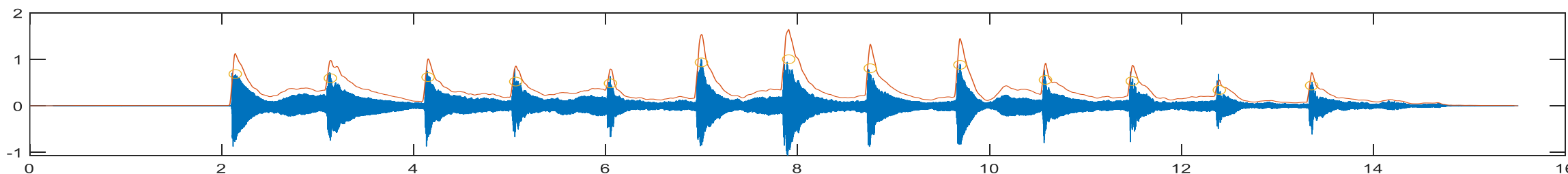
$$\text{峰间隔}_{\min} = \frac{60(\text{秒})}{100} \cdot \frac{1}{4} \cdot 44100 \cdot \frac{1}{100} = 66.15 \xrightarrow[\text{floor}]{\text{取整}} 66$$

Step4. 根据包络信号寻找音符的起始点

- 编写去冗余峰值点的程序，仅记录间隔超过最小峰间隔的点

$$\text{峰间隔}_{\min} = \frac{[\text{每拍时长(s)}][\text{最小拍数}]}{[\text{采样间隔}T][\text{帧长}]} = \frac{\left[\frac{60(\text{秒})}{\text{每分钟节拍数}}\right]\left[\frac{1}{\text{最大分拍数}}\right]}{[\text{采样间隔}T][\text{帧长}]}$$

```
Beat_Min=100; Min_sub_beat=4; Frame_Num=100;
Min_Gap = floor((60/Beat_Min)*Fs/Min_sub_beat/Frame_Num);
```



横坐标在Loc3变量中，纵坐标在Amp3变量中