AMATH582 Hw2: Classic Rock and Roll Shredding

Xiangyu Gao

Feb 10, 2021

Abstract

In this report, we analyzed a portion of two of the greatest rock and roll songs by showing their time-frequency spectrum. With the spectrum, we tried to reconstruct the music score of particular instrument (guitar or bass) from each song using the band-pass filter on frequency domain.

1 Introduction and Overview

Two of the greatest rock and roll songs are analyzed in this report: Sweet Child Of Mine by Guns N' Roses, and Comfortably Numb by Pink Floyd. Guitar World ranked the Sweet Child Of Mine No. 37 all time, and ranked the Comfortably Numb No. 4 all time.

We use a portion of these two songs and name them as GNR.m4a (14 seconds clip) and Floyd.m4a (60 seconds) and convert them as vectors representing the music, thus we can easily edit the music by modifying the vector via MATLAB.

We perform three tasks in this report in total. First, through the use of the Gabor filtering, we reproduce the music score for the guitar in the GNR clip, and the bass in the Floyd clip. Second, we use a band-pass filter in frequency space to try to isolate the bass in Comfortably Numb clip. Third, we put together as much as possible the guitar solo in Comfortably Numb clip by looking at the spectrum of smaller portions of the clip.

2 Theoretical Background

The Gabor transform, also known as the short-time Fourier transform ($STFT$) is defined as the following:

$$\mathcal{G}[f](t,\omega) = \tilde{f}_g(t,\omega) = \int_{-\infty}^{\infty} f(\tau)\bar{g}(\tau - t)e^{-i\omega\tau}d\tau = (f, \bar{g}_{t,\omega})$$

where the bar denotes the complex conjugate of the function, the function $g(\tau - t)$ acts as a time filter for localizing the signal over a specific window of time, $g_{t,\omega}(\tau) = e^{i\omega\tau}g(\tau - t)$ is the Gabor kernel [1].

3 Algorithm Implementation and Development

We introduce the algorithm implementation details below and attach corresponding MATLAB code in Appendix C.

To reproduce the music score for guitar in the GNR clip, we first read the music data as a vector and transform the time-sequence vector into the time-frequency spectrum using Gabor filtering and FFT. The window interval for Gabor filtering is 0.5, and the attenuation is 3000. We compare the obtained spectrum (that is expressed as the $\log(\|s\| + 1)$ format) with the original music score and determine the frequency range of guitar in the spectrum. The overtones (out of the selected frequency band) are filtered out using the band-pass filter (set value zero for overtones).

To reproduce the music score for bass in the Floyd clip, we perform the same operations as above. Particularly, the bass is isolated by filtering out all overtones using band-pass filter.

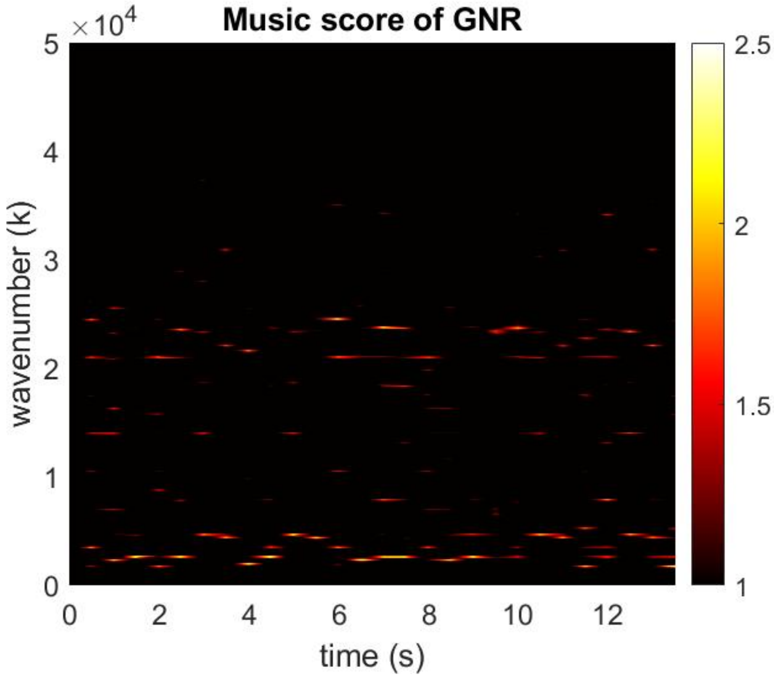

Figure 1: The spectrum of GNR clip.

To put together all guitar solo in Comfortably Numb, we need to filter out the bass and overtones from corresponding spectrum. To guide the reconstruction of guitar music score and reduce computation load, we cut the Floyd clip into 4 pieces and analyze each piece carefully.

4 Computational Results

We show the spectrum of GNR clip in Fig. 1 that only presents wave number (k) ranging from 0 to 5×10^4. The isolated spectrum of guitar (wave number k ranging from 1.5×10^4 to 3×10^4) in GNR clip is shown in Fig. 2. Compared to the original music score presented in Fig. 3, the obtained spectrum has similar nodes. I didn't label the nodes on spectrum since it is hard for me to read and understand the music score.

We show the spectrum of Floyd clip in Fig. 4 that only presents wave number (k) ranging from 0 to 3×10^4. The isolated spectrum of bass (wave number k ranging from 0.3×10^4 to 0.6×10^4) in Floyd clip is shown in Fig. 5. From this figure, we can see that the nodes of bass is covered by some sounds with similar wave number (frequency).

To extract the spectrum of guitar solo from Fig. 4, we use a band-pass filter to filter out the region with wave number ranging from 2×10^4 to 3×10^4 and show the guitar solo spectrum in Fig. 6, Fig. 7, Fig. 8, and Fig. 9. From them, we can clearly identify the nodes of guitar music score.

5 Summary and Conclusions

In this report, we analyzed a portion of two of the greatest rock and roll songs by showing their time-frequency spectrum. With the spectrum, we reconstructed the music score of particular instrument (guitar or bass) from each song by using the band-pass filter on frequency domain, and compared it with the original music score.

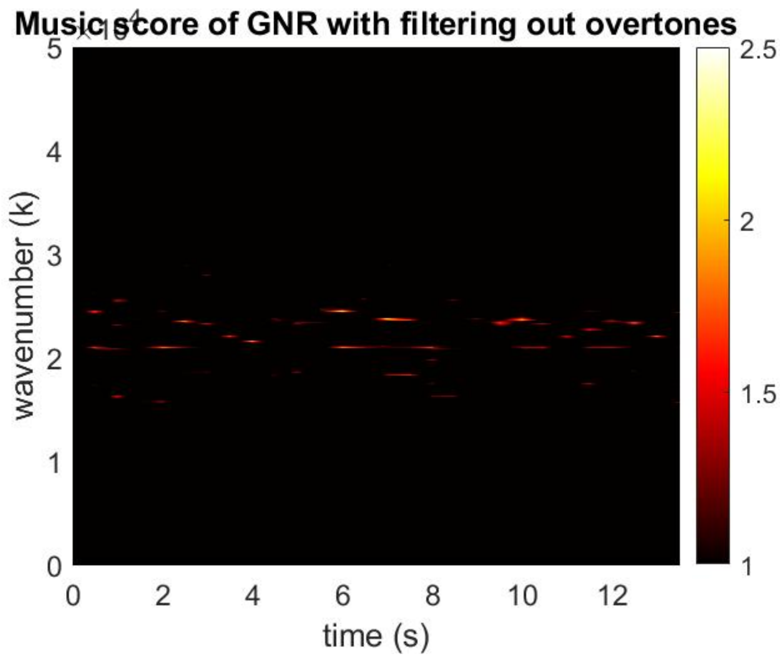

Figure 2: The spectrum of guitar in GNR clip, which is obtained by filtering out overtones using band-pass filter.

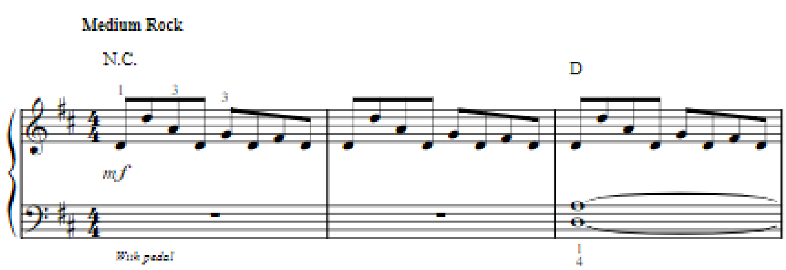

Figure 3: The music score of GNR clip.

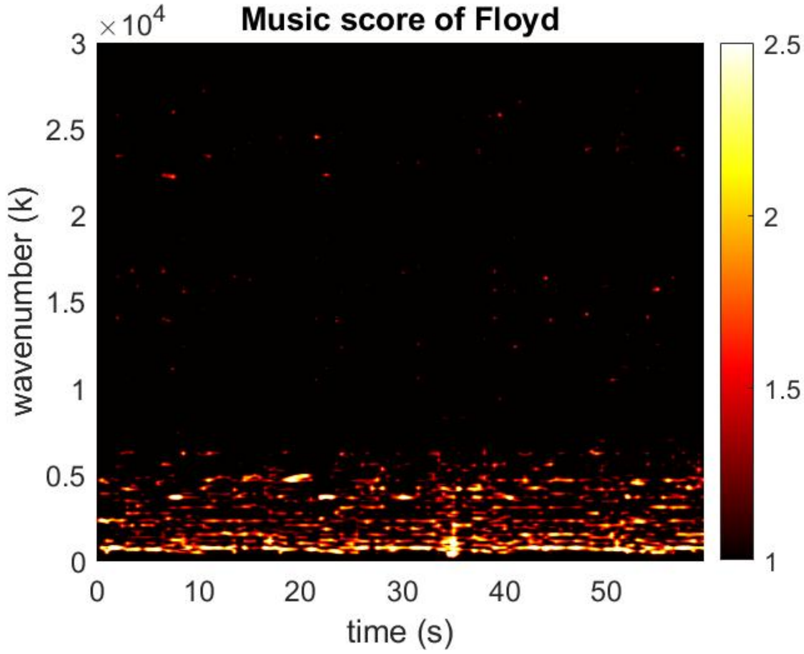

Figure 4: The spectrum of Floyd clip.

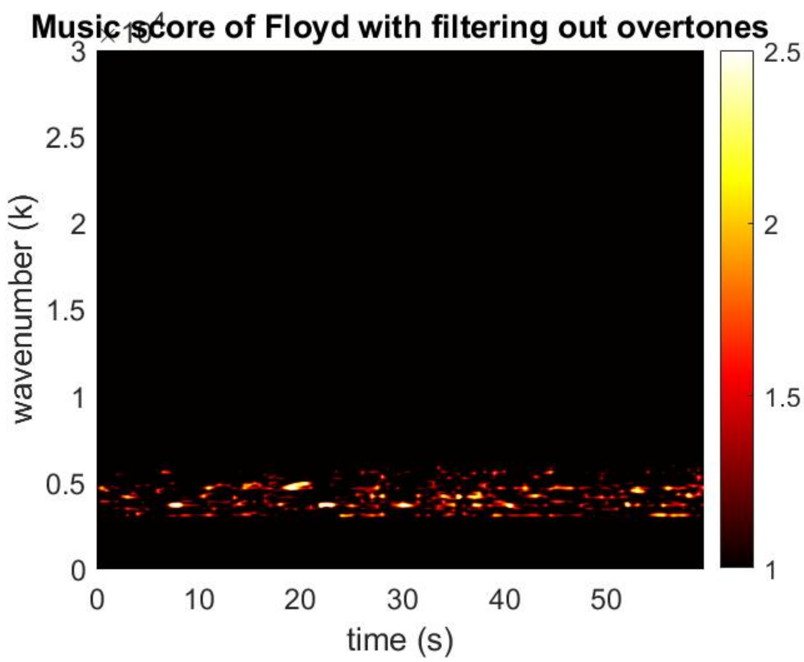

Figure 5: The spectrum of bass in Floyd clip, which is obtained by filtering out overtones using band-pass filter.

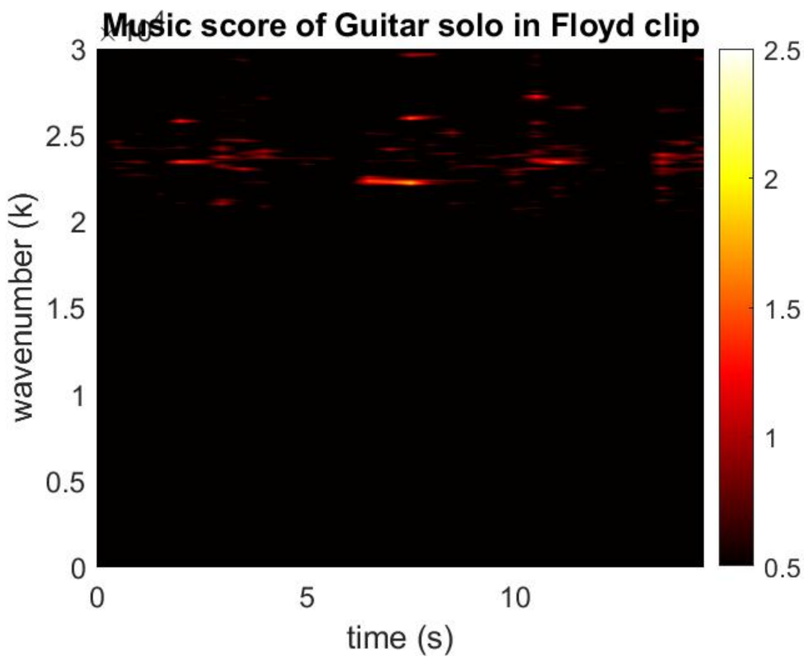

Figure 6: The spectrum of guitar in the first 15 seconds of Floyd clip, which is obtained by filtering out bass and overtones using band-pass filter.

References

[1] Jose Nathan Kutz. *Data-driven modeling & scientific computation: methods for complex systems & big data*. Oxford University Press, 2013.

Appendix A MATLAB Functions

The important MATLAB functions and thier implementation explanations are listed here:

- `y = fft(X,N,DIM)` applies the fft operation across the dimension DIM.
- `[Y, FS]=audioread(FILENAME)` reads an audio file specified by the character vector or string scalar FILENAME, returning the sampled data in Y and the sample rate FS, in Hertz.
- `pcolor(X,Y,C)` where X and Y are vectors or matrices, makes a pseudocolor plot on the grid defined by X and Y. X and Y could define the grid for a "disk", for example.

Appendix B Figures

We show the left spectrum images of guitar solo of Floyd in Fig. 7, 8, and 9.

Appendix C MATLAB Code

The MATLAB codes for Problem 1-3 are shown here. We also stored corresponding codes on Github under the repository (https://github.com/Xiangyu-Gao/AMATH-582-Computational-method-for-data-analysis/tree/main/Hw2).

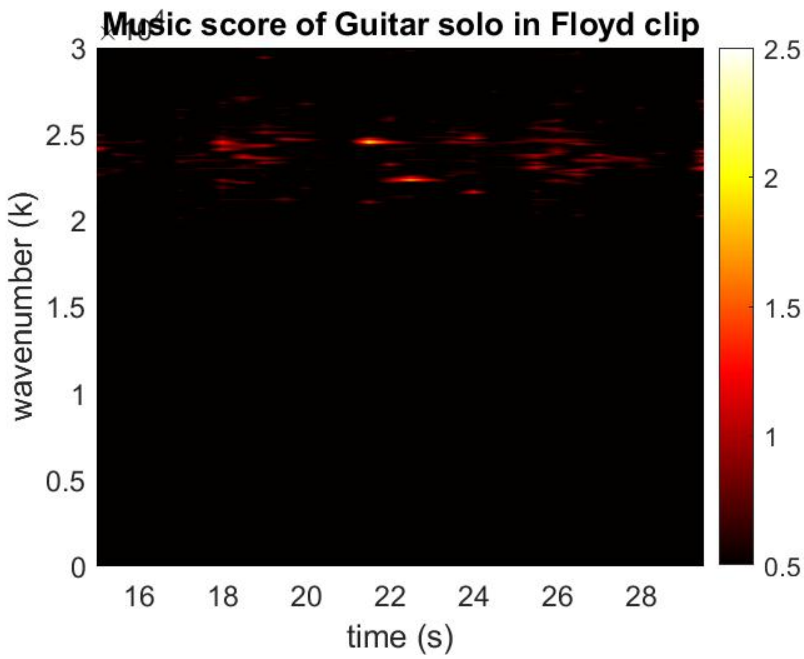

Figure 7: The spectrum of guitar in the second 15 seconds of Floyd clip, which is obtained by filtering out bass and overtones using band-pass filter.

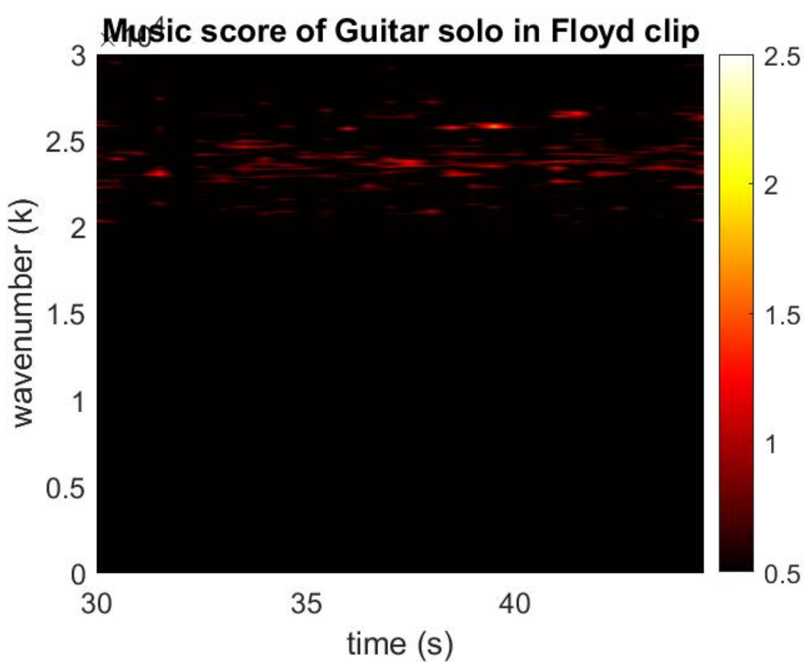

Figure 8: The spectrum of guitar in the third 15 seconds of Floyd clip, which is obtained by filtering out bass and overtones using band-pass filter.

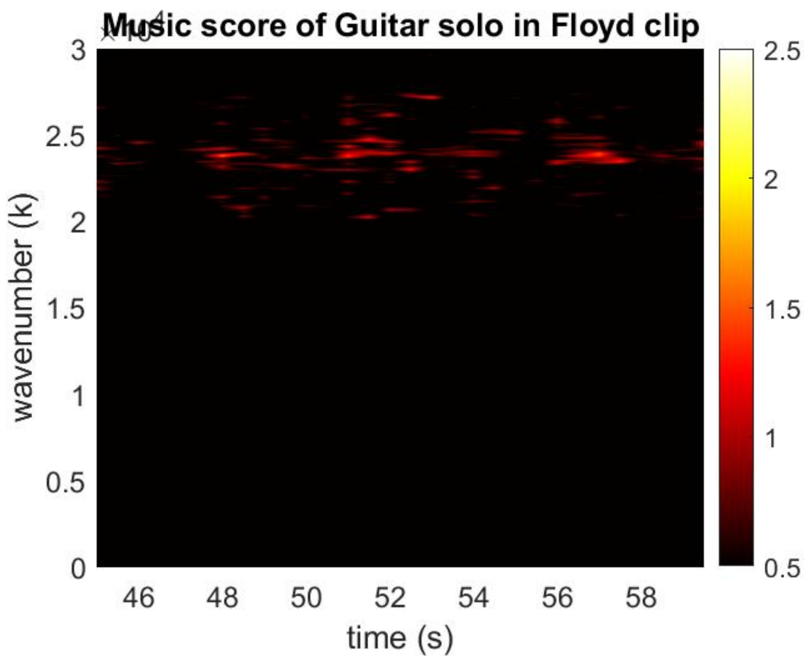

Figure 9: The spectrum of guitar in the last 15 seconds of Floyd clip, which is obtained by filtering out bass and overtones using band-pass filter.

```
clc
clear all
close all

%% Read input data
[y, Fs] = audioread('C:\Users\Xiangyu Gao\Desktop\AMATH\AMATH 582\hw2\GNR.m4a');
tr_gnr = length(y)/Fs; % record time in seconds
figure(1)
plot((1:length(y))/Fs,y);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Sweet Child O Mine');
% p8 = audioplayer(y,Fs); playblocking(p8);

%% Parameter CONSTRUCTION
n = length(y);
L = n/Fs; % second
t2 = linspace(0,L,n+1);
t = t2(1:n);
k = (2*pi/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);

%% SLIDING GABOR WINDOW
Sgt_spec = [];
tslide = 0:0.5:L;

for j = 1:length(tslide)
    g = exp(-3000*(t-tslide(j)).^2); % Gabor
    Sg = g .* y';
    Sgt = fft(Sg);
    Sgt_spec = [Sgt_spec; log(abs(fftshift(Sgt))+1)];
end

figure(2)
pcolor(tslide, ks, Sgt_spec.'),
shading interp
set(gca,'Ylim',[0, 5e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([1.0 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of GNR')

%% filter out overtones and Isolate Guitar
filtered_range = [1.5e4, 3e4];
filter = zeros(size(ks));
[~, id1] = min(abs(ks-filtered_range(1)));
[~, id2] = min(abs(ks-filtered_range(2)));
filter(id1:id2) = 1;

figure(3)
Sgt_spec_filter = Sgt_spec .* filter;
pcolor(tslide, ks, Sgt_spec_filter.'),
shading interp
set(gca,'Ylim',[0, 5e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([1.0 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of GNR with filtering out overtones')
```

```
clc
clear all
close all

%% Read input data
[y, Fs] = audioread('C:\Users\Xiangyu Gao\Desktop\AMATH\AMATH 582\hw2\Floyd.m4a');
tr_gnr = length(y)/Fs; % record time in seconds
figure(1)
plot((1:length(y))/Fs,y);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Comfortably Numb');
% p8 = audioplayer(y,Fs); playblocking(p8);

%% Parameter CONSTRUCTION
y = y(1:length(y)-1);
n = length(y);
L = n/Fs; % second
t2 = linspace(0,L,n+1);
t = t2(1:n);
k = (2*pi/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);

%% SLIDING GABOR WINDOW
Sgt_spec = [];
tslide = 0:0.5:L;

for j = 1:length(tslide)
    g = exp(-3000*(t-tslide(j)).^2); % Gabor
    Sg = g .* y';
    Sgt = fft(Sg);
    Sgt_spec = [Sgt_spec; log(abs(fftshift(Sgt))+1)];
end

% figure(2)
% pcolor(tslide, ks, Sgt_spec.'),
% shading interp
% set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
% colormap(hot)
% colorbar
% caxis([1.0 2.5])
% xlabel('time (s)')
% ylabel('wavenumber (k)')
% title('Music score of Floyd')

%% Isolate Bass
filtered_range = [0, 0.6e4];
filter = zeros(size(ks));
[~, id1] = min(abs(ks-filtered_range(1)));
[~, id2] = min(abs(ks-filtered_range(2)));
filter(id1:id2) = 1;

% figure(3)
% Sgt_spec_filter = Sgt_spec .* filter;
% pcolor(tslide, ks, Sgt_spec_filter.'),
% shading interp
% set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
% colormap(hot)
% colorbar
% caxis([1.0 2.5])
% xlabel('time (s)')
% ylabel('wavenumber (k)')
```

```matlab
%% Obtain Guitar
filtered_range = [2e4, 3e4];
filter = zeros(size(ks));
[~, id1] = min(abs(ks-filtered_range(1)));
[~, id2] = min(abs(ks-filtered_range(2)));
filter(id1:id2) = 1;

figure(4)
Sgt_spec_filter = Sgt_spec .* filter;
pcolor(tslide(1:30), ks, Sgt_spec_filter(1:30,:).')
shading interp
set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([0.5 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of Guitar solo in Floyd clip')

figure(5)
Sgt_spec_filter = Sgt_spec .* filter;
pcolor(tslide(31:60), ks, Sgt_spec_filter(31:60,:).')
shading interp
set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([0.5 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of Guitar solo in Floyd clip')

figure(6)
Sgt_spec_filter = Sgt_spec .* filter;
pcolor(tslide(61:90), ks, Sgt_spec_filter(61:90,:).')
shading interp
set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([0.5 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of Guitar solo in Floyd clip')

figure(7)
Sgt_spec_filter = Sgt_spec .* filter;
pcolor(tslide(91:120), ks, Sgt_spec_filter(91:120,:).')
shading interp
set(gca,'Ylim',[0, 3e4],'Fontsize',[14])
colormap(hot)
colorbar
caxis([0.5 2.5])
xlabel('time (s)')
ylabel('wavenumber (k)')
title('Music score of Guitar solo in Floyd clip')
```

Listing 3: MATLAB code for Problem 3.