Amath 482 Homework 2: Rock & Roll and the Gabor Transformation

Yilin Cao

January 27, 2021

Abstract

The purpose of this report is to make time-frequency analysis on a portion of two of the greatest rock and roll songs, which are the songs *Sweet Child O' Mine* by Guns N' Roses and *Comfortably Numb* by Pink Floyd. We will apply Gabor transformation, Gaussian filter, Shannon filter, along with the use of spectrogram to complete the analysis.

I. Introduction and Overview

In this assignment, we are given two audio files, *GNR.m4a* and *Floyd.m4a,* which contain two greatest rock and roll music pieces for us to make further analysis. To begin with, we will import the music pieces into Matlab and make a plot of sound amplitude on the time axis. Then we will perform time-frequency analysis in the following steps.

In the first step, we will apply Gabor filtering to reproduce the music score for the guitar in the GNR clip, and the bass in the Floyd clip. Next, we will come up with a new filter to the frequency space of Floyd clip in order to isolate the bass portion. After filtering out the bass, we will be able to reconstruct the music notes for guitar solo in Comfortably Numb. I will present these steps in the following sections.

II. Theoretical Background

1. Gabor Transformation

Gabor transformation, also known as the Short-time Fourier transform, takes the singal function f(t) multiplying by a window function g(t) to localize in time with the Fourier Transform. It is given precisely by

$$\bar{f}_g(\tau, k) = \int_{-\infty}^{\infty} f(t) g(t-\tau) e^{-ikt} dt$$

Sweep the value of tau over the entire signal

The larger the a, the smaller the window.
Localize in time, but spread out in frequency space

2. Spectrogram

Spectrogram allows to visualize the movement video in 2-dimensional medium. A spectrogram will review the spectral of the signal in a plot where the horizontal direction is the window centre τ and the vertical direction is the frequency domain k.

III. Algorithm Implementation and Development

Part 1.

- Read in the audio files and take their DFTs
- Create Gabor window and use Gaussian filter to make Gabor transformation
- Generate spectrogram on time axis with increment in dt
- Labels the music scores on y-axis

Part 2.

- Read in Floyd audio clips and take its DFT.
- Create Shannon filter and apply to the frequency space
- Plot the graph of unfiltered frequency in audio data with the filter
- Plot the graph of frequency space after applying Shannon filter

Part 3.

- Generate spectrogram of filtered Floyd audio clip

IV. Computational Results

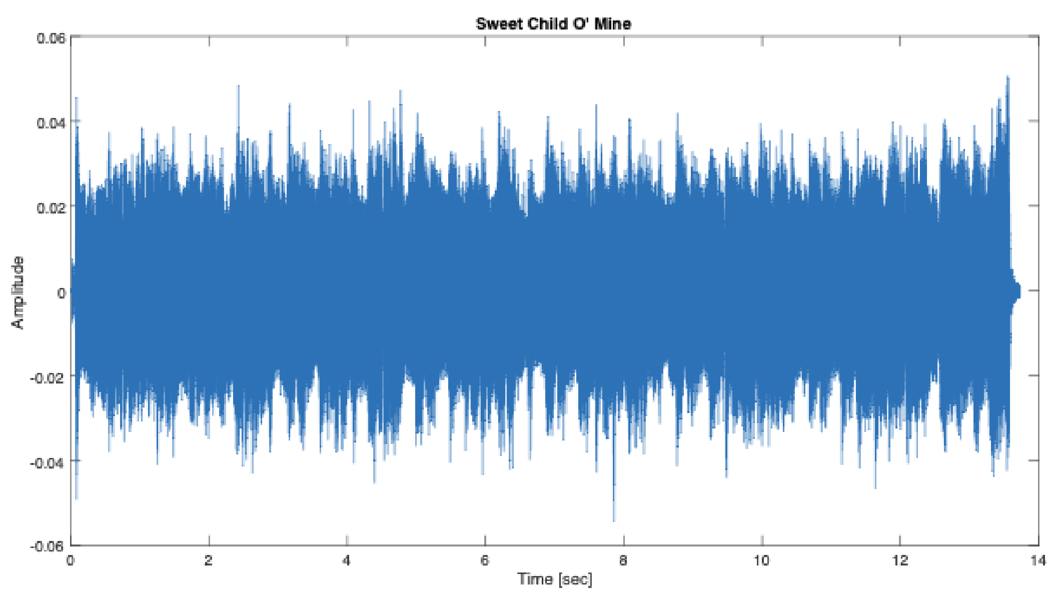

Figure 1. Signals of GNR clip

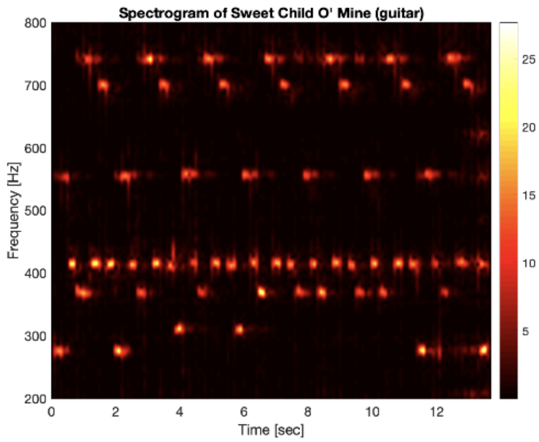

Figure 2. Spectrogram for the guitar in GNR clip

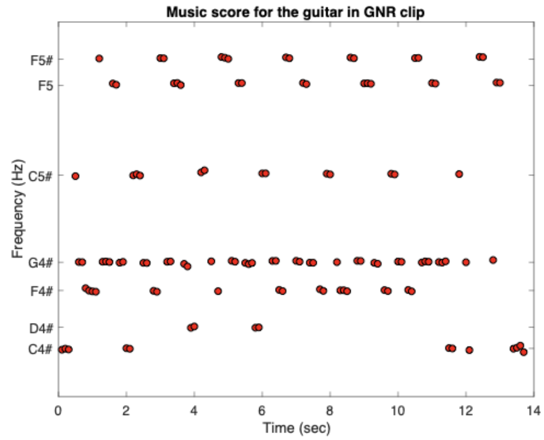

Figure 3. Music score for the guitar in GNR clip

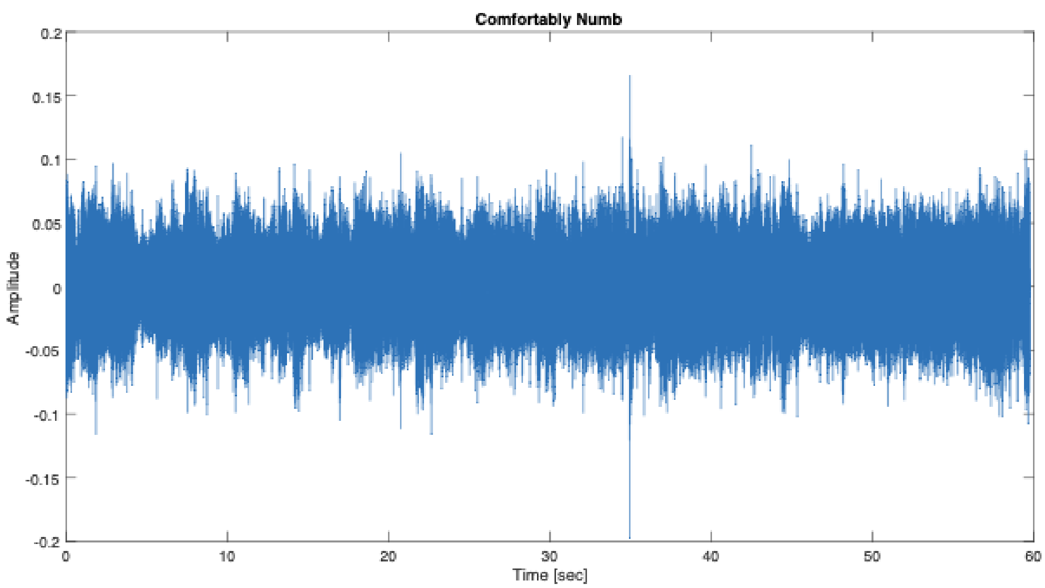

Figure 4. Signals of Floyd clip

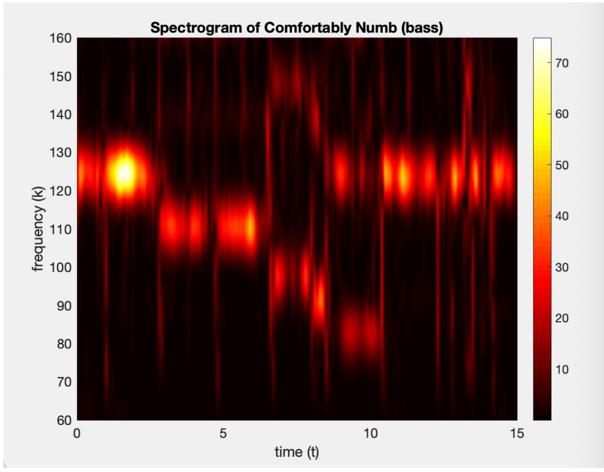

Figure 5. Spectrogram for the guitar in in Floyd clip

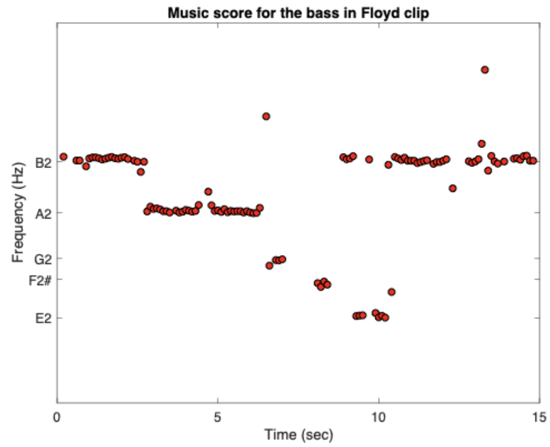

Figure 6. Music score for the bass in Floyd clip

Figure 1 and 4 shows the signals distribution with increment of time of two audio clips. Figure 2 shows a spectrogram visualization of the music notes for the guitar in GNR clip, and Figure 3 clearly identifies the music notes are **C4# D4# F4# G4# C5# F5 F5#** with frequencies 277.18, 311.13, 369.99, 415.30, 554.37, 698.46, 739.99 in Hertz.

Similarly, Figure 5 visualizes the music notes being played by bass in Floyd clip using spectrogram, and the notes are clearly identified in Figure 6, which are **E2 F2# G2 A2 B2** with frequencies 82.407, 92.499, 97.999, 110.00, 123.47 in Hertz.

V. Summary and Conclusions

In this project, we have successfully used Gabor transformation and spectrogram to find music scores for the guitar in the song *Sweet Child O' Mine* and the bass in the song *Comfortably Numb*. We have also filtered out the bass in the Floyd clip and reconstructed the music score to perform a guitar solo in the end.

Meanwhile, we also realize the negative impact of oversampling will result the Matlab taking a long time to run out the code.

Appendix A. MATLAB functions

abs(X): calculates the absolute value of the elements of X. When input is complex, output returns the complex modulus (magnitude).

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.

fft(X): performs fast fourier transform and returns the N-dimensional discrete fourier transform of the N-D array X.

fftshift(X): rearranges the elements of X by shifting zero-frequency component to center of spectrum.

pcolor(C): creates a pseudocolor or "checkerboard" plot of matrix C.

yticks(ticks): specifies the values for the tick marks along the y-axis of the current axes.

yticklabels(labels): specifies the y-axis tick labels for the current axes.

Appendix B. MATLAB codes

```
% Clean workspace
clear all; close all; clc

%% Part 1. GNR Clip
figure(1)
[y, Fs] = audioread('GNR.m4a');
tr_gnr = length(y)/Fs; % record time in seconds
plot((1:length(y))/Fs,y);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Sweet Child O'' Mine');
% p8 = audioplayer(y,Fs); playblocking(p8);

% Set up
y = y.'; n = length(y); L = tr_gnr;
t1 = linspace(0,L,n+1); t = t1(1:n);
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);

% Apply Gabor Transformation
a = 500;
tau = 0:0.1:L;
Sgt_spec = [];
gnr_guitar = [];
for j = 1:length(tau)
    g = exp(-a*(t-tau(j)).^2);
    Sg = g.*y;
    Sgt = fft(Sg);
    Sgt_spec(:,j) = fftshift(abs(Sgt));
    [M,I] = max(Sgt);
    gnr_guitar = [gnr_guitar; abs(k(I))];
end

% Plot the Spectrogram of GNR clip
figure(2)
pcolor(tau,ks,Sgt_spec)
shading interp
colormap(hot)
colorbar
```

```
set(gca,'Fontsize',12,'ylim',[200 800]);
xlabel('Time [sec]'); ylabel('Frequency [Hz]');
title('Spectrogram of Sweet Child O'' Mine (guitar)');

% Plot the music score for the guitar in GNR clip
figure(3)
plot(tau, gnr_guitar, 'ko', 'MarkerFaceColor', 'r')
xlabel('Time (sec)'), ylabel('Frequency (Hz)')
yticks([277.18, 311.13, 369.99, 415.30, 554.37, 698.46, 739.99])
yticklabels({'C4#', 'D4#', 'F4#', 'G4#', 'C5#', 'F5', 'F5#'})
set(gca,'Fontsize',12,'ylim',[200 800])
title('Music score for the guitar in GNR clip')

%% Part 1. Floyd Clip
figure(4)
[y, Fs] = audioread('Floyd.m4a');
tr_floyd = length(y)/Fs; % record time in seconds
plot((1:length(y))/Fs,y);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Comfortably Numb');
% p8 = audioplayer(y,Fs); playblocking(p8);

% Set up
L = 15; n = L*Fs; y = y(1:n).';
t1 = linspace(0,L,n+1); t = t1(1:n);
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);

% Apply Gabor Transformation
a = 500;
tau = 0:0.1:L;
Sgt_spec = [];
floyd_bass = [];
for j = 1:length(tau)
    g = exp(-a*(t-tau(j)).^2);
    Sg = g.*y;
    Sgt = fft(Sg);
    Sgt_spec(:,j) = fftshift(abs(Sgt));
    [M,I] = max(Sgt);
    floyd_bass = [floyd_bass; abs(k(I))];
```

```
end

% Plot the Spectrogram of Floyd Clip
figure(5)
pcolor(tau,ks,Sgt_spec)
shading interp
colormap(hot)
colorbar
set(gca,'Fontsize',12,'ylim',[60 160]);
xlabel('time (t)'); ylabel('frequency (k)');
title('Spectrogram of Comfortably Numb (bass)');

% Plot the music score for the bass in GNR clip
figure(6)
plot(tau, floyd_bass, 'ko', 'MarkerFaceColor', 'r')
xlabel('Time (sec)'), ylabel('Frequency (Hz)')
yticks([82.407, 92.499, 97.999, 110.00, 123.47])
yticklabels({'E2', 'F2#', 'G2', 'A2', 'B2'})
set(gca,'Fontsize',12,'ylim',[60 160])
title('Music score for the bass in Floyd clip')

%% Part 2. Isolate the bass in Comfortably Numb
[y, Fs] = audioread('Floyd.m4a');
tr_floyd = length(y)/Fs; % record time in seconds
y = y(1:end-1);
yt = fft(y);

y = y.'; n = length(y); L = tr_floyd;
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);

% Shannon filter
[Ny,Nx] = size(y);
wy = 10000;
sfilter = ones(size(y));
sfilter(wy+1:Nx-wy+1) = zeros(1,Nx-2*wy+1);

ytf = yt.*sfilter;
yf = ifft2(ytf);
```

```
figure(7)
subplot(2,1,1)
plot(ks, abs(fftshift(yt)),'r')
hold on
plot(ks, fftshift(sfilter),'k','Linewidth',2)

subplot(2,1,2)
plot(ks,abs(yf),'Linewidth',2);
```