Amath 482 Homework 2 Report
Rock and Roll and the Gabor Transform

Colin Kong

February 11th, 2021

Abstract: The project aims to provide a frequency-time analysis for two famous Rock n Roll songs by Pink Floyd and Guns N' Roses. We will use spectrograms and Gabor transform in order to analyze the two songs.

Section I: Introduction and Overview

Introduction: The problem is about analyzing a portion of two of the greatest rock and roll songs of all time. Download the two files GNR.m4a (14 second clip) and Floyd.m4a (60 second clip) that are included with the homework. These files play clips of the songs Sweet Child O' Mine by Guns N' Roses and Comfortably Numb by Pink Floyd, respectively.

Overview: To set the frequency-time analysis, we should first apply the Gaber transform to the data after loading each of the two songs. We should filter out the overtones to get clean results and figure them out as spectrograms. Later, in order to isolate the bass part in the song Comfortably Numb, I added a Shannon filter to the original signal.

Section II: Theoretical Background

Gabor Transforms:

Let us consider a filter function g(t). Then, shifting by τ and multiplying by a function f(t) represents the filtered function f(t)g(t−τ), with the filter centered at τ. The Gabor transform, or STFT, is then given precisely by

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

That is, for a fixed τ the function $\widetilde{f_g}(\tau, k)$ gives you information about the frequency components near time τ. We should note that your results are dependent on the choice of filter g(t), hence the subscript g on $\tilde{f}$. There are many choices for g and some commonly used assumptions are:

1. The function g is real and symmetric.
2. $\|g\|_2 := (\int_{-\infty}^{\infty} |g(t)|^2 dt)^{\frac{1}{2}} = 1$. That is, the L_2-norm of g is set to unity.

Note: the L_2-norm $\|\cdot\|_2$ is commonly understood to represent the total energy of a function.

The inverse of the Gabor transform is given by:

$$f(t) = \frac{1}{2\pi\|g\|_2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \widetilde{f_g}(\tau, k)\, g(t-\tau) dk d\tau$$

Filtering:
Filtering is to filter out the noise in order to better detect the signal. A filter is a mathematical function that we can apply:

$$F(k) = e^{-\tau(k-k_0)^2}$$

This is a Gaussian function with k standing for the frequency, τ determining the width of the filter and the constant k_0 determining the center of the filter. Using the filter, we will be able to denoise the data and determine the path of submarine by multiplying filter to the Fourier Transformed data and later we will be using the inverse Fourier Transform to set the results back to the time domain.

Shannon Filter:
This is just a step function - it takes the values 0 or 1 - so it is just a sharp cut off in frequency space that removes all frequencies above some threshold. Shannon filter is based on the Nyquist–Shannon sampling theorem is a theorem in the field of signal processing which serves as a fundamental bridge between continuous-time signals and discrete-time signals. It establishes a sufficient condition for a sample rate that permits a discrete sequence of samples to capture all the information from a continuous-time signal of finite bandwidth.

Section III: Algorithm implementation and Development

The project has four steps:

1. **Run the start code in MATLAB:** Import and convert the GNR.m4a (14 second clip) and Floyd.m4a (60 second clip).

2. **Through the use of the Gabor filtering we used in class, reproduce the music score for the guitar in the GNR clip, and the bass in the Floyd clip. Both are clearly identifiable, which has the music scale in Hertz:** Use Gabor transform to transform the two songs into a frequency-time analysis which is filtered by the algorithm of Gaussian filter. Draw the spectrograms and music sheets based on the filtered signal of the two songs. We can see the nodes of the music score for the guitar in the GNR clip, and the bass in the Floyd clip from the spectrograms and the music sheet.

3. **Use a filter in frequency space to try to isolate the bass in Comfortably Numb:**

Apply the Shannon filter by setting the width to 10000 and finish it with Fast Fourier Transform. After that, when you play the song Comfortably Numb, the bass part would not be heard.

4. **See how much of the guitar solo you can put together in Comfortably Numb. It may help to look at smaller portions of the clip to guide your reconstruction of the music score:** draw the spectrum of the Floyd.clip

Section IV: Computational Results

1. Spectrogram of GNR.clip

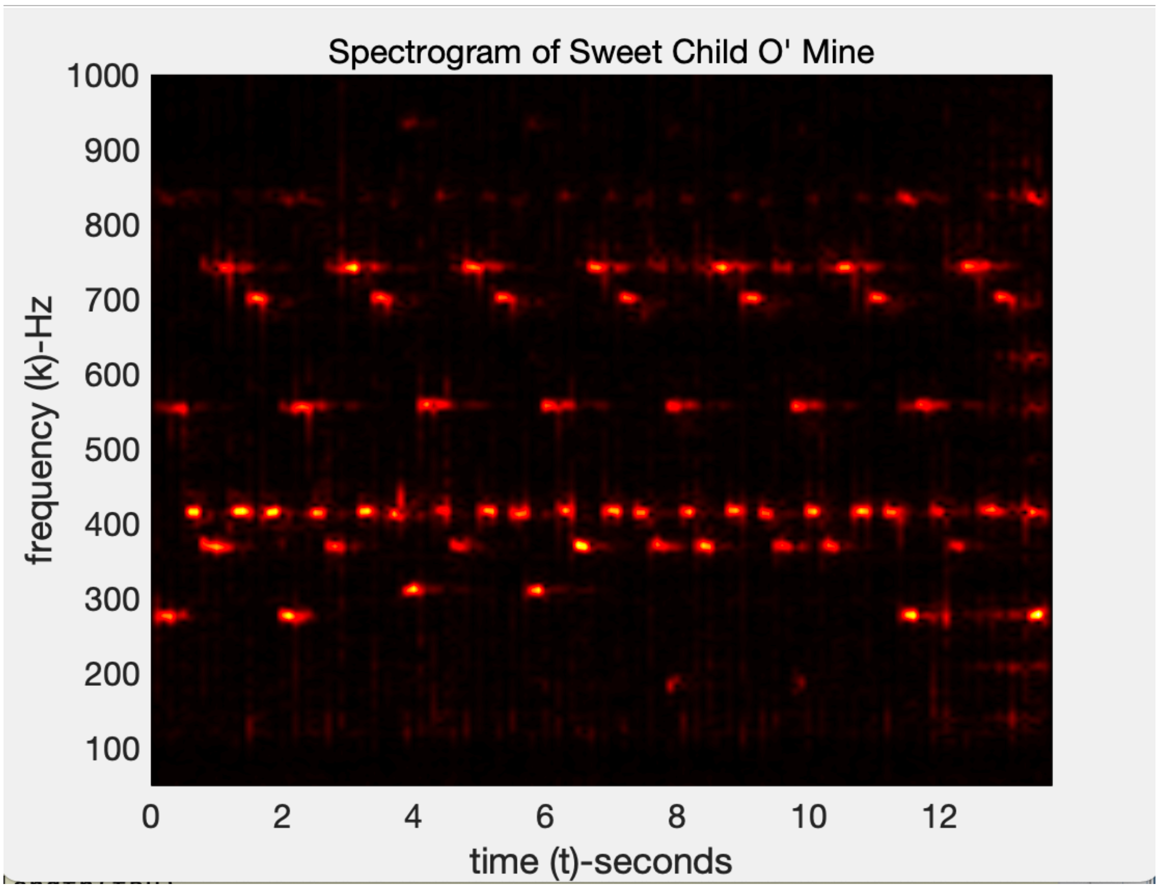

Figure (1): the frequency is set to the limit of 50 to 1000Hz, since I searched that the normal frequency of a guitar is between 88 to 1200Hz.

2. Music score of the guitar part of GNR.clip

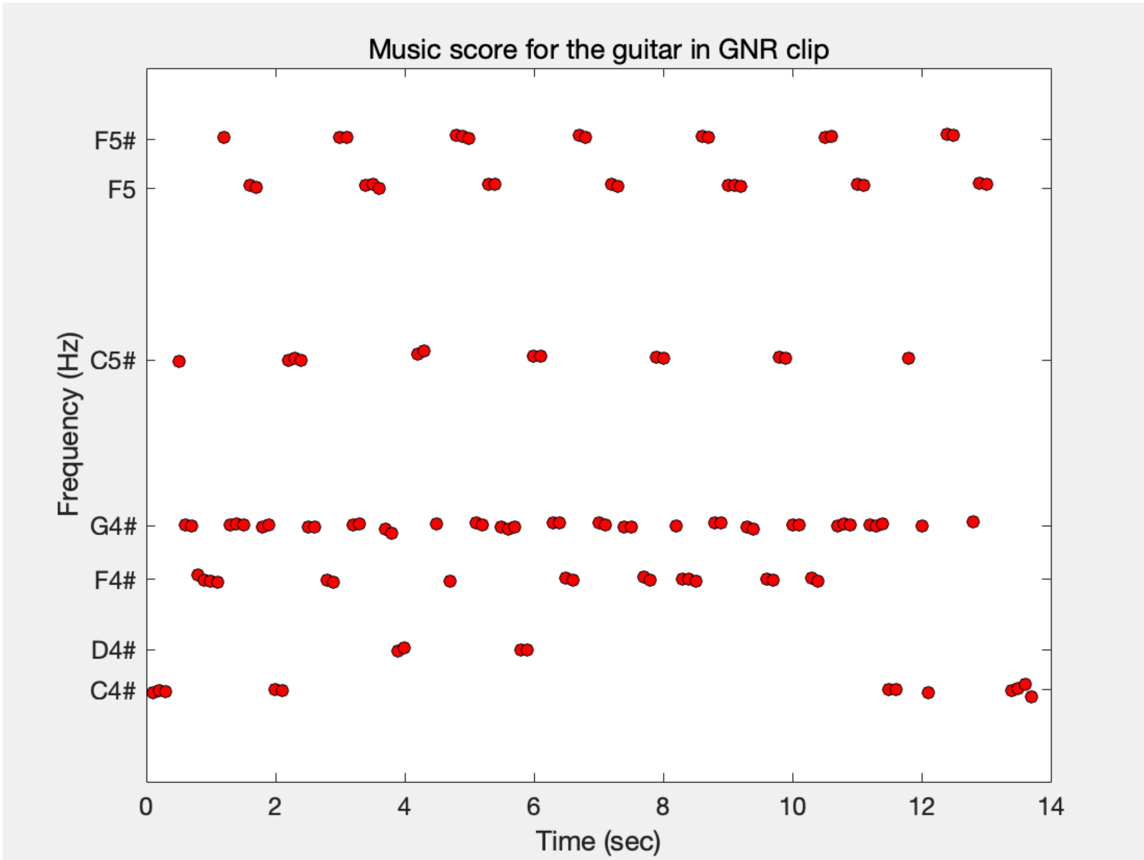

Figure (2): The music scores are F5#, F5, C5#, G4#, F4#, D4# and C4#, which their frequencies are (739.99, 698.46, 554.37, 415.30, 369.99, 311.13, 277.18).

3. Spectrogram of Floyd.clip

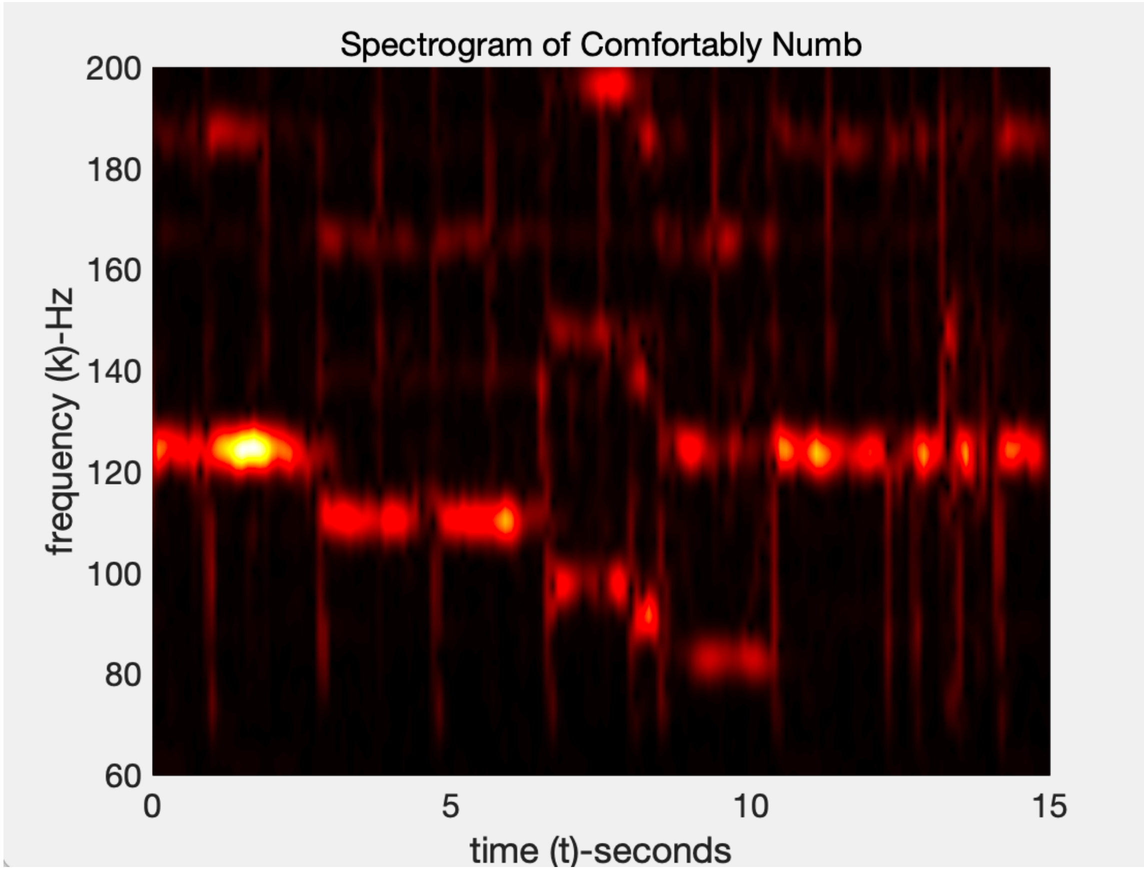

Figure (3): the frequency is set to the limit of 80 to 200Hz, since I searched that the normal frequency of a bass is between 16 to 256Hz.

4. Music score of the bass part of Floyd.clip

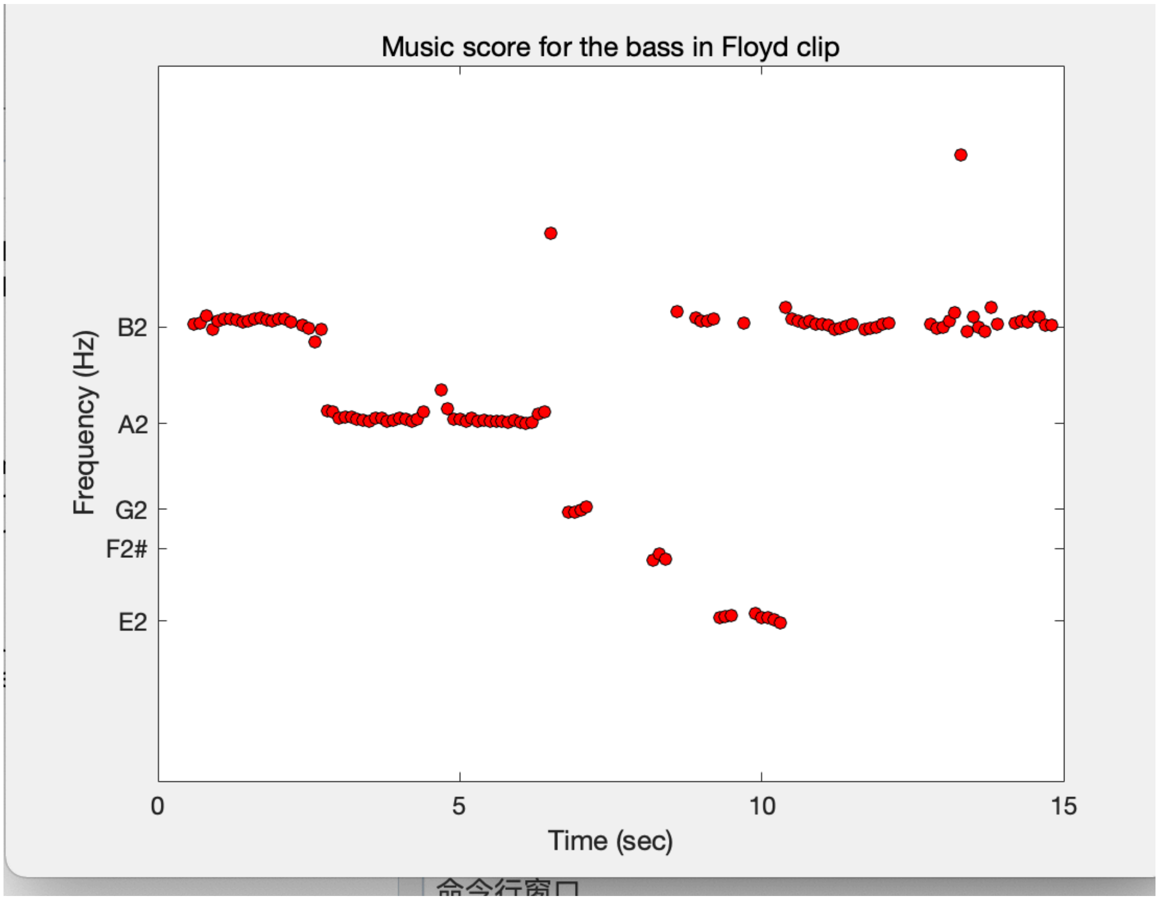

Figure (4): The music scores are B2, A2, G2, F2#, E2 and back to B2, which their frequencies are (123.47, 110.00, 97.999, 92.499, 82.407 and 123.47).

Section V: Summary and Conclusion

After applying the method of Gabor Transform and Shannon filter, we successfully reproduce the music score for the guitar in the GNR.clip and the bass in the Floyd.clip and be able to isolate the bass part of the Floyd clip. We find the music scores of Sweet Child O' Mine are F5#, F5, C5#, G4#, F4#, D4# and C4#, which their frequencies are (739.99, 698.46, 554.37, 415.30, 369.99, 311.13, 277.18). Further, the music scores of Comfortably Numb are B2, A2, G2, F2#, E2 and back to B2, which their frequencies are (123.47, 110.00, 97.999, 92.499, 82.407 and 123.47). When we replay the Floyd.clip after applying the Shannon filter, we could clearly find the song is played without the bass part. And the matrix is too large for the spectrogram of the guitar part of Floyd.clip to be drawn.

Appendix A: MATLAB Functions Used

1. **[y,fs] = audioread(filename)** – reads data from the file named filename, and returns sampled data, y, and a sample rate for that data, Fs.
2. **linspace(x1, x2)** – returns a row vector of 100 evenly spaced points between x1 and x2.
3. **fftshift(x)** – rearranges a Fourier transform X by shifting the zero-frequency component to the center of the array.
4. **fft(x)** – computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.
5. **pcolor(X,Y,C)** – creates a pseudocolor plot using the values in matrix C. A pseudocolor plot displays matrix data as an array of colored cells (known as *faces*) and specifies the *x*- and *y*-coordinates for the vertices.
6. **ifft2(y)** – returns the two-dimensional discrete inverse Fourier transform of a matrix using a fast Fourier transform algorithm. If Y is a multidimensional array, then ifft2 takes the 2-D inverse transform of each dimension higher than 2. The output X is the same size as Y.

Appendix B: MATLAB Codes

```
%% Start code
figure(1)
[y, Fs] = audioread('GNR.m4a');
y = y.';
trgnr = 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);
%%
n = length(y);
L = trgnr;
t2 = linspace(0,L,n+1);
t = t2(1:n);
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);
a = 500;
tau = 0:0.1:L;
Gnrguitar = [];
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);
    Gnrguitar = [Gnrguitar; abs(k(I))];
end
figure(2) %% spectrogram of the guitar part of GNR.clip
pcolor(tau,ks,abs(Sgt_spec))
shading interp
set(gca,'ylim',[50 1000],'Fontsize',16)
colormap(hot)
xlabel('time (t)-seconds'), ylabel('frequency (k)-Hz')
title('Spectrogram of Sweet Child O'' Mine','Fontsize',16)
figure(3) %% Music scoree sheet
plot(tau, Gnrguitar, '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')
%%
figure(4)
[y, Fs] = audioread('Floyd.m4a');
trfloyd = length(y)/Fs; % record time in seconds
plot((1:length(y))/Fs,y);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Comfortably Numb');
L = 15;
n = L*Fs;
y = y(1:n).';
t2 = linspace(0,L,n+1);
t = t2(1:n);
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);
a = 200;
tau = 0:0.1:L;
Floydbass = [];
for j = 1:length(tau)
    g = exp(-a*(t - tau(j)).^2);
    Sg = g.*y;
    Sgt = fft(Sg);
    Sgt_spec2(:,j) = fftshift(abs(Sgt));
```

```
    [M,I] = max(Sgt);
    Floydbass = [Floydbass; abs(k(I))];
end
figure(5) %% spectrogram of the bass part of Floyd.clip
pcolor(tau,ks,abs(Sgt_spec2))
shading interp
set(gca,'ylim',[60 200],'Fontsize',16)
colormap(hot)
xlabel('time (t)-seconds'), ylabel('frequency (k)-Hz')
title('Spectrogram of Comfortably Numb','Fontsize',16)
figure(6) %% Music scoree sheet
plot(tau, Floydbass, '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')
%%
[y, Fs] = audioread('Floyd.m4a');
trfloyd=length(y)/Fs;
y = y(1:end-1);
y = y.';
n = length(y);
L = trfloyd;
k = (1/L)*[0:n/2-1 -n/2:-1];
ks = fftshift(k);
yt = fft(y);
[Ny,Nx] = size(yt);
wx = 10000;
wy = 10000;
filter = ones(size(y));
filter(wy+1:Nx-wy+1) = zeros(1,Nx-2*wy+1); %% apply Shannon filter
yf = yt.*filter;
ynf = ifft2(yf);
figure(7)
plot(ks,abs(ynf),'Linewidth',2);
t2 = linspace(0,L,n+1);
t = t2(1:n);
a = 500;
tau = 0:0.1:L;
for j = 1:length(tau)
```

```
    g = exp(-a*(t - tau(j)).^2);
    Sg = g.*ynf;
    Sgt = fft(Sg);
    Sgt_spec3(:,j) = abs(Sgt);
end
figure(8)
pcolor(tau,ks,abs(Sgt_spec3))
shading interp
set(gca,'ylim',[50 1000],'Fontsize',16)
colormap(hot)
xlabel('time (t)-seconds'), ylabel('frequency (k)-Hz')
title('Spectrogram of Comfortably Numb','Fontsize',16)
```