Assignment 2

Rock & Roll and the Gabor Transform

Author: Sofia Chen

Abstract: This assignment is to analyze a portion of two of the greatest rock and roll songs.

Section I. Introduction and Overview

We have two files play clips of the songs Sweet Child O' Mine by Guns N' Roses and Comfortably Numb by Pink Floyd, respectively. First, we want to reproduce the music score for the guitar in the GNR clip, and the bass in the Floyd clip. Then, we want to isolate the bass in the Floyd clip. Lastly, I will see how much of the guitar solo that can put together in Comfortably Numb.

Section II. Theoretical Background

We use Gabor transform to transform the signal to the frequency domain. The function to be Gabor transformed is first multiplied by a Gaussian function, which can be regarded as a window function, and the resulting function is then transformed with a Fourier transform to derive the time-frequency analysis. We use Gabor transform instead of Fourier transform since we want to able to consider all time windows of a given length (Slide the time filter across the domain and consider all possible locations for the filter.

Then we make a spectrum with the horizontal direction being the value of window centre and the vertical direction the frequency. And then we identify the notes being played by the known frequencies of the instruments. The frequencies which have the highest occurrence, represents the notes in frequency domain.

Notice that in order to get wavenumbers in Hz instead of angular frequency, we scale our wavenumbers by 1/(length of recording in seconds).

Section III. Algorithm Implementation and Development

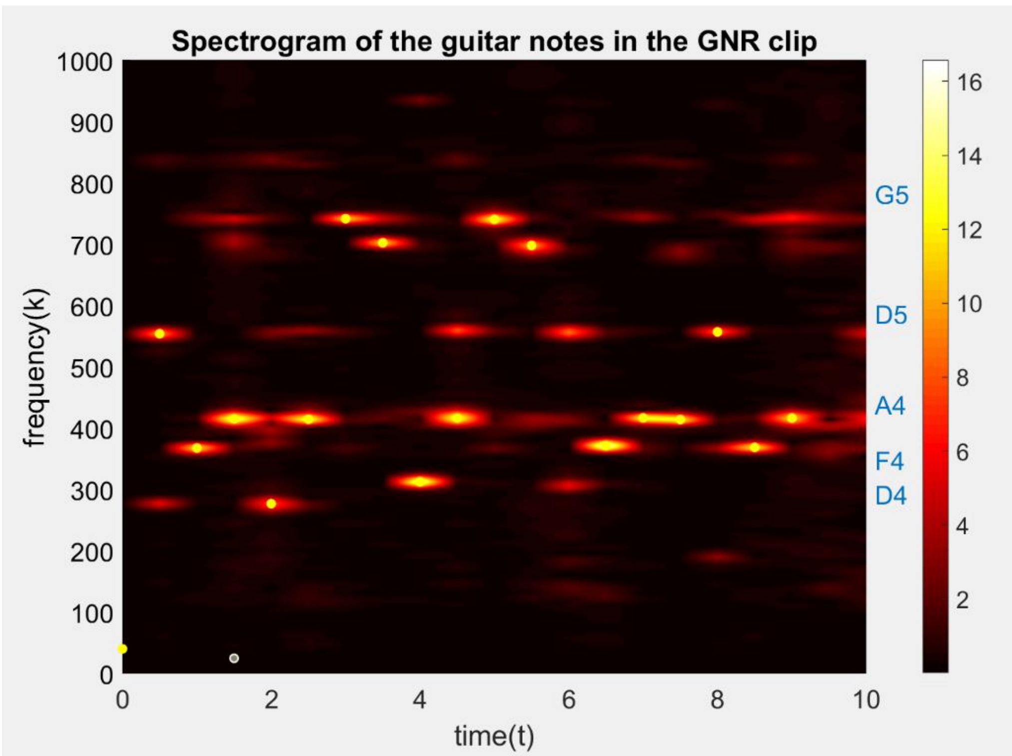
I first imported the music pieces and converted it to a vector representing the music. After that I used Gabor transform to turn the vector into frequency space. Then I wanted to visualize things in the time-frequency domain. I created the spectrogram. Since the music pieces have repeated pattern, I only used the first 10 seconds to analysis for each pieces of song.

I create the spectrogram by doing a for-loop. Inside each loop, I multiplied the changing window function with original vector, then do the Gabor transform, and I stored the results in a variable called ygt_spec. I also found out the maximums of the transformed function to find the frequencies that has the highest occurrences in each loop, which are the brightest points on the spectrogram. Moreover, I used a big ***a*** value (1000 or 500) in order to make the notes more identifiable in time domain. After that, I plot the scatter points, which are the corresponding notes, on the spectrogram.

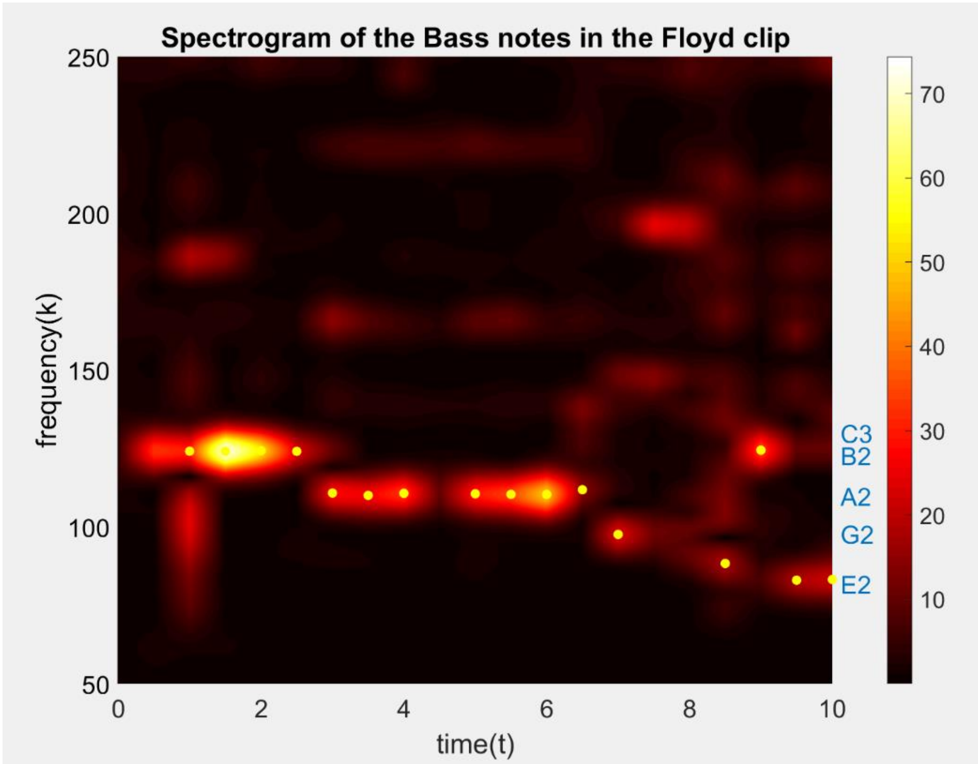
For the song *Sweet Child O' Mine,* there is only guitar playing. The notes spread between 300Hz to 800Hz. I also labeled to corresponding names of note in the right.

For the song *Comfortably Numb*, the approach is similar, but I add two filters to find the bass notes and guitar notes respectively. Bass has the frequency between 50Hz to 250Hz, so I filter the frequency between this interval. Similarly, Guitar has the frequency between 250Hz to 500Hz, so I filter the frequency between this interval to find the Guitar notes in this interval.

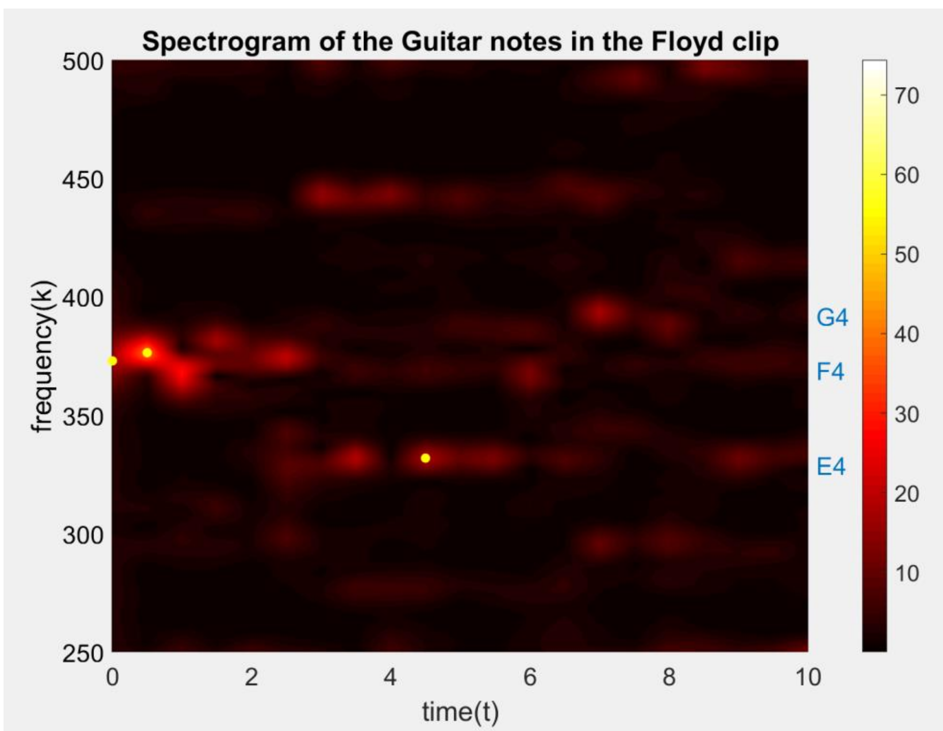
Section IV. Computational Results



This spectrogram represents the guitar notes (yellow points) in the GNR clip. The labels in the right illustrate the name of the corresponding notes.



This spectrogram represents the bass notes (yellow points) in the Floyd clip. The labels in the right illustrate the name of the corresponding notes.



This spectrogram represents the guitar notes (yellow points) in the Floyd clip. The labels in the right illustrate the name of the corresponding notes.

Section V. Summary and Conclusions

In this assignment, I used Gabor transform to convert the music signal. In order to better visualize the data in frequency-time domain, I create the spectrograms for each pieces of music. Then, according to the music scale in Hertz, we find out the two instruments notes on the spectrogram. We calculate the maximums of the transformed function to find where exactly the notes in. We also use filter to distinguish different instruments in the data. It is pretty cool and fun to play music in Matlab.

Appendix A. MATLAB functions used and brief implementation explanation

- ***audioread*** - Read audio file: This MATLAB function reads data from the file named filename, and returns sampled data, y, and a sample rate for that data, Fs.
- ***fft*** - Fast Fourier transform: This MATLAB function computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm
- ***fftshift*** - Shift zero-frequency component to center of spectrum: This MATLAB function rearranges a Fourier transform X by shifting the zero-frequency component to the center of the array.
- ***pcolor*** - Pseudocolor plot: This MATLAB function creates a pseudocolor plot using the values in matrix C.
- ***scatter*** - Scatter plot: This MATLAB function creates a scatter plot with circles at the locations specified by the vectors x and y
- ***yticks*** - Set or query y-axis tick values: This MATLAB function sets the y-axis tick values, which are the locations along the y-axis where the tick marks appear.
- ***yticklabels*** - Set or query y-axis tick labels: This MATLAB function sets the y-axis tick labels for the current axes.

Appendix B. MATLAB codes

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

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);

k = (1/tr_gnr)*[0:length(y)/2-1 -length(y)/2:-1];
ks = fftshift(k);
t2 = linspace(0,tr_gnr,length(y)+1); t = t2(1:length(y));

% Creating a spectrogram
a = 1000;
tau = 0:0.5:10;
```

```
Notes = zeros(length(tau),1);
for j = 1:length(tau)
    g = exp(-a*(t - tau(j)).^2);
    yg = g.*(y.');
    ygt = fft(yg);
    ygt_spec(:,j) = fftshift(abs(ygt));
    [m,i] = max(fftshift(abs(ygt)));
    Notes(j,1) = abs(ks(i));
end
figure(2)
pcolor(tau, ks, ygt_spec);
shading interp
set(gca, 'ylim', [0 1000], 'Fontsize', 16)
colormap(hot)
colorbar
xlabel('time(t)'), ylabel('frequency(k)')
yyaxis right
scatter(tau, Notes, 'y', 'filled')
ylim([0 1000])
yticks([293 349 440 587 783])
yticklabels({'D4' 'F4' 'A4' 'D5' 'G5'})
title('Spectrogram of the guitar notes in the GNR clip')

figure(3)
[y2, Fs2] = audioread('Floyd.m4a');
y2(end)=[];
tr_flo = length(y2)/Fs2; % record time in seconds
plot((1:length(y2))/Fs2,y2);
xlabel('Time [sec]'); ylabel('Amplitude');
title('Comfortably Numb');
p8 = audioplayer(y2,Fs2); playblocking(p8);

k2 = (1/tr_flo)*[0:length(y2)/2-1 -length(y2)/2:-1];
ks2 = fftshift(k2);
t22 = linspace(0,tr_flo,length(y2)+1); t1 = t22(1:length(y2));

%% Filter for Bass in Floyd clip
y2t = fft(y2);
y2_filter = ifft(y2t.*fftshift(50<abs(y2t)<250));

% Creating a spectrum
a = 500;
tau = 0:0.5:10;
Notes2 = zeros(length(tau),1);
for j = 1:length(tau)
    g = exp(-a*(t1 - tau(j)).^2);
    y2g = g.*(y2_filter.');
    y2gt = fft(y2g);
    y2gt_spec(:,j) = fftshift(abs(y2gt));
    [m,i] = max(fftshift(abs(y2gt)));
    Notes2(j,1) = abs(ks2(i));
end
figure(4)
```

```
pcolor(tau, ks2, y2gt_spec);
shading interp
set(gca, 'ylim', [50 250], 'Fontsize', 16)
colormap(hot)
colorbar
xlabel('time(t)'), ylabel('frequency(k)')
yyaxis right
scatter(tau, Notes2, 'y', 'filled')
ylim([50 250])
yticks([82 98 110 123 130])
yticklabels({'E2' 'G2' 'A2' 'B2' 'C3'})
title('Spectrogram of the Bass notes in the Floyd clip')

%% Filter for Guitar in Floyd clip
y2t = fft(y2);
y2_filter = ifft(y2t.*fftshift(250<abs(y2t)<500));

% Creating a spectrum
a = 500;
tau = 0:0.5:10;
Notes2 = zeros(length(tau),1);
for j = 1:length(tau)
    g = exp(-a*(t1 - tau(j)).^2);
    y2g = g.*(y2_filter.');
    y2gt = fft(y2g);
    y2gt_spec(:,j) = fftshift(abs(y2gt));
    [m,i] = max(fftshift(abs(y2gt)));
    Notes2(j,1) = abs(ks2(i));
end
figure(5)
pcolor(tau, ks2, y2gt_spec);
shading interp
set(gca, 'ylim', [250 500], 'Fontsize', 16)
colormap(hot)
colorbar
xlabel('time(t)'), ylabel('frequency(k)')
yyaxis right
scatter(tau, Notes2, 'y', 'filled')
ylim([250 500])
%yticks([329 369 392])
%yticklabels({'E4' 'F4' 'G4'})
title('Spectrogram of the Guitar notes in the Floyd clip')
```