Coding Project 2: Parsing musical frequency signatures

Tyler Shakibai

Abstract

Frequency analysis using the Fourier transform is very effective when analyzing stationary signals. However, this method is unable to describe when those frequencies occur. In this project, we utilize the Gabor transform to perform time-frequency analysis in order to isolate and reconstruct the instrument types from a mystery song. In this particular song, instruments are introduced one at a time which makes isolating frequency signatures more intuitive. Using the Gabor transform, we create a spectrogram in order to study the frequencies from the given sound clip of the song. To do so, we employ the discrete Gabor transform to numerically transform the frequencies we measure to be interpretable. After correctly applying the transform, we are able to isolate the frequencies of the drumbeats from those of the guitar and other string instruments.

1 Introduction

In music theory, a note is the representation of a musical sound. Notes are used to represent the pitch and duration of a sound when using musical notation. The fundamental frequency is the lowest frequency produced by a particular instrument. If the ratio of the fundamental frequency of two notes is equal to any integer power of two, they are perceived very similarly. Because of that, all notes with these kinds of relations can be grouped under the same pitch class. The human ear can only hear frequencies that are limited to a specific range. The audible frequency range for humans is typically given as being between about 20 Hz and 20,000 Hz (20 kHz).

In this report, we will discuss the theoretical background of the Gabor transform as it pertains to signal processing. We will then apply this transform to our data in order to assist in the isolation of varying instrument notes based on their frequency signature. A quick look at our data yields the following plot.

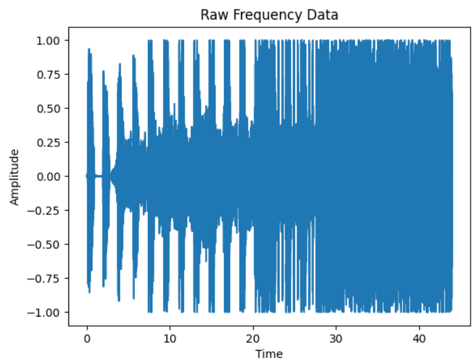

Figure 1: Amplitude versus time of the raw frequency data.

Using the Gabor transform, we hope to make sense of these convoluted signal measurements. After we successfully transform and filter our data, we will discuss the results found and discuss the efficacy of our approach.

2 Theoretical Background

In this section, we will delve into the theoretical underpinnings of the Gabor transform. This discussion will be focused primarily on the derivation of the Gabor transform as it relates to processing information in frequency space.

2.1 The Gabor Transform

The Fourier transform is ill-suited to perform time-frequency analysis, and as such, we will look to another transform that is more applicable to our problem. Named after Dennis Gabor, the Gabor transform is a special case of the short-time Fourier transform. It is used to determine the frequency of a signal sinusoidal along with the phase content of local sections through time.

The function that is subject to the Gabor transform is first multiplied by a Gaussian filter, which can be interpreted as a window function, with the resulting function then being transformed with a Fourier transform. Here, a window function means a greater weight will be associated with the signal near the time being analyzed. The Gabor transform of a signal $f(t)$ is defined as

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

Here, $g(t)$ is defined as the centered Gaussian filter over some sub-interval of the domain, or window. In the formulated provided, we consider τ to be the location at which the filter is centered, outside of which the signal will be damped. Each window captures a different portion of the signal, which is then transformed into frequency space. It is important to note that the Gabor transform is invertible, as such there exists an inverse Gabor transform.

2.2 The Discrete Gabor Transform

To be numerically applicable, the discrete Gabor transform is used to run on computing hardware where the domain must be discretized. The discrete Gabor transform is defined as

$$y(t) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C_{nm} \cdot g_{nm}(t)$$

where $g_{nm}(k) = s(k - mN) \cdot e^{i\Omega nk}$. In the discrete version, the continuous parameter t is replaced by the discrete time k.

Similar to the discrete Fourier transform, the frequency domain is split into N discrete partitions. Taking the inverse transformation of these N spectral partitions yields N values $y(k)$ for each time window, which in turn consists of N sample values. For M time windows with N samples, each signal $y(k)$ contains $K = N \cdot M$ samples.

3 Numerical Methods

First, the data is loaded and plotted, in order to get an initial indication of what is being dealt with, but it should not be discernible at this point.

To begin our signal processing, we first partition the data into four separate windows which will be needed when we apply the discrete Gabor transform. We then define our parameters which include the size of our domain, including the number of discrete points it is comprised of, and the linear spaces for both time and $/tau$ as defined above. We then use these parameters to define our frequency space and use the fast Fourier shift to shift our frequency domain.

We are now ready to take the Gabor transforms and create our spectrograms. We start by taking applying a Gaussian filter centered at $/tau$ to the windowed signal. After that, we move our signal into frequency space by taking the fast Fourier transform. We then search for the peak frequencies by pruning through the frequencies with the greatest magnitude, this is relevant because these frequencies can be complex in nature. Once we find the peak frequency we record the indices in order to filter around the peak frequency in the Gaussian. We then apply another Gaussian filter centered around the indices we just isolated. Lastly, we take our filtered frequency data and use the fast Fourier transform shift to realign our signal. We are now able to plot the spectrogram to see the various frequencies coming from our music clip. We perform this entire process for all four windows we created earlier.

To extract the insight we originally sought, we can use the four spectrograms we produce to inform us about which frequencies correspond with what instruments. We can then simply set the frequency values of the frequency range we are looking to isolate to zero, thus leaving the sound of only a certain instrument.

4 Results

Our application of the Gabor transform proved very successful. After applying the transform, we were able to produce spectrograms that distilled the various instrument's frequency signatures in frequency space. The following plots are the spectrograms for the first through fourth window of our Gabor transform.

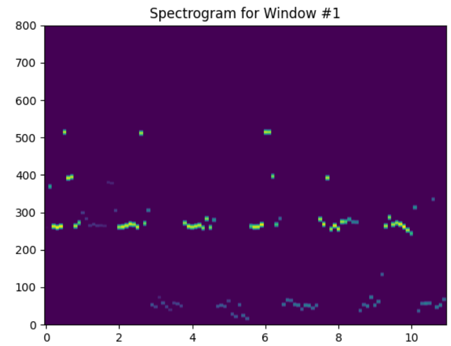

Figure 2: Spectrogram of the first window of the transformed music clip as Hz versus time.

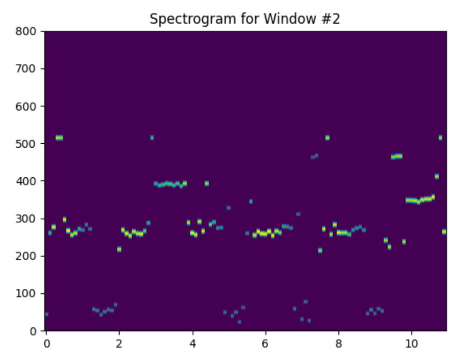

Figure 3: Spectrogram of the second window of the transformed music clip as Hz versus time.

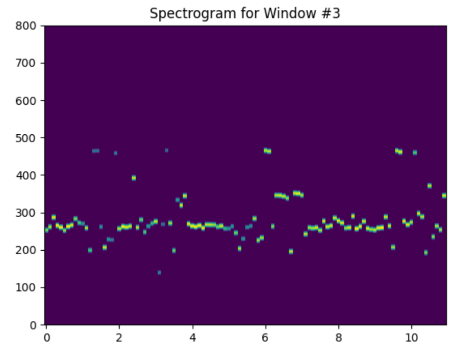

Figure 4: Spectrogram of the third window of the transformed music clip as Hz versus time.

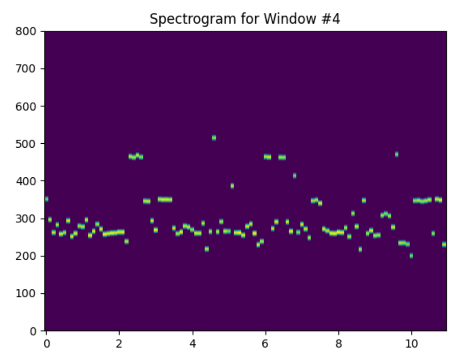

Figure 5: Spectrogram of the fourth window of the transformed music clip as Hz versus time.

These four spectrograms indicate at what frequency and where in time these instruments produce signals, with the intensity of the plot (yellow representing greater intensity, green representing lesser) corresponding to the amplitude of the frequency. By listening to music while examining these plots, we can recognize that the bassline frequencies exist in the interval of $[0, 250)$ Hz, and the guitar frequencies are represented in the interval of $(250, 800]$ Hz. By setting unwanted frequencies to zero, we were able to distill the notes produced individually by both the bassline and guitar.

5 Conclusion

In this project, we utilized the Gabor transform to perform time-frequency analysis in order to isolate and reconstruct the instrument signals from a song clip. Using the discrete Gabor transform, we created a spectrogram in order to study the frequencies from the given sound clip of the song. We were then able to isolate the frequencies of the drumbeats from those of the guitar and listen to them individually. We found that the bassline notes lay in the frequency range between 0 and 250 hertz, while the string instruments resided above 250 but below 800 hertz. The mystery song is called "I'm Shipping Up to Boston" by the Dropkick Murphys. Something interesting I learned during this project is the existence of musical set theory, a way of rigorously defining concepts using mathematical set theory to aid the study of music.

Acknowledgment

I attended office hours with the course's TAs, Katie and Michael, and heavily utilized the Numpy and Scipy Documentation.