Analysis of Piano Sounds

Summer 2021 MDS Course

James Fleming, Bradley Goedde

Mentors: Hengyang Li, Xiaoyu Xie, Sourav Saha

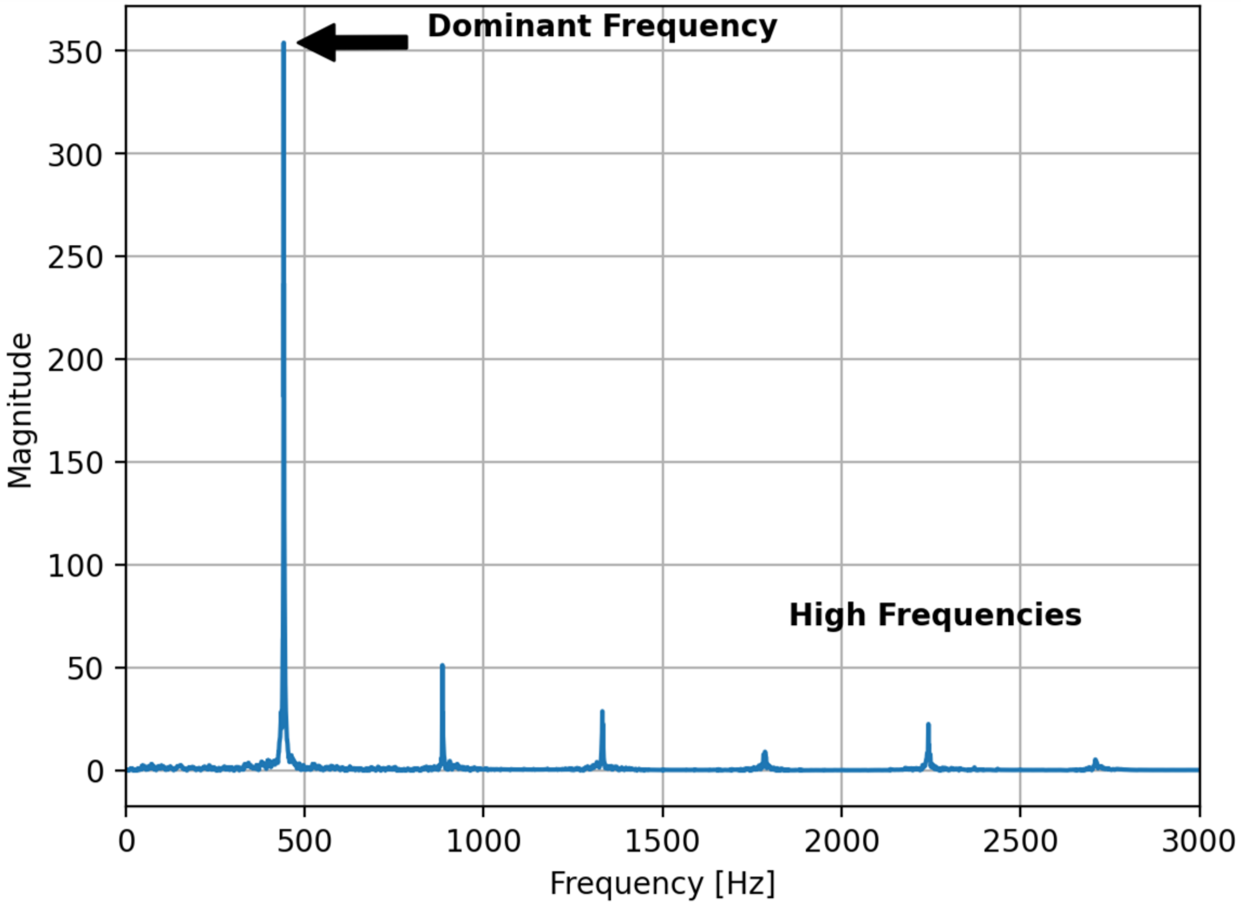
Goal of Project

- For this project, our goal is to learn how to use feature extraction techniques like Fast Fourier Transform and Short-Time Fourier Transform to analyze any piano key and transform the sound.

- The first iteration of the sound modification app was a good framework for our project direction, and since then our goal for the final product is to develop a full piano key database and use FFT, STFT, and neural network.

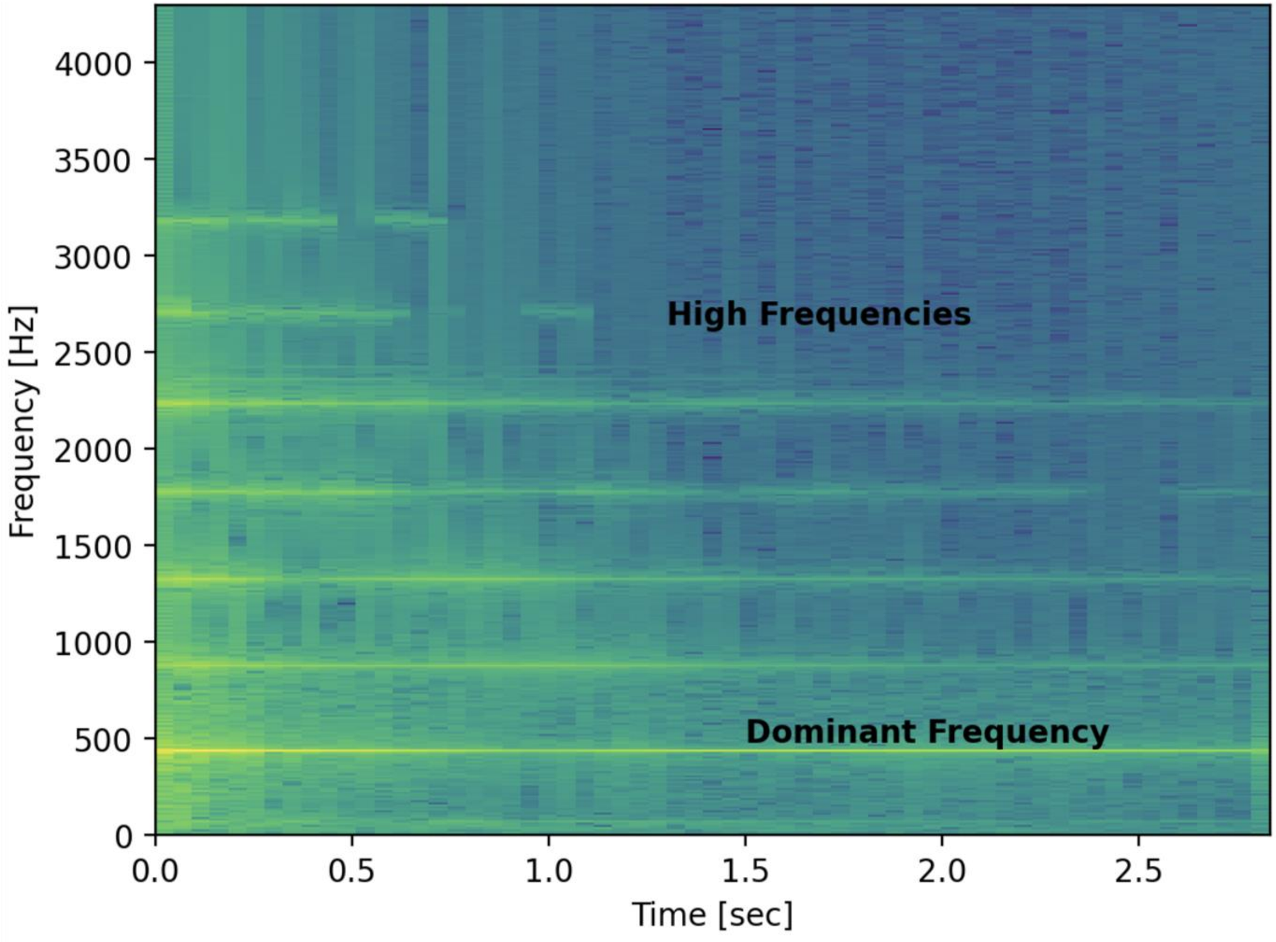
Feature Extraction Techniques

Fast Fourier Transform



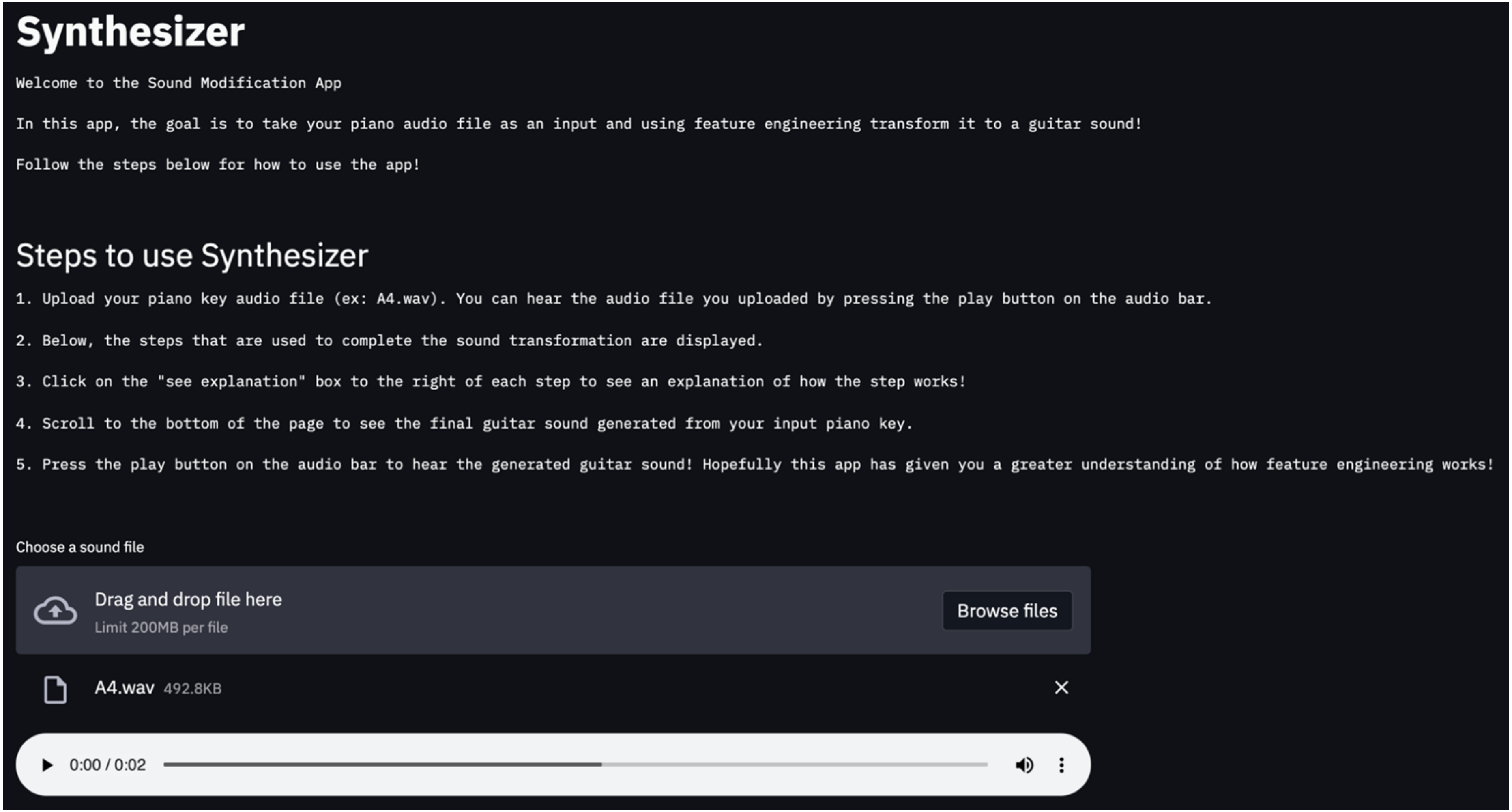
Feature Extraction Techniques

Short Time Fourier Transform



Sound Modification App

https://share.streamlit.io/mhe314/research/sound_mod.py



Synthesizer

Welcome to the Sound Modification App

In this app, the goal is to take your piano audio file as an input and using feature engineering transform it to a guitar sound!

Follow the steps below for how to use the app!

Steps to use Synthesizer

1. Upload your piano key audio file (ex: A4.wav). You can hear the audio file you uploaded by pressing the play button on the audio bar.

2. Below, the steps that are used to complete the sound transformation are displayed.

3. Click on the "see explanation" box to the right of each step to see an explanation of how the step works!

4. Scroll to the bottom of the page to see the final guitar sound generated from your input piano key.

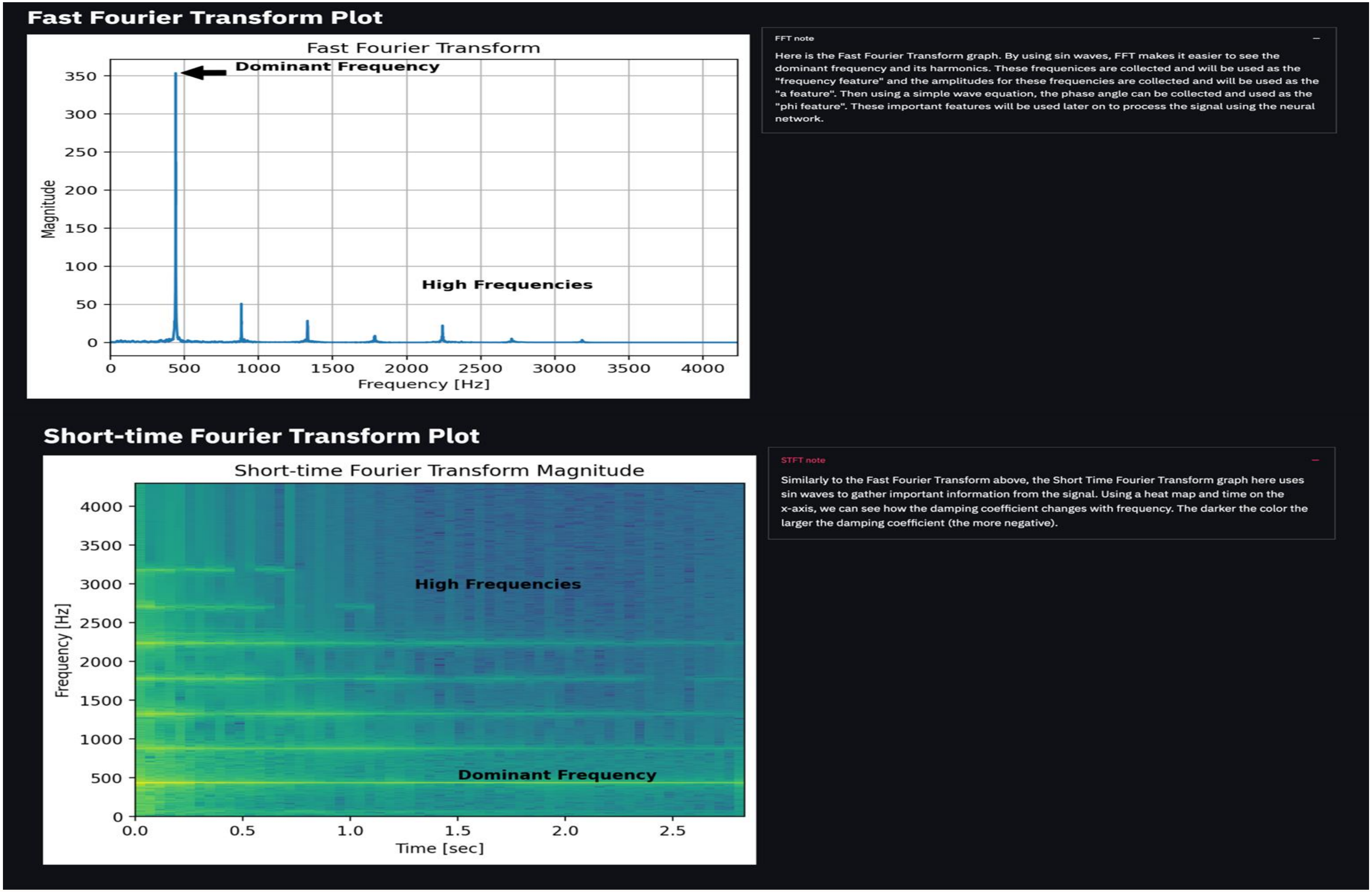
5. Press the play button on the audio bar to hear the generated guitar sound! Hopefully this app has given you a greater understanding of how feature engineering works!

Choose a sound file

A4.wav 492.8KB

Sound Modification App

https://share.streamlit.io/mhe314/research/sound_mod.py



Fast Fourier Transform Plot

FFT note

Here is the Fast Fourier Transform graph. By using sin waves, FFT makes it easier to see the dominant frequency and its harmonics. These frequenices are collected and will be used as the "frequency feature" and the amplitudes for these frequencies are collected and will be used as the "a feature". Then using a simple wave equation, the phase angle can be collected and used as the "phi feature". These important features will be used later on to process the signal using the neural network.

Short-time Fourier Transform Plot

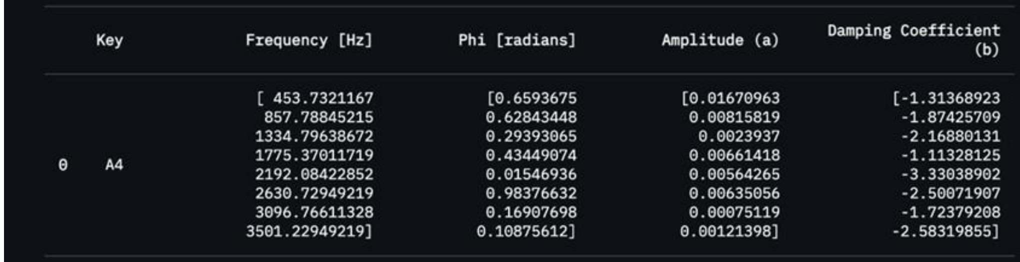
STFT note

Similarly to the Fast Fourier Transform above, the Short Time Fourier Transform graph here uses sin waves to gather important information from the signal. Using a heat map and time on the x-axis, we can see how the damping coefficient changes with frequency. The darker the color the larger the damping coefficient (the more negative).

Sound Modification App

https://share.streamlit.io/mhe314/research/sound_mod.py

Features

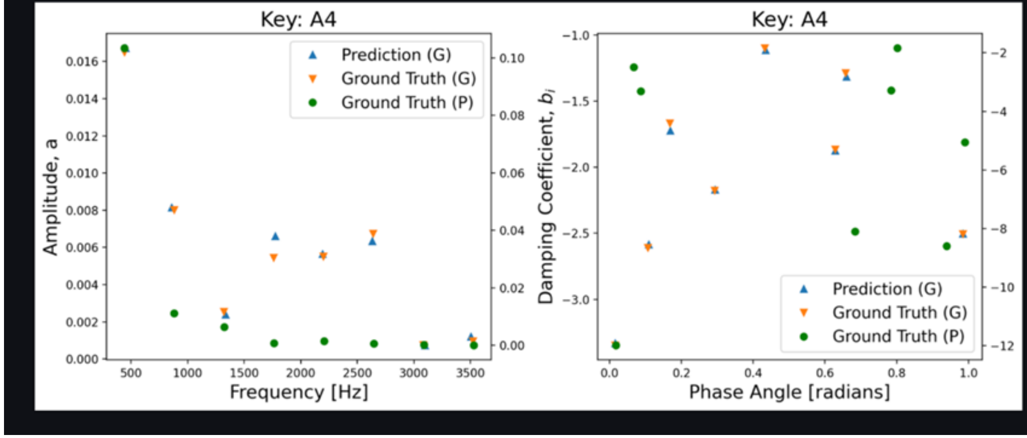


	Key	Frequency [Hz]	Phi [radians]	Amplitude (a)	Damping Coefficient (b)
0	A4	[453.7321167 857.78845215 1334.79638672 1775.37011719 2192.08422852 2630.72949219 3096.76611328 3501.22949219]	[0.6593675 0.62843448 0.29393065 0.43449074 0.01546936 0.98376632 0.16907698 0.10875612]	[0.01670963 0.00815819 0.0023937 0.00661418 0.00564265 0.00635056 0.00075119 0.00121398]	[-1.31368923 -1.87425709 -2.16880131 -1.11328125 -3.33038902 -2.50071907 -1.72379208 -2.58319855]

Features Table explanation –

Here are the resulting guitar features! Once the 4 features displayed in the table were collected from the original audio input using FFT and STFT above, they were passed through a neural network which transformed the features the predicted guitar features displayed in the table.

Plot of Features



Features Plot explanation –

Here we have a plot that compares the generated guitar features with the real sound featuers. The predicted guitar features are compared with the real piano features and the real guitar features. From this overlay, it is clear that the neural network has been trained well, and the features generated are very close to what they should be!

Generated Audio

Generated Audio explanation –

Yay! We've generated a guitar sound! After uploading your piano key, going through feature extraction, then using the neural network to generate the predicted guitar features, these features are used to finally produce the generated guitar audio.

Completed Timeline

1. Learn about FFT, STFT. Develop code and apply FFT and STFT to extract features.

1. Extract features and train neural network. Learn about neural networks from lectures and book (chapter 6). Analyze piano keys neural network code and modify it for use for the app.

1. Use trained neural network to generate the guitar features and compare them with the real piano and guitar features that were used to train the neural network. Then generate the guitar audio.

1. The original app was not working properly, and now a lot of the plots and figures are being generated as opposed to pulled from pre-existing files. Now everything is generated in real time.

1. Add explanations for the steps. This app is a teaching tool, so it should provide clarity for the steps involved in the process.

Demo

Things to consider in the future

1. Fix sound_generator.
    - Generated audio worked correctly in the google colab version of the code, for some reason it is not working correctly in streamlit.

1. Small bug fixes.

1. Add piano interactive keyboard so that user could click a key rather than uploaded a key .wav file

1. Wavelet transform
    - It would be interesting to add wavelet transform and a new neural network and compare the results of the two feature extraction processes.

Vision

- At the end of the 5 weeks, we envision that we will have a fully completed app that can clearly demonstrate the feature extraction process to modify the given sound input to the desired output sound.

- One of our main goals is to be able to develop this app as an educational tool to teach future students about the steps for the MDS project.

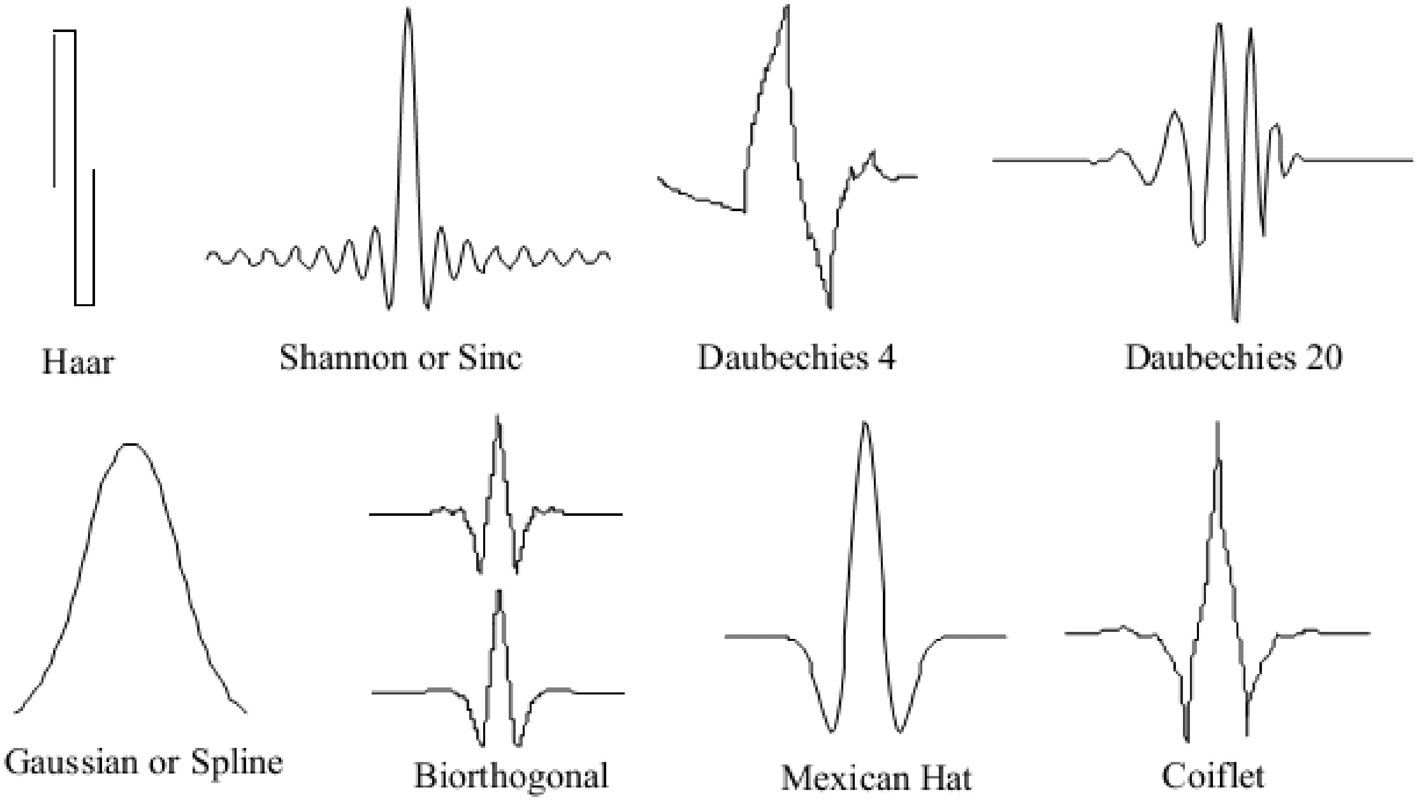
- Overall, we’ve accomplished our goals we set at the beginning of the five weeks.

References

- https://www.gaussianwaves.com/2015/11/interpreting-fft-results-obtaining-magnitude-and-phase-information/
- https://www.youtube.com/watch?v=QX1-xGVFqmw
- https://blog.streamlit.io/introducing-new-layout-options-for-streamlit/
- https://surendraredd.github.io/Books/concepts.html
- https://docs.streamlit.io/en/latest/api.html#streamlit.beta_set_page_config
- https://www.sjsu.edu/people/burford.furman/docs/me120/FFT_tutorial_NI.pdf
- https://ccrma.stanford.edu/~unjung/mylec/WTpart1.html
- Graps, A. “An Introduction to Wavelets.” *IEEE Computational Science and Engineering*, vol. 2, no. 2, 1995, pp. 50–61., doi:10.1109/99.388960.

Feature Extraction Techniques

Wavelet Transform



Haar

Shannon or Sinc

Daubechies 4

Daubechies 20

Gaussian or Spline

Biorthogonal

Mexican Hat

Coiflet