AUTOMATIC MUSIC CLASSIFICTION

Shankar Subramanian Ravichandran ID:1361168

Shrivatsa Neerkaje ID:1359812

Rebeca Tijiboy ID:1365129

INTRODUCTION

The amount of music available online grows every day. With the increase in number of songs and artists, finding music one likes has become more difficult. This project seeks to tackle the problem of classifying music into genres automatically. Such task will be done by finding the ideal features and conducting feature extraction in testing and training data sets. These values will then be used to train a classifier and to test the classifier's performance.

This project can be the basis of online and mobile applications that can help people sort through songs. It is difficult for one person to correctly sort thousands of songs into genres. This will enable, for example, an online music store to automatically classify their music for users to buy. In a mobile application users would be able to easily find music they like within their music library or identify the genre of any song they are listening to wherever they are.

DATA SET AND CLASSIFICATION OVERVIEW

90 songs were collected across classical, rock and jazz genres. Songs were selected from several different performers and were instrumental songs, no lyrics. Because songs are usually over 3 minutes in length, audio cutter software was used to cut the running times of each of the 90 audio files to 15 secs. This length was chosen because processing of audio files requires a lot of processing time and memory. The next task was to extract specific features from each audio clip. Several features which were believed to differ from genre to genre were chosen. After extensive research and various trials, the features selected were amplitude, centroid, pitch, filtered frequency and count above 0.4 as our extraction parameters. MATLAB was used for feature extraction from the audio clips. Output Value for Each of these parameters for each genre was collected as a Matlab variable and given as an input to the classifier. The classifiers were to sort music into classical, jazz and rock based on the value of these parameters. Each parameter is chosen in such way that, its output shows a clear distinction between one genre and the other two. The next step was to test classifiers to evaluate their performance using the combination of the selected features. More detail on the features used and the classifier will be given in the following sections.

FEATURES

The following features were carefully selected by analyzing their individual performance. The final features selected were count of amplitude spikes, spectral centroid, filtered frequency, magnitude spectrum, and variations in amplitude in the time domain. Many other features were tested but showed low performance like: RMS energy, kurtosis, phase and bandwidth.

AMPLITUDE SPIKES

The time domain distribution of the music genres vary according to the genres. As the instruments used in rock music are stronger than the rest, the spikes will have a higher amplitude.

Design:

1) All the audio signals containing different genres were imported into MATLAB using an inbuilt function called audioread() which normalizes the amplitude to a scale of [-1,1]
2) The number of spikes or signal components which are above a value of 0.4 were calculated.

Observation:

On observing Figures 1, 2, and 3, one can conclude that

a) The rock music has a higher number of spikes above the normalized amplitude of 0.4.
b) There is not much of a difference in the count of spikes greater than 0.4 between jazz and classical visually.

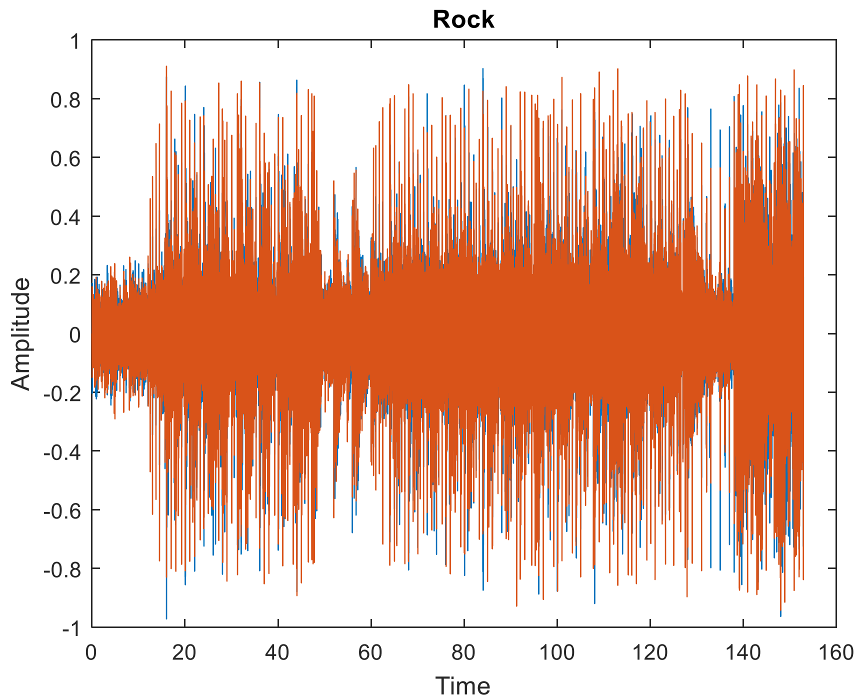

Fig. 1 Rock amplitude in the time domain plot.

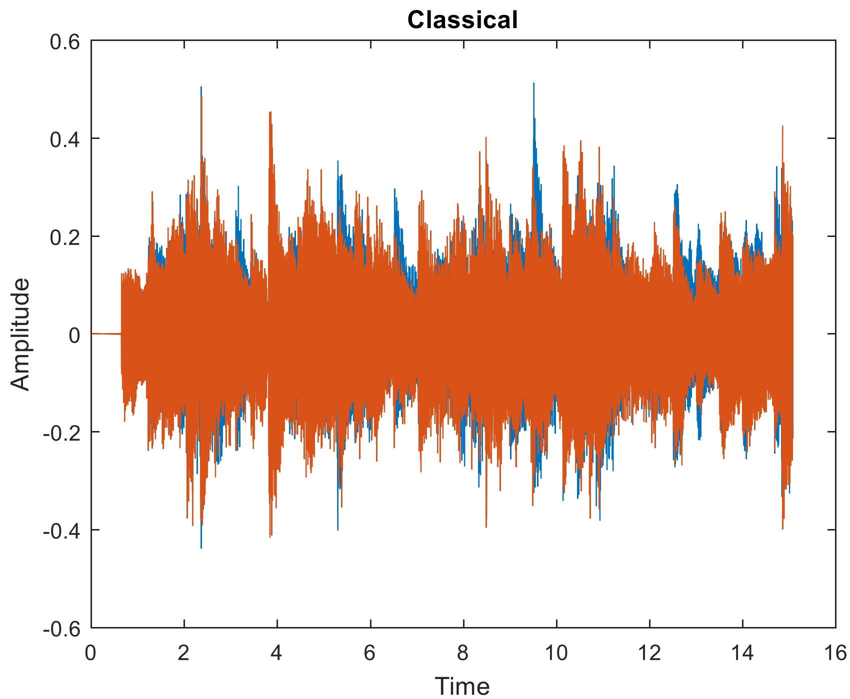

Fig. 2 Classical amplitude in the time domain plot.

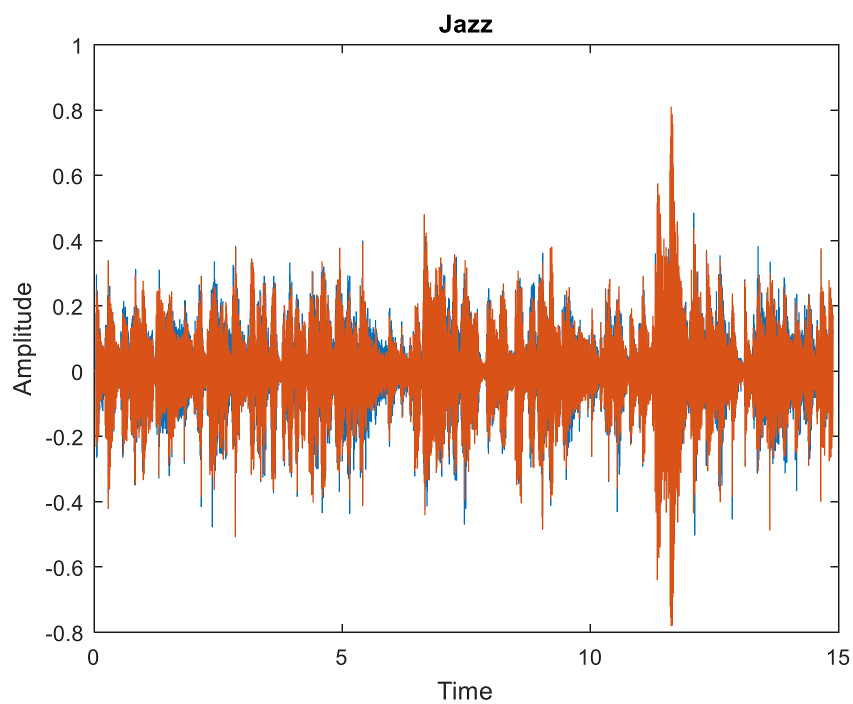

Fig. 3 Jazz amplitude in the time domain plot.

SPECTRAL CENTROID

This feature is adapted from the field of psychoacoustics and music cognition. It is evaluated in the frequency domain of the signal. It is defined as the averaged frequency weighted by the amplitudes divided by the sum of amplitude.

Design:

a) A Hamming window of a fixed window size was used. Then convoluted with the signal in time domain.
b) The centroid of each block was calculated using the formula 'sum (m.*F)/sum(F)' where m is center frequency and F is the weighted frequency value.
c) The centroids for all the frames were calculated.
d) The maximum value of the array of centroids was considered as the representative of spectral centroid for the song.

Observations:

a) The distinction between Rock and the other genres (Jazz and classical) was good.
b) The distinction between jazz and classical was poor.

FILTERED FREQUENCY

The frequency spectrum of the three genres vary widely. Generally the overtones produced by rock instruments cause the signals to have a wider frequency spectrum. Jazz, on the other hand has a narrower spectrum when compared to rock but wider than that of classical. Consider the following figures. On a normalized frequency scale, the frequency spectrum of a classical music stops at around 0.2 π (rad/sample). Similarly the spectrum of jazz music has frequencies up to a value of around 0.45 π (rad/sample) whereas the frequency spectrum of rock music has frequencies up to a value of 0.8 π (rad/sample). (See Figures 4, 5, and 6)

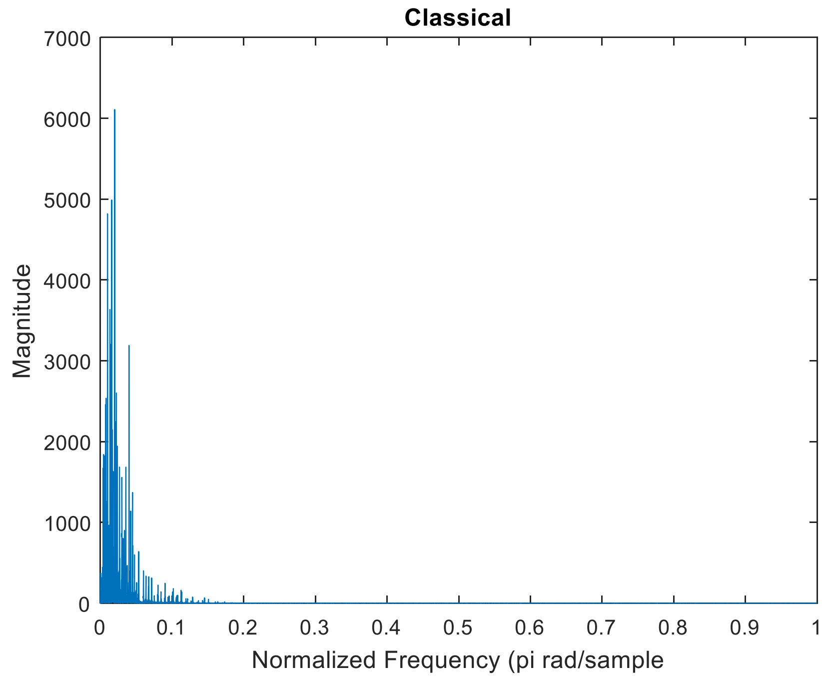

Fig 4. Frequency spectrum of Classical song

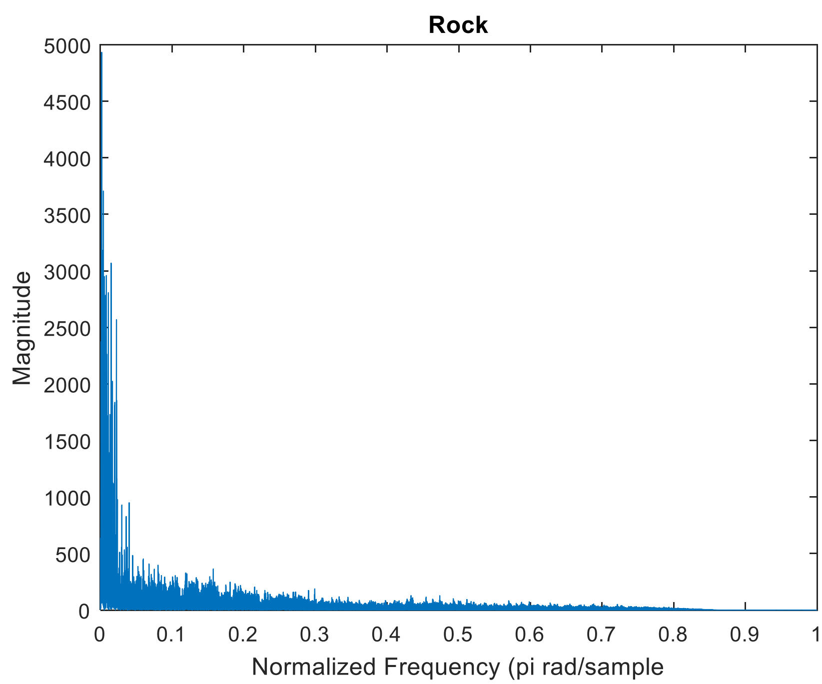

Fig 5. Frequency spectrum of Rock song

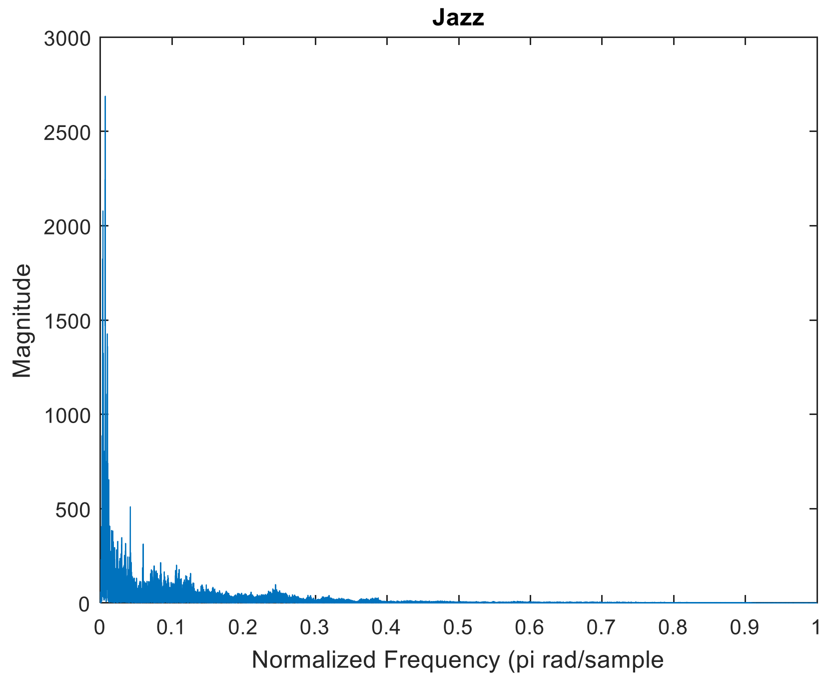

Fig 6. Frequency spectrum of Classical song

In order to get the exact numbers of the frequency components present in the various frequency bins, the music signals are to be passed through a high pass Butterworth filter.

BUTTERWORTH FILTER

A Butterworth filter is an IIR filter designed to have a flat frequency response. The cutoff frequency of the filter used was 0.25 π (rad/sample) because as seen in Figure 3 there is no frequencies above that value for classical songs. The order of the Butterworth filter is approximated to 2 as lower orders tend to have a gradual rise in the frequency response (See Figure 6). The gradual rise is preferred because it passes many of the frequency components present in range of 0.2 π to 0.4 π between which the crucial frequency components of jazz music are present.

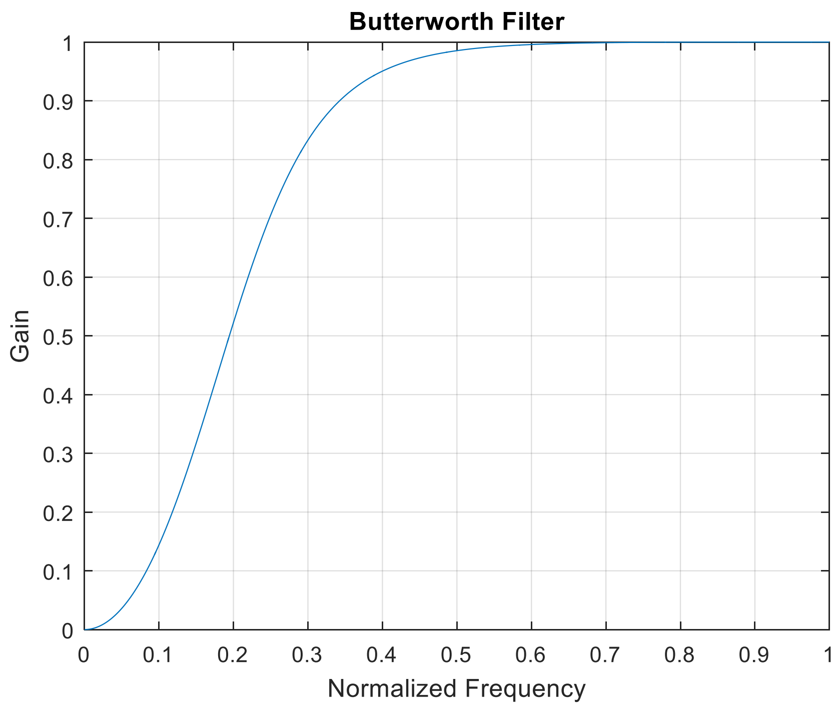

Fig 7. Butterworth Filter

<u>Design</u>:

a) The frequency domain of the music signals was constructed using the FFT algorithm.
b) A Butterworth filter of gain=1, order=2 and cut off frequency 0.25 pi was constructed.
c) The frequency spectrum of the music signals was passed through the high pass Butterworth filter.
d) The frequency spectrum of the output was evaluated.
e) The number of frequency components above 0.2 π (rad/sample) having an amplitude value of greater than 15 units was calculated and used as a feature.

<u>Observation</u>:

a) The count of frequency components greater than 0.2 pi(rad/sample) for classical music was almost 0 in all the samples.
b) The count of frequency components greater than 0.2 pi(rad/sample) for jazz music was in the range of thousands
c) The count of frequency components greater than 0.2 pi(rad/sample) for rock music was in the range of ten thousands
d) A clear distinction was created among the three genres viz:- classical, rock and jazz

MAGNITUDE SPECTRUM

The magnitude spectrum was obtained by performing a Fast Fourier Transform (FFT) algorithm on the audio signal. The magnitude spectrum gives an idea about the strength of each frequency component of the music signal. Most of the other features give a good differentiation between rock music and the other genres whereas this feature gives a good difference between classical and rest of the genres (rock, jazz) thereby compensating for the inefficiencies of the other features. From Figures 8, 9, 10 below it can be seen that the maximum frequency strength of classical songs is in the range of (3000-8000) units. The amplitude of rock music was in the range of (60k-80k) whereas for jazz music it was in the range of (11k – 80k)

Design:

a) The fourier transform of the signal was calculated with the help of FFT algorithm.
b) The amplitude value of the most dominant frequency was calculated and taken as a variable.

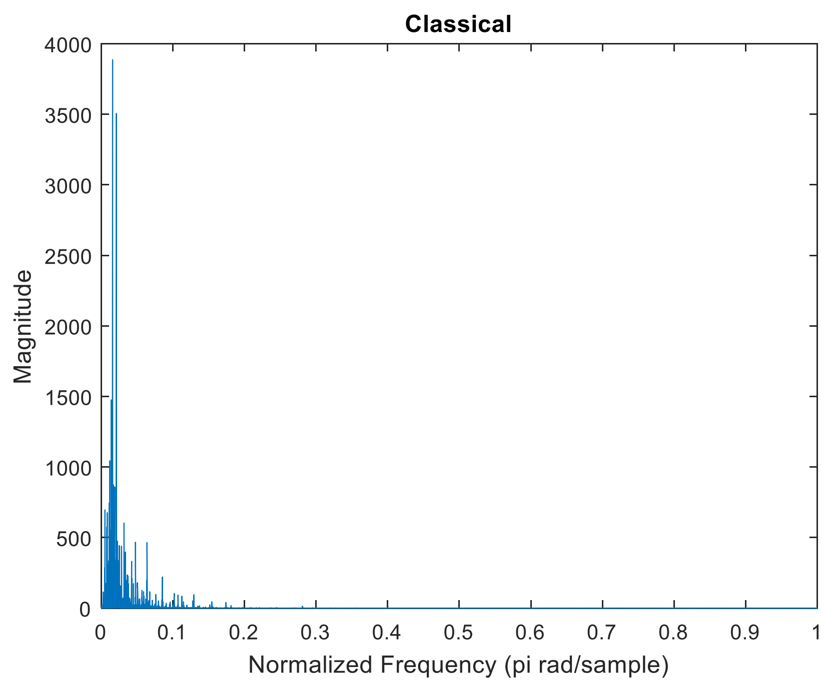

Fig. 8. Frequency of a Classical Song

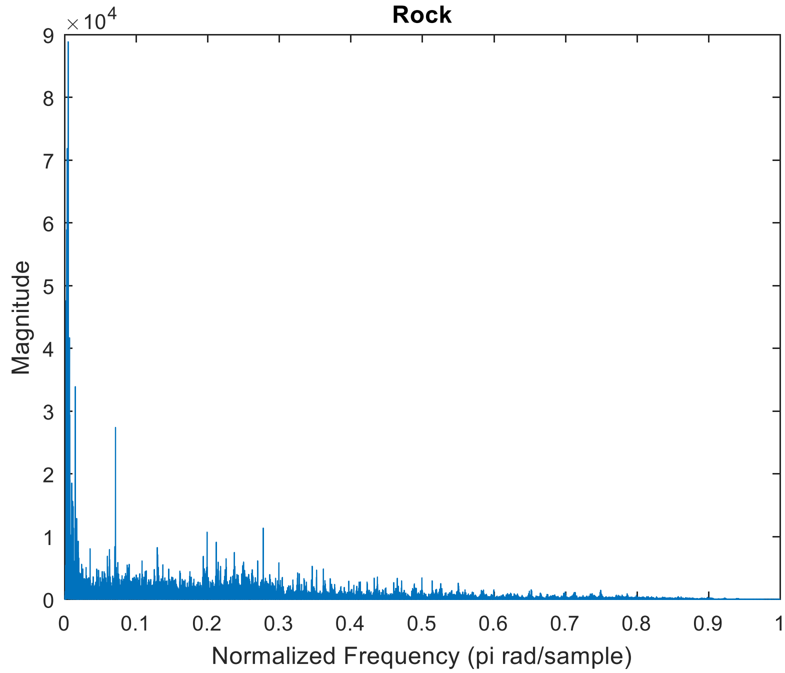

Fig. 9. Frequency of a Rock Song

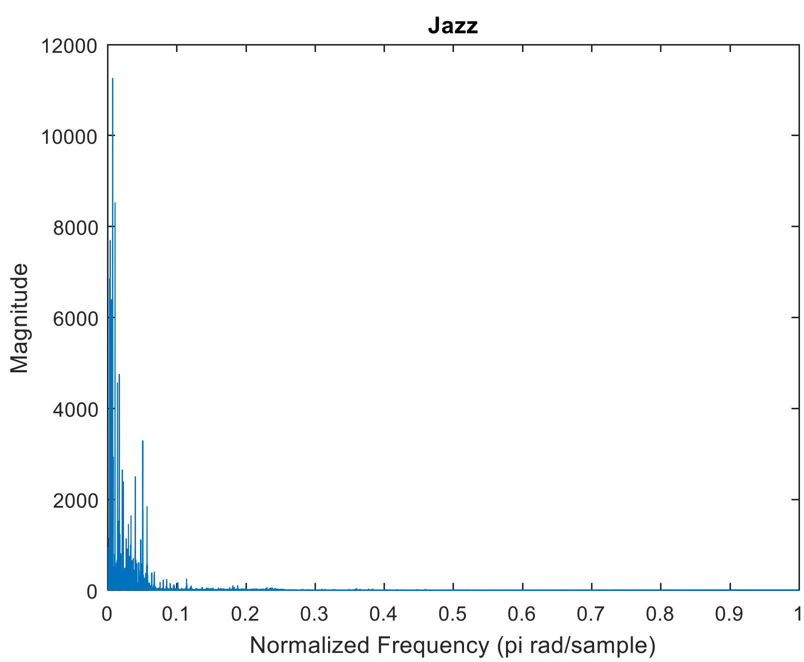

Fig. 9. Frequency of a Jazz Song

Observation:

a) The amplitude of rock and jazz are very high compared to classical thereby aiding the separation of classical from the rest.

VARIATION IN TIME DOMAIN

There is a widespread variation in the time domain distribution of the songs with respect to genres. For classical music (see Fig. 10), the amplitude increases gradually with time and decreases gradually. For rock music (See Fig. 11), the amplitude is very unstable and remains relatively high. For jazz music (see Fig. 12) there is sudden increases in amplitude which it decreases with time, then this cycle repeats.

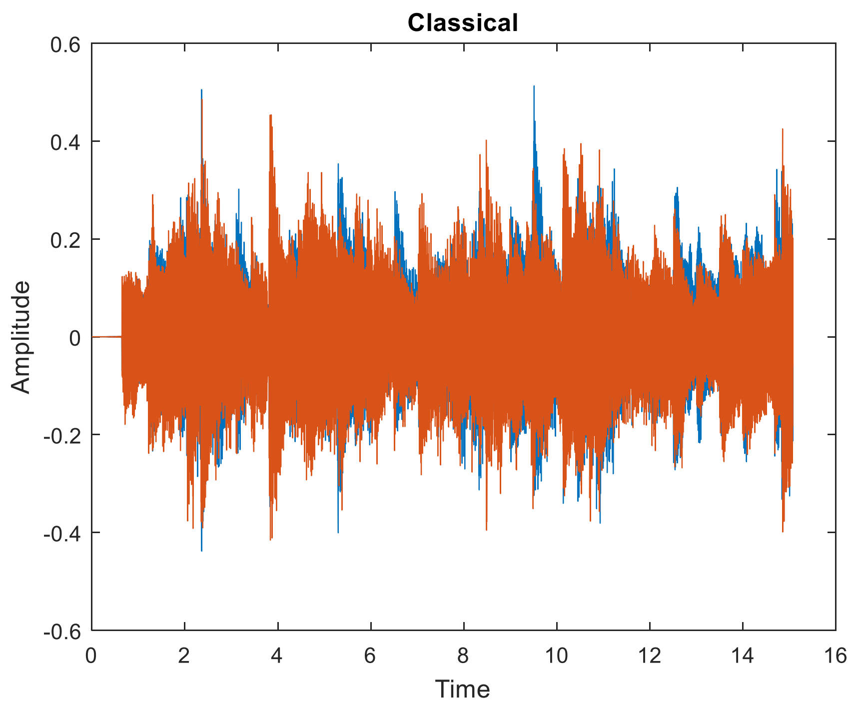

Fig. 10. Amplitude vs Time for a classical song

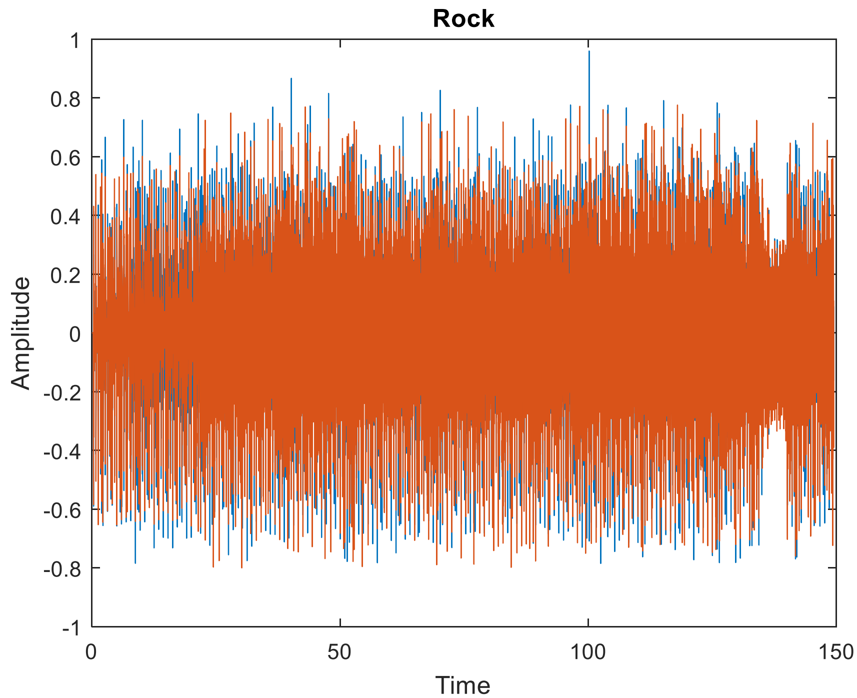

Fig. 11. Amplitude vs Time for a rock song

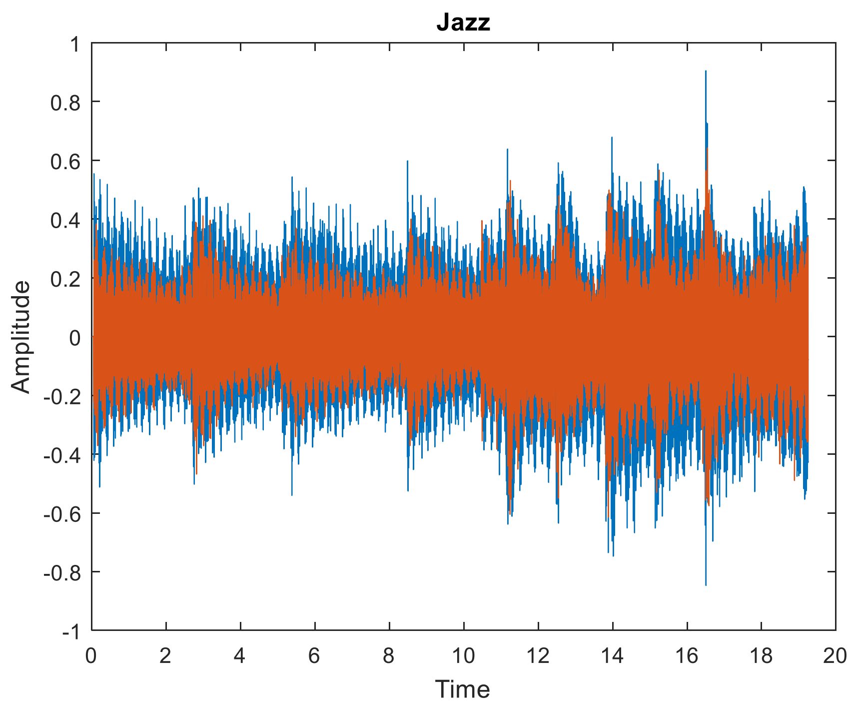

Fig. 12. Amplitude vs Time for a jazz song

Design:

a) The audio signals are imported to MATLAB using the command audioread().
b) The standard deviation of the signals was calculated and taken as a feature.

Observation:

a) A clear difference in the standard deviations of rock and other genres was seen.
b) The accuracy between classical and jazz music was around 75%

CLASSIFICATION

For this project we decided to test two different classifiers, Naïve Bayes and Neural Networks. The two classifiers were to be tested using the best features and compared to find which had the best performance. Neural network toolbox and Statistical toolboxes were used in Matlab to build the classifiers.

Naïve Bayes

Naïve Bayes classifies data by calculating the posterior probability of each data set based on the priori probability of each class. The data point is assigned to whichever class the posterior probability is greater for. The posterior probability is calculated using the following formula:

$$p(C_k|\mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})}.$$

Matlab was used to train and test the Naïve Bayes classifier. All five features previously discussed where used. Leave-one-out cross validation was used. Figure 13 shows the resulting confusion matrix.

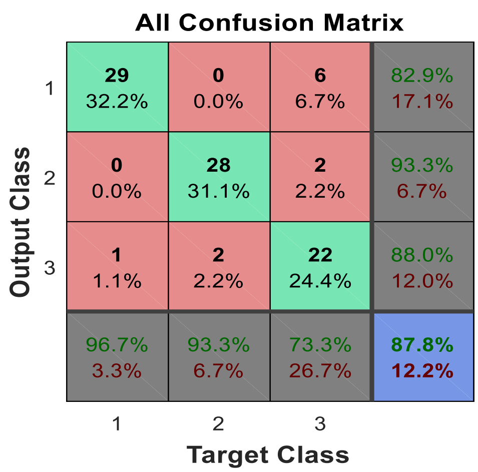

Fig. 13. Confusion Matrix from Naïve Bayes with Leave-one-out cross validation

The green boxes along the diagonal show the correctly classified audio clips. Class 1 represents classical music, class 2 is rock music, and class 3 is jazz music. The overall performance of the classifier is 87.8% correct.

Neural Network

A neural network is a nonlinear system which has hidden layers of neurons. The connections between neurons are weighted, these weights are assigned during training. There is one input for each feature used and one output for each class. In the used Neural Net there were five inputs and three outputs and the optimal number of hidden layers was found to be 20 (See Figure 14 for Neural Net structure). Supervised learning was used.

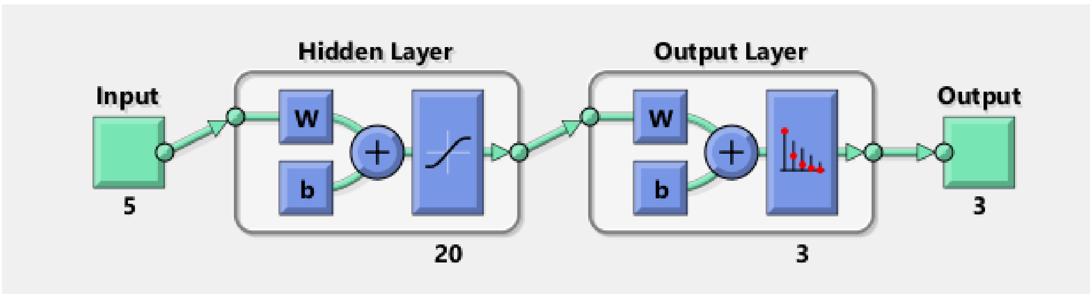

Figure 14. Structure used for Neural Network Classification.

The result of the classification can be seen in Figure 15. Again, the green boxes along the diagonal show the correctly classified audio clips. Class 1 represents classical music, class 2 is rock music, and class 3 is jazz music. The overall performance of the classifier is 91.1% correct.

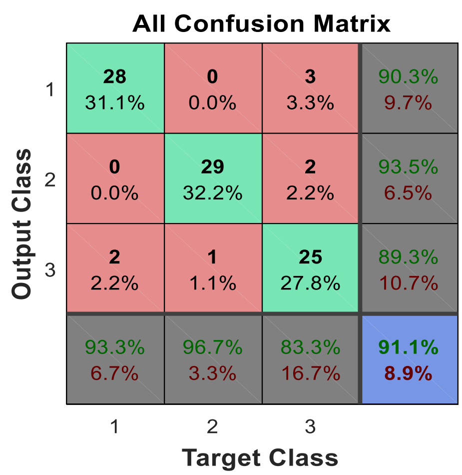

Fig. 15 Confusion Matrix result from Neural Network Classification

Conclusion

Music of all three genres, was classified efficiently. Neural Network had the best performance out of the two classifiers with 91% accuracy. The best features in classifying for classifying these three genres were amplitude spikes, spectral centroid, filtered frequency, magnitude spectrum, and variation in the time domain.

The same Naïve Bayes classifier was used to find the priori probability of fusion music, to help people figure out how much of each genre their fusion music is. However, this classifier would give every fusion song a priori probability of 1. The songs would be classified as being completely of a single genre.

Future work includes further exploration of fusion music classification. Also, implementation of the automatic classification into mobile applications and web applications making it available to users and to online vendors or publishers.