CS 529 CLASS PROJECT 3: MUSIC CLASSIFIER

Md Amanul Hasan[1]
Farhan Asif Chowdhury [2]

November 13, 2018

Abstract

In this class project, we have implemented three different classifiers with three different feature extraction methods to classify the music genres of the given GTZAN dataset. The implemented classifiers are Support Vector Machine (SVM), Artificial Neural Network (ANN) and Convolution Neural Network (CNN) respectively. Whereas, Fast Fourier Transform (FFT), Mel-Frequency Cepstrum Coefficients (MFCCs) and Octave-Based Spectral Contrast (OSC) are used as the feature extraction method. Firstly, 900 known data set was divided into 675 training and 225 validation subsets. Using these subsets, we evaluated effectiveness of each pair of classifier and feature. We also combined multiple features to improve the accuracies. Finally, Kaggle accuracy was evaluated using the testing data set. It is found that, SVM performed better than the ANN and CNN. This is might be due to small dataset. Among the features, MFCCs alone gives the maximum accuracies, however, we got highest accuracy after combining MFCCs, OSC and other spectral information. The maximum accuracy found from the Kaggle is 78.571% when we use MFCCs + OSC + other spectral information as the feature and SVM as the classifier.

Keywords: *Machine Learning, Music Classifier, FFT, MFCC, OSC, SVM, ANN, CNN, Accuracies.*

[1] Graduate student, Department of Civil Engineering, University of New Mexico, Albuquerque, email: amanulhasan@unm.edu

[2] Graduate student, Department of Computer Science, University of New Mexico, Albuquerque, email: fasifchowdhury@unm.edu

Both authors have contributed equally.

INTRODUCTION

Music genre classification is the task of identifying the genre of a song [1]. The primary goal of this project is to implement two different machine learning classifiers with three different feature selection techniques to classify genre of an unknown music. The given dataset is the GTZAN dataset which is frequently used to benchmark music genre classification task. It has a total of 1000 songs divided equally into 10 distinct genres: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock. The training data consists of the first 30 seconds of 90 songs per genre. The testing data consists of 100 unlabeled songs. Our task is to classify the unlabeled songs into 10 genres. The goal of the project is to extract individual frequency intensities from the raw sample readings and feed them into a classifier. Here we have used 3 feature extraction techniques: Fast Fourier Transform (FFT), Mel Frequency Cepstral Coefficients (MFCC) and Octave-Based Spectral Contrast (OSC) incorporated with Other information. The two classifier techniques that have been used are Support Vector Machine and Artificial Neural Network (ANN). Beyond the specific project requirements, we have also implemented Convolution Neural Network (CNN) here. Based on the various combination of features and classifiers, we have calculated the accuracies of the test data, compared the result and documented the best accuracy gained from the combination of feature extraction technique and classifier.

FEATURE EXTRACTION

Feature extraction is an important part of machine learning and according to the project requirements, three distinct features need to be employed in this project. Therefore, we used Fast Fourier Transform (FFT), Mel-Frequency Cepstrum Coefficients (MFCCs) and added Octave-Based Spectral Contrast (OSC) incorporated with Other information as the third one. However, our final proposed feature is the combination of MFCC, OSC and Other features.

Fast Fourier Transform (FFT)

The Fast Fourier Transform (FFT) is one of the most used algorithms that transforms a function of time into a function of frequency. Sometimes it is described as transforming from the time domain to the frequency domain. For example, it samples a signal over the time (or space) and divides it into its frequency components [1]. These frequency components are single sinusoidal oscillations at distinct frequencies each with their own amplitude and phase. This transformation is illustrated in Fig. 1.

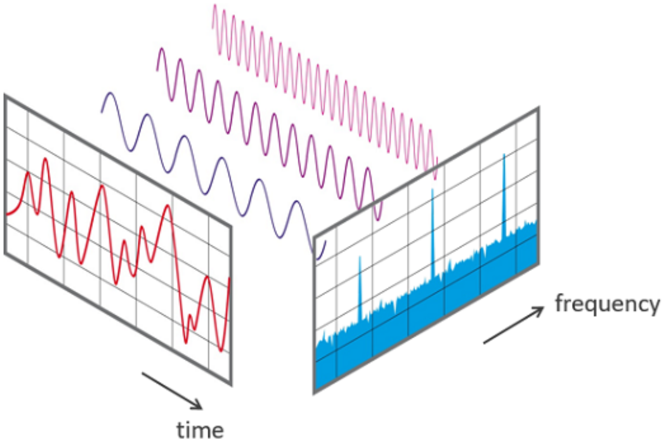


Fig. 1: FFT (Source: Wikipedia.org).

An FFT computes the Discrete Fourier Transform (DFT) and produces exactly the same result as evaluating the DFT definition directly; the most important difference is that an FFT is much faster. The mathematical representation of DFT is given in Eq. 1.

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \qquad (k = 0, \ldots\ldots, N-1) \tag{1}$$

where x_0, ..., x_{N-1} be complex numbers, *N* represents the number of points.

The FFT is one of the most powerful tools in digital signal processing. Because, it has made working in frequency domain equally computationally feasible as working in temporal or spatial domain.

Mel-Frequency Cepstrum Coefficients (MFCCs)

The Mel-Frequency Cepstrum Coefficients (MFCCs) are the short time power spectral representation of a sound signal, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. The Mel scale correlates perceived frequency of a pure tone to its actual measured frequency. The formula to convert frequency f hertz into Mel mf is given by Eq. 2. Inverse relationship is shown in Eq. 3. Fig. 1 shows the steps for determination of MFCCs.

$$m = 2595 * log_{10}(1 + {}^{f}/_{700}) \tag{2}$$

$$f = 700 * (10^{m/2595} - 1) \tag{3}$$

Since humans are much better at discerning small changes in pitch at low frequencies than they are at high frequencies, integrating this scale makes our features match more closely what humans hear. Therefore, the MFCC is one of the widely used features in automatic music classification and speech recognition. Past studies also showed that the MFCC is very effective in automatic speech recognition and in modeling the subjective frequency content of audio signals.

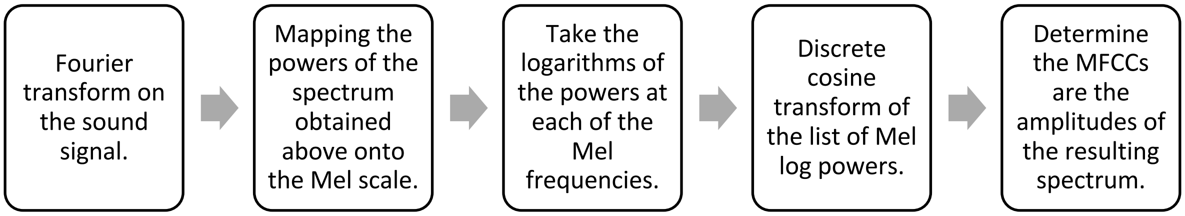

Fourier transform on the sound signal.
Mapping the powers of the spectrum obtained above onto the Mel scale.
Take the logarithms of the powers at each of the Mel frequencies.
Discrete cosine transform of the list of Mel log powers.
Determine the MFCCs are the amplitudes of the resulting spectrum.

Fig. 2: Determination of MFCCs.

First, we tried the average MFCCs over the frames and we got improve prediction accuracy over the FFT feature. After that, we added standard deviation of MFCCs over the frames and got further improvement.

Proposed Feature

Octave-Based Spectral Contrast (OSC)

While looking for the possible third feature for our project, we found from the literature that Octave-Based Spectral Contrast (OSC) is good for audio genre classification [2]. The OSC considers the spectral peak and valley in each sub-band separately and the difference between spectral peaks and spectral valleys will reflect the spectral contrast distribution. Because, the spectral peaks correspond to harmonic components and the spectral valleys correspond to non-harmonic components or noise in a music piece. Past studies showed that OSC can give an improve prediction accuracy for music genre, therefore, we employed the OSC as the third feature for our project.

A music piece is first decomposed into a number of overlapped frames and the spectrum of each frame is obtained by FFT. After that, the spectrum of each frame is then divided into a number of sub-bands by the octave scale filters. The discrete spectrum of each sub-band of each frame is then sorted in a descending order to determine the peaks and the valleys. Instead of taking one single peak (or valley), average of few neighboring peak (or valley) is taken. The flow chart is shown in Fig. 3.

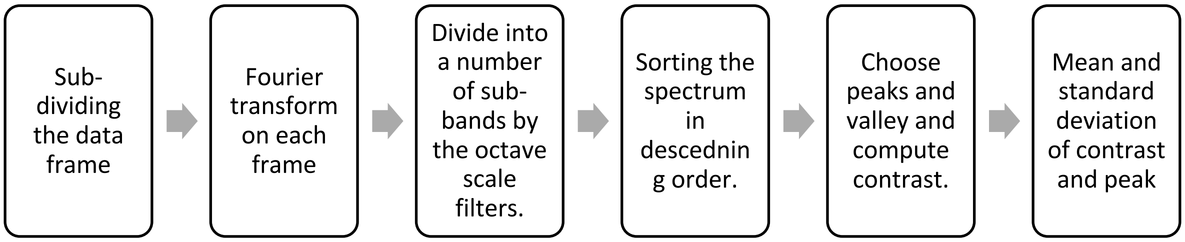

Sub-dividing the data frame
Fourier transform on each frame
Divide into a number of sub-bands by the octave scale filters.
Sorting the spectrum in descednin g order.
Choose peaks and valley and compute contrast.
Mean and standard deviation of contrast and peak

Fig. 3: Determination of OSC parameters.

If $x_{b,1}$, $x_{b,2}$,...., x_{b,N_b} represent the spectrum of the b-th sub-band, N_b is the total number of FFT frequency bins in the b−th sub-band, the spectrum is then sorted in a descending order represented as ($x_{b,d(1)}, x_{b,d(2)}, ...,x_{b,d(N1)}$) where $x_{b,d(1)} \geq x_{b,d(2)} \geq \geq x_{b,d(Nj)}$. The spectral peak and the spectral valley in the b-th sub-band is estimated as follows:

$$Peak_b = \log\left(\frac{1}{\alpha N_b} \sum_{i=1}^{\alpha N_b} \mathrm{x}_{\mathrm{b,d(i)}}\right) \tag{4}$$

$$Valley_b = \log\left(\frac{1}{\alpha N_b} \sum_{i=1}^{\alpha N_b} \mathrm{x}_{\mathrm{b,d(N}_b\mathrm{-i+1)}}\right) \tag{5}$$

where α is a neighborhood factor (chosen as 0.02). Finally, the spectral contrast between the spectral peak and the spectral valley of the b-th sub-band is given by:

$$Contrast_b = Peak_b - Valley_b \tag{6}$$

To represent the whole music piece, mean and standard deviation of the spectral contrast and spectral peak of all frames are used as the spectral contrast features. In this project, the number of sub-bands B is chosen as 8. Therefore, the feature dimension of OSC is 4 * B = 32. Table 1 shows the used frequency band in our project (followed by [2]).

Table1: Used frequency band

Band	Lower Frequency (Hz)	Upper Frequency (Hz)
1	0	100
2	100	200
3	200	400
4	400	800
5	800	1600
6	1600	3200
7	3200	8000
8	8000	22050

Incorporation of Other Spectral Information

In addition to OSC, we added other possible information that can be easily extracted from the audio signal such as mean and standard deviation of the chromagram, the root-mean-square energy (RMSE), the centroid, the spectral bandwidth, the roll-off frequency and the zero-crossing rate (ZCR) of audio signals []. Furthermore, it is known that the FFT, MFCC or OSC feature represent the short-term feature, therefore, we planned to add a long-term feature (tempo) as a feature of our data.

Final Proposed Feature

From our analysis, we have found that each feature has limitation to predict one or more genres. We have also observed that trouble genres for each feature are different, thus, we planned to apply combination of features (instead of a single feature). Literature also shows that that combining multiple features can improve over the performance of classification using a single feature type [3]-[6]. Finally, our proposed feature is the combination of MFCC, OSC and other spectral features. We avoided the FFT while combining the features, because, it gives lower accuracy.

CLASSIFIERS

As per the project requirement, we used two different classifiers in this project. The first one is the Support Vector Machine (SVM) and other is the Neural-Network (NN). In addition, we have also employed the Convolution Neural Network (CNN) in this project.

Support Vector Machine (SVM)

The Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. It is a supervised learning machine which uses the given labeled training data, the algorithm outputs an optimal hyperplane which categorizes new examples. In two-dimensional space, this hyperplane is a line dividing a plane in two parts where in each class lay in either side. It consists of picking a loss function, and then minimizing our cost function on the training data. Understanding this, the idea is quite simple: consider a specific choice of loss function being the Hinge loss for a single training example x as shown in Eq. 7. The cost function is then straightforwardly computed from there.

$$l(x) = max\{0, 1 - x\} \tag{7}$$

The SVM's are very good when there is no idea on the data, because, it works well with even unstructured and semi structured data like signal, text, images and trees. Furthermore, there are few advantages of SVM classifier. Firstly, it applies a regularization parameter to avoiding over-fitting. Secondly, it uses the kernel to transform the data to easily determine the hyperplane. In addition, an SVM is defined by a convex optimization problem (no local minima) for which there are efficient methods (e.g. SMO). Moreover, it approximates a bound on the test error rate. Considering all these advantages, we chose the SVM classifier for our project.

Artificial Neural Network (ANN)

Our second classifier is the Artificial Neural Network (NN). The ANN are computing systems vaguely inspired by the biological neural networks that constitute human brains [1]. The ANN itself is not an algorithm, but rather a framework (as shown in Fig. 4) for many different machine learning algorithms to work together and process complex data inputs. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules.

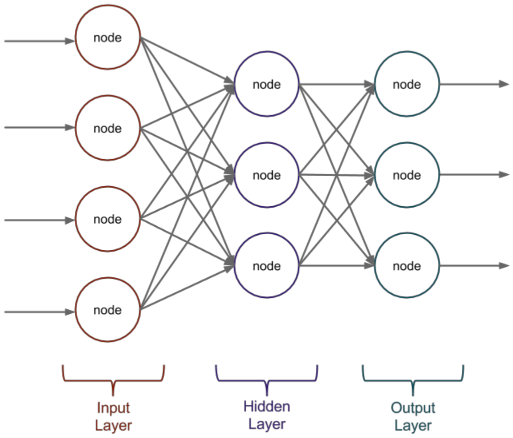


Fig. 4: Schematic of Neural Network.

Like the SVM, the ANN also has few advantages those motivates us to choose it for our project. It is known that the ANN can handle most of the datatype and it very prone to customization, this method seemed to be a good choice. As it can store the entire network information, after ANN training, the data may produce output even with incomplete information and the loss of performance here depends on the importance of the missing information. Therefore, we assumed that the ANN should be suitable for our project.

Convolutional Neural Network (ANN)

The convolutional neural network (CNN) is a class of deep, feed-forward artificial neural networks, most commonly applied to analyzing visual imagery. The most fundamental advantage of a convolutional neural network is automatic feature extraction for the given task; provided that the input can be represented as a tensor in which local elements are correlated with one another. Fig. 5 shows the visual representation of used CNN in this project. We chose the optimizer: Adam regularization: L2 norm regularization, activation function: Rectified Linear Unit, Cost Function: Softmax entropy, learning rate: 0.001, batch size: 512 and drop out: 0.4. Fig. 6 shows the Architecture for CNN.

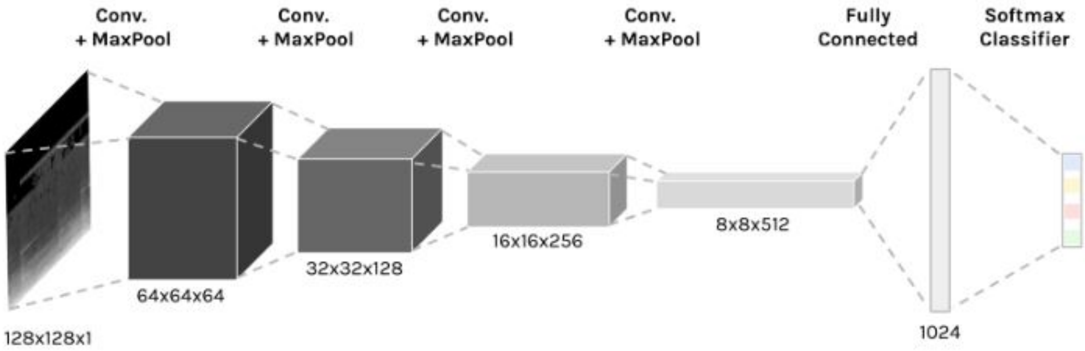


Fig. 5: Visual representation of used CNN.

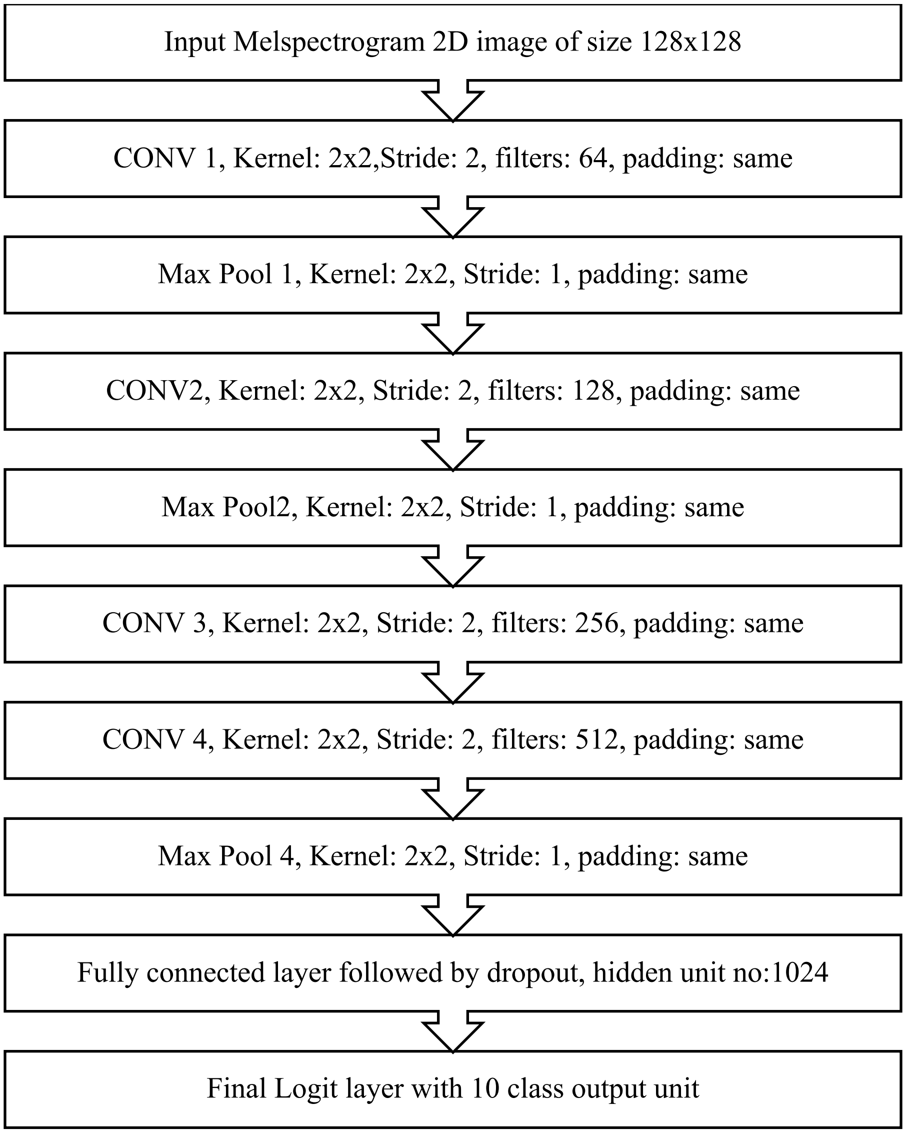

Input Melspectrogram 2D image of size 128x128
CONV 1, Kernel: 2x2,Stride: 2, filters: 64, padding: same
Max Pool 1, Kernel: 2x2, Stride: 1, padding: same
CONV2, Kernel: 2x2, Stride: 2, filters: 128, padding: same
Max Pool2, Kernel: 2x2, Stride: 1, padding: same
CONV 3, Kernel: 2x2, Stride: 2, filters: 256, padding: same
CONV 4, Kernel: 2x2, Stride: 2, filters: 512, padding: same
Max Pool 4, Kernel: 2x2, Stride: 1, padding: same
Fully connected layer followed by dropout, hidden unit no:1024
Final Logit layer with 10 class output unit

Fig. 6: CNN Architecture used in this project.

ALGORITHMS STEPS

Datasets

At first, we splitted the dataset into two subsets: one subset is used as the training data (675 songs) and another one is used as the validation data (225 songs).

Methods

The training data is then used to train the algorithms. Validation data was used to adjust the different model parameters. Finally, trained algorithms are used to classify the testing data. The straightforward algorithms are shown in Fig. 7.

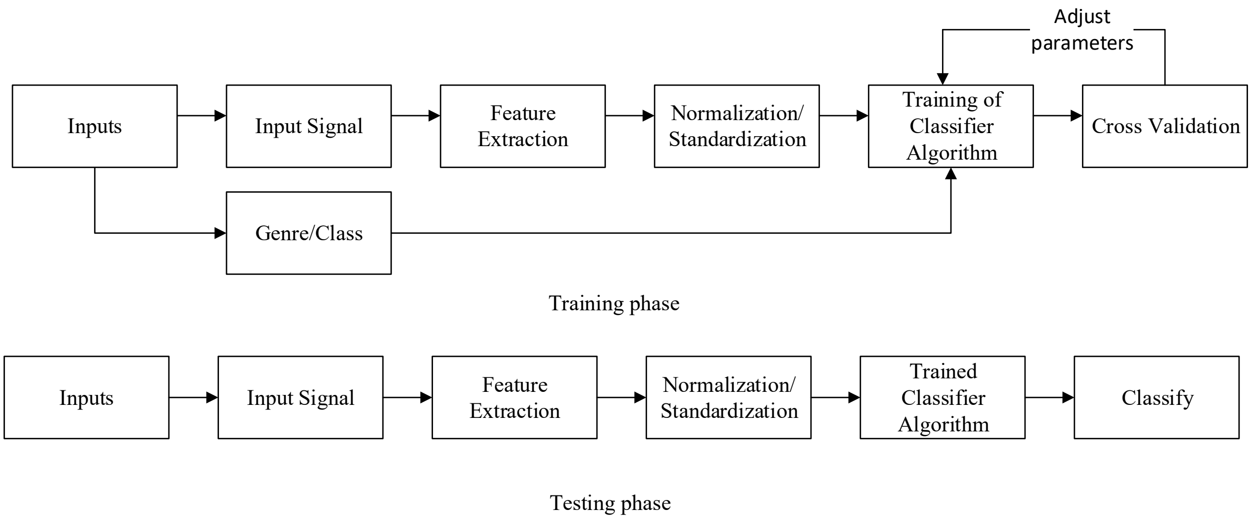


Fig. 7 Methodology

Feature Scaling

Feature scaling is used to standardize the features of the data into the range of 0 to 1. There are two types of feature scaling technique are being used: normalize or standardize the features extracted. We calculated accuracy for using both scaling features on the data and standardization provided us with better results.

Normalization

The formula for this method is the following:

$$x' = \frac{x - min(x)}{max(x) - min(x)} \tag{8}$$

where x is the original feature, min(x) is the minimum value, max(x) is the maximum value and x' is the scaled or normalized feature.

Standardization

The formula for this method is the following:

$$x' = \frac{x - \mathrm{avg}(x)}{\mathrm{std}(x)} \tag{9}$$

where x is the original feature, avg(x) is the mean or average value of the column, std(x) is the standard deviation and x' is the scaled or normalized feature.

RESULTS & ACCURACIES

In this section, we are going to discuss the results we have got from the implementation of the project.

10-Fold Cross Validation

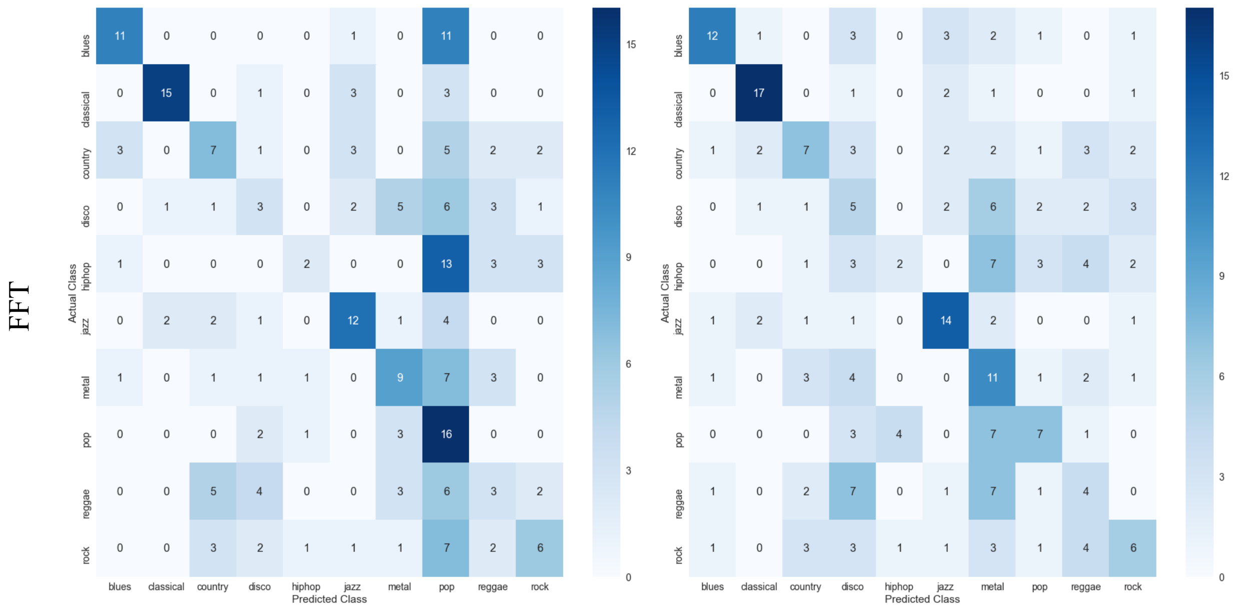
We have summarized the 10-fold cross-validation accuracies of different combinations by varying the features and classifiers in Table 2. Cross validation is a technique to evaluate learning models. In this technique, the training data is splitted into two samples, where one sample is used as the training data (675 songs) and one sample is used as the validation data (225 songs). This validates the method as we have a ground truth for the validation data here as it is already labeled. It shows that we got lower accuracies for ANN classifier than the SVM classifier. This is might be due to the smaller sample size of training data. Another reason of lower accuracies of ANN might be that we failed to choose appropriate ANN algorithms. While comparing the accuracies, we found that FFT feature performed the worst among the other feature (accuracy 33.92%). On contrary, the MFCC_Avg_Std itself performed better (accuracy 72.17%) than the FFT and OSC (accuracy 67.99%). However, we got highest accuracy (accuracy 73.33%) for the combined features. For the CNN with 2-D Melspectrogram, we got accuracy around 62.54%.

Table 2: 10-fold accuracies

Feature	Classifier	Accuracy (%)
FFT	SVM	33.92
FFT	ANN	33.28
MFCC_Avg	SVM	61.31
MFCC_Avg	ANN	41.00
MFCC_Avg_Std	SVM	72.17
MFCC_Avg_Std	ANN	55.03
OSC	SVM	67.99
OSC	ANN	48.50
Proposed=OSC+Other	SVM	64.00
Proposed=OSC+Other	ANN	54.65
Final Proposed = MFCC+OSC+Other	SVM	73.33
Final Proposed = MFCC+OSC+Other	ANN	60.13
2D Melspectrogram	CNN	62.54

Confusion Matrices

Figs. 8-10 show the confusion matrix for individual features. It is found that each feature has limitation to predict one or more genres. For example, FFT failed to provide satisfactory result for “country”, “disco”, “reggae” and “rock” genres. Similarly, MFCC_Avg_Std failed to provide satisfactory result for “rock” genre. On the other hand, OSC have improved performance for “rock” genre and lower performance for “disco” genre. Fig. 11 shows the confusion matrix after combining the features.



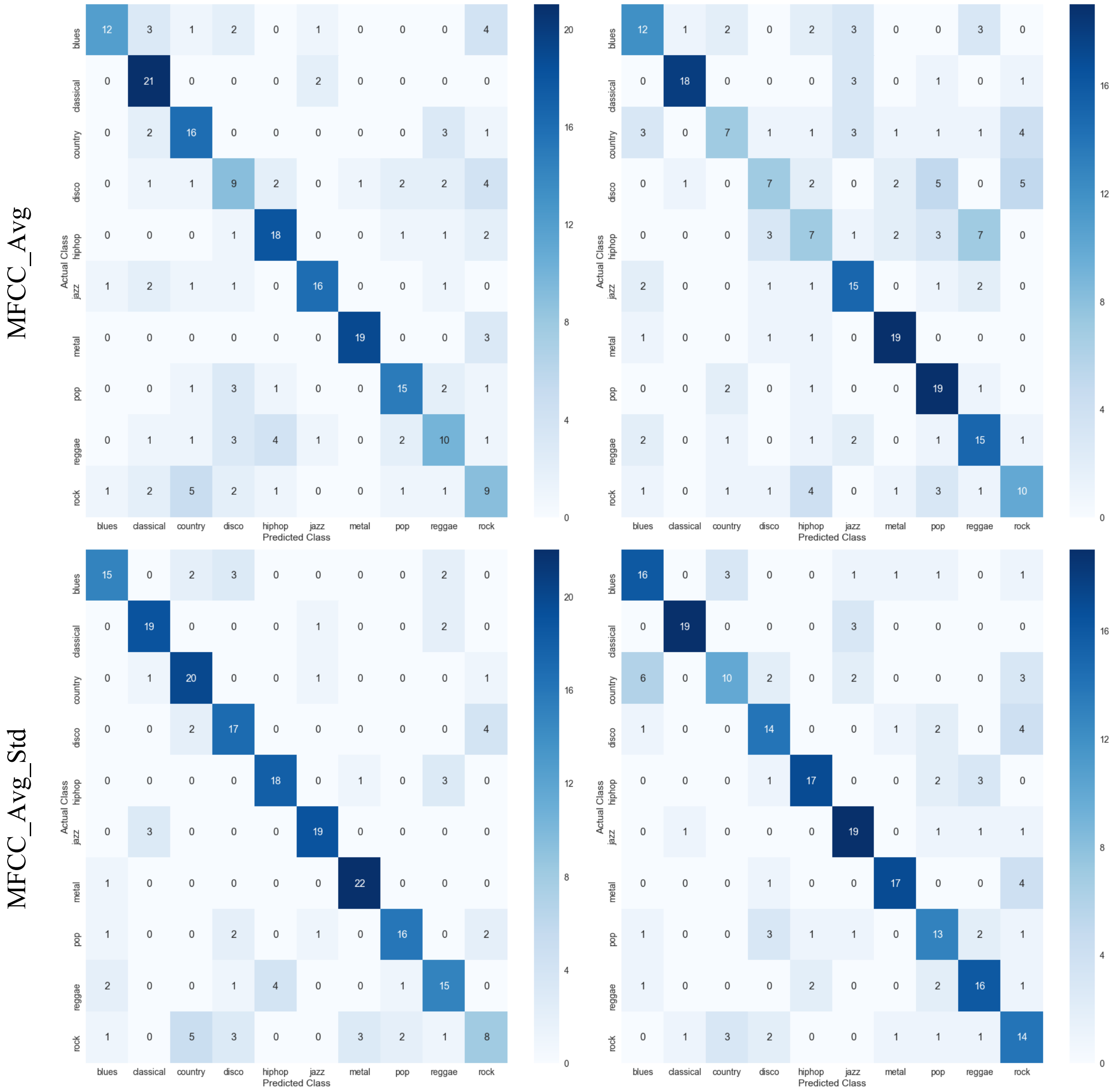
a) SVM

	blues	classical	country	disco	hiphop	jazz	metal	pop	reggae	rock
blues	11	0	0	0	0	1	0	11	0	0
classical	0	15	0	1	0	3	0	3	0	0
country	3	0	7	1	0	3	0	5	2	2
disco	0	1	1	3	0	2	5	6	3	1
hiphop	1	0	0	0	2	0	0	13	3	3
jazz	0	2	2	1	0	12	1	4	0	0
metal	1	0	1	1	1	0	9	7	3	0
pop	0	0	0	2	1	0	3	16	0	0
reggae	0	0	5	4	0	0	3	6	3	2
rock	0	0	3	2	1	1	1	7	2	6

b) ANN

	blues	classical	country	disco	hiphop	jazz	metal	pop	reggae	rock
blues	12	1	0	3	0	3	2	1	0	1
classical	0	17	0	1	0	2	1	0	0	1
country	1	2	7	3	0	2	2	1	3	2
disco	0	1	1	5	0	2	6	2	2	3
hiphop	0	0	1	3	2	0	7	3	4	2
jazz	1	2	1	1	0	14	2	0	0	1
metal	1	0	3	4	0	0	11	1	2	1
pop	0	0	0	3	4	0	7	7	1	0
reggae	1	0	2	7	0	1	7	1	4	0
rock	1	0	3	3	1	1	3	1	4	6

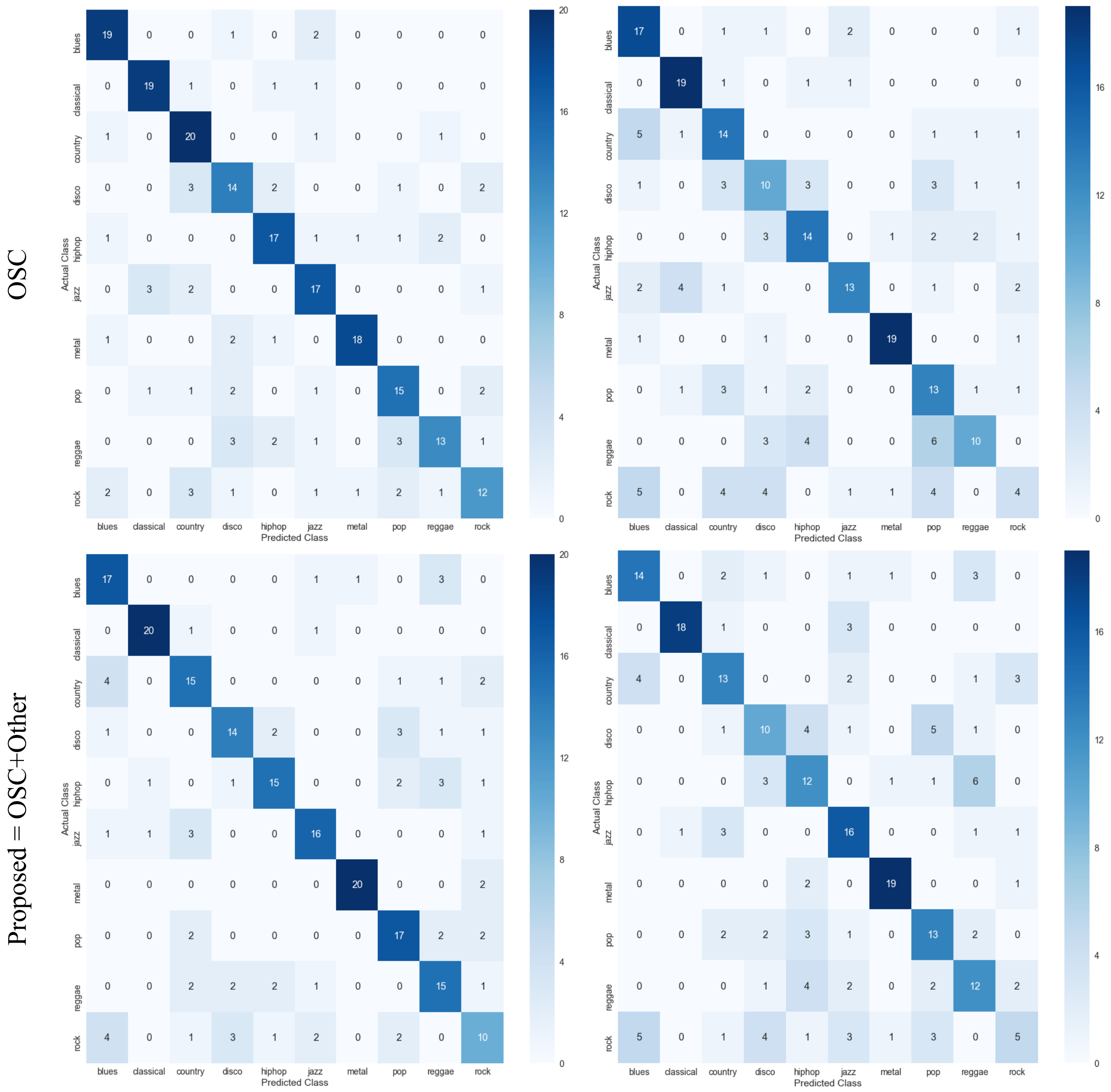
Fig. 8 Confusion Matrix, Feature: FFT



a) SVM

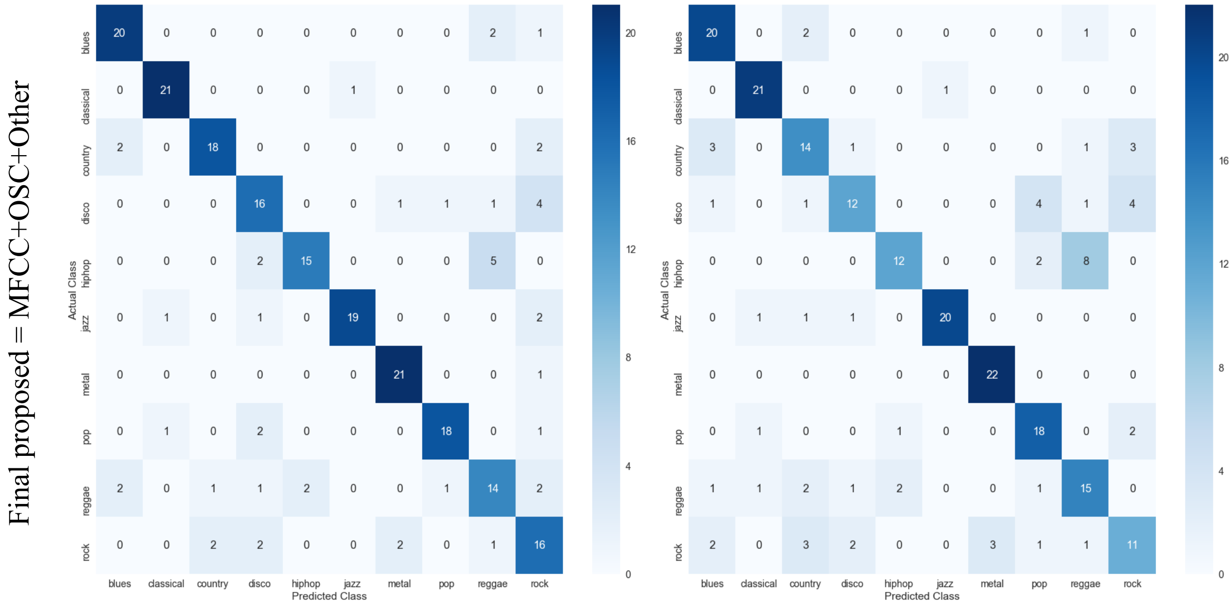
b) ANN

Fig. 9 Confusion Matrix, Feature: MFCC



a) SVM b) ANN

Fig. 10 Confusion Matrix, Feature: Proposed Third Feature



a) SVM b) ANN

Fig. 11 Confusion Matrix, Feature: Final Combined Feature

REFERRENCES

In this class project, we have implemented three different classifiers with three different feature extraction methods to classify the music genres of the given GTZAN dataset. The implemented classifiers are Support Vector Machine (SVM), Artificial Neural Network (ANN) and Convolution Neural Network (CNN) respectively. Whereas, Fast Fourier Transform (FFT), Mel-Frequency Cepstrum Coefficients (MFCCs) and Octave-Based Spectral Contrast (OSC) incorporated other information are used as the feature extraction methods. Firstly, 900 known data set was divided into 675 training and 225 validation subsets. Using these subsets, we evaluated effectiveness of each pair of classifier and feature. We also combined multiple features to improve the accuracies. Finally, Kaggle accuracy was evaluated using the testing data set. It is found that, SVM performed better than the ANN and CNN. This is might be due to small dataset or choosing wrong network algorithms or initialization. Among the features, MFCCs alone gives the maximum accuracies, however, we got highest accuracy after combining MFCCs and OSC. The maximum accuracy found from the Kaggle is 78.571% when we use combined features and SVM as the classifier.

REFERRENCES

[1] www.wikipedia.org

[2] C-H. Lee, J-L. Shih, K-M. Yu, and J-M Su, "Automatic Music Genre Classification using Modulation Spectral Contrast Feature," Proc. ICME 07, 2007.

[3] Z. Fu, G. Lu, K. Ting, and D. Zhang, "On feature combination for music classification," in Proc. Int. Workshop Statistical Pattern Recognition, 2010.

[4] Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Trans. Speech and Audio Processing 10(5), 293–302, 2002.

[5] Bergstra, J., Casagrande, N., Erhan, D., Eck, D., Kegl, B.: Aggregate features and ada boost for music classification. Machine Learning 65(2-3), 473–484, 2006.

[6] Lee, C.H., Shih, J.L., Yu, K.M., Lin, H.S.: Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features. IEEE Trans. Multimedia 11(4), 670–682, 2009.