Music Genre Classification Using Novel Song Structure Derived Features

Rustem Ozakar
Department of Computer Engineering
Erzurum Technical University
Erzurum, Turkey
rustem.ozakar@erzurum.edu.tr

Eyup Gedikli
Department of Software Engineering
Karadeniz Technical University
Trabzon, Turkey
gediklie@ktu.edu.tr

***Abstract*—Rapid grow of the digital music content and service providers worldwide everyday increases the importance of music genre classification. Most genre classification still relies heavily on human effort. Signal processing combined with machine learning methods aims to solve this problem autonomously for decades. In this work, we introduce novel high-level features derived from song structures and examine their performance through both CNN and a Voting Classifier. Results show that these features alone increases the classification accuracy significantly compared to random prediction and has potential of use in combination with other various features.**

Keywords*—*Music genre classification, music information retrieval, feature extraction, machine learning

I. Introduction

Music has became a large part of peoples' lives owing to internet and easier access to music streaming services and content providers. Users are recommended with songs which they may like based on their preferences. These recommendations often use similarity by making use of music genres. With millions of songs and dozens of genres, this task can easily become overwhelming.

Application of digital signal processing and machine learning methods to the music has created the field of music information retrieval (MIR). It is still a challenge to classify music genres autonomously because of several reasons. There are no strict boundaries between some music genres and subgenres within the same genre makes the task even more difficult Audio analysis of the songs (both in time and frequency domain) often gives no guaranteed evidence of a specific genre. This leads the researchers to extract more defining features to correctly classify music genres.

In this work, we define a new set of features derived from the song structure and measure the performance through two classifiers In first method, we use a Convolutional Neural Network (CNN). Song structures of four different genres are generated, preprocessed and then given into a CNN. Second method makes use of a Voting Classifier based on derived features from the song structures.

The organization of this paper is as follows; related works and existing methods in this field are discussed in Section II. In Section III, proposed methods are described in detail. In Section IV, experimental results are given and the Section V covers the conclusion and possible future works.

II. Related Work

Music genre classification is a vast field of research combining machine learning and digital signal processing. There are many approaches on the subject, from analysing spectrogram features to examining the lyrical content. Common methods that are used for classification includes k-NN (k Nearest Neighbours), GMM (Gaussian Mixture Models), SVM (Support Vector Machines), AdaBoost, LDA (Linear Discriminant Analysis), CNN and RNN (Recurrant Neural Networks).

Some examples of the works in this field can be summarized as follows; Pye [1] used GMM and tree based vector quantization methods on MFCC (Mel-Frequency Cepstral Coefficient and MP3CEP features to classify genres. Tzanetakis and Cook [2] used timbral texture, rhythmic content and pitch content features with GMM and k-NN classifiers Li et al. [3] proposed a new feature extraction method namely DWCH (Daubechies Wavelet Coefficient Histograms), and compared it with other features using classifiers like SVM, LDA, GMM and k-NN. Lidy and Rauber [4] introduced two new features namely statistical spectrum descriptors and rhythm histogram and measured the performance using SVM classifiers Cataltepe et al. [5] used MIDI data along with LDA classifier and k-NN. Meng et al. [6] proposed a multivariate autoregressive feature model and used generalized linear model for classification. Panagakis et al. [7] used multiscale spectro-temporal modulation features with SVM. Lee et al. [8] used long-term modulation spectral analysis of spectral and cepstral features using LDA as classifier. Costa et al. [9] treated spectrograms as textures and choose SVM as classifier. They extracted features using LBP (Local Binary Pattern) method. Nanni et al. [10] proposed using both visual and acoustic features together with SVM classifiers Senac and Pellegrini [11] proposed using eight different features derived from dynamics, timbre and tonality with CNN. Bahuleyan [12] compared the performance of various existing classifiers with various features. Durdag and Erdogmus [13] used various methods like Short Time Fourier Transform and Discrete Cosine Form to transform audio signals into coloured images for classification

There are two well-known databases for music analysis, GZTAN [2] and FMA [14]. These databases consist of 30 second samples as well as full length songs from different genres. In this study, we chose to build our own dataset because our work is focused on full length songs rather than short samples. This dataset consists of 2786 full length songs from four different genres in MP3 format. Dataset details provided in Section III.

III. Proposed Method

In this work, our proposed features are derived from song structures. Often, some music genres have common song

978-1-7281-7565-2/20/$31.00 ©2020 IEEE

structures that are distinct from other genres. For example, classic music song structures are generally expected to be more complex than a pop song's. To extract a song structure, Librosa [15], a Python library is used. Librosa is a library which allows to perform common digital signal processing operations on audio file as well as extracting features such as mel-spectrogram, tempo and such. McFee and Ellis [16] introduced a method which composes given song to clusters based on it's spectrogram similarities via recurrence matrix with Librosa. On top of this approach, laplacian segmentation can be performed to generate song structure. An example of this segmentation can be seen in [17].

A song's structure is the similar sections (for example it can be chorus, verse, etc.) within the song represented by same colors and labels. A song is transformed into a color represented PNG (Portable Network Graphics) file through the mentioned laplacian segmentation method. In addition, we present a separate CSV (Comma Seperated Values) file generated to resemble the song's structure in numerical data including tempo and utempo calculated by Librosa. A Python script is written to traverse through all songs categorized by four genre folders and generate song structures which are also categorized by genre. A total CSV file is saved to disk for every genre containing all song structure information for feature extraction and analysis. An example of an actual song structure in CSV and PNG format is shown in Table I and Fig. 1. Librosa parameters used for generating song structures are given in Table II. To determine the cluster count, mean shift algorithm is used with the eigenvector count to given in Table II. To prevent unnecessary sections, a minimum threshold of 7 seconds is chosen. Any section shorter than 7 seconds merged with previous section during structure generation.

TABLE I
EXAMPLE STRUCTURE OF A SONG IN CSV

section	**length**
6	10,89
0	39,89
1	16,27
0	38,61
1	39,86
3	20,59
0	39,03
1	33,62
135,99 **(tempo)**	143,55 **(utempo)**
-1 **(end of the song)**	-1 **(end of the song)**

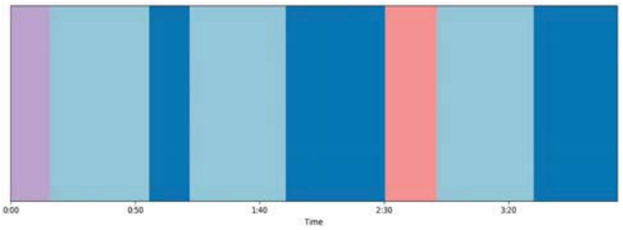


Fig. 1. Example structure of a song in PNG.

A. Dataset

In this study we collected 2786 full length songs in MP3 format with 44.100 kHz sampling rate each and bitrate ranging from 128 to 320 kbit/s. Collection consisted of four genres; Classic, Metal, Pop and Trance. Songs were collected completely randomly, except that too short or too long songs were omitted. Number of samples for each genre is given in Table III.

TABLE II
LIBROSA PARAMETERS BY GENRE FOR SEGMENTATION

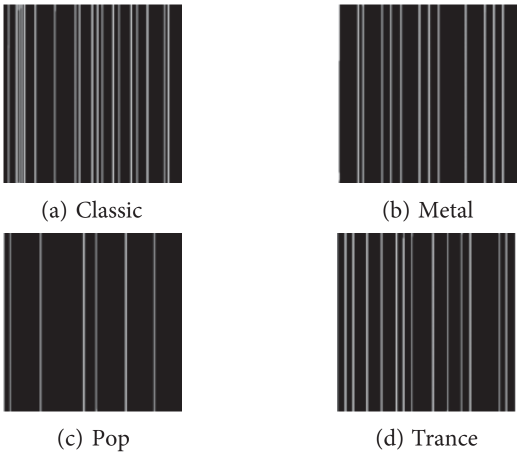
Parameter	**Classic**	**Metal**	**Pop**	**Trance**
sr	44100	44100	44100	44100
n_fft	4096	4096	4096	4096
hop_length	1024	1024	1024	1024
quantile (mean shift)	0.05	0.10	0.20	0.10
n_samples (mean shift)	1000	500	500	500
Eigenvector count (k)	20	10	5	10

TABLE III
DATASET DETAILS BY GENRE

Genre	**Training**	**Test**	**Total**
Classic	522	50	572
Metal	667	50	717
Pop	828	50	878
Trance	569	50	619
Total	**2586**	**200**	**2786**

B. Convolutional Neural Network

All song structure images generated using laplacian segmentation method were batch-processed to be CNN inputs. We used OpenCV library to crop the white space around generated structure image and performed edge detection through Canny method. Then the image is resized into 300x300 pixels and put into relative genre folder. Resulting images are one of the novel features we present in this work. In Fig. 2, some examples of the input images (hence our features) are shown.



(a) Classic

(b) Metal

(c) Pop

(d) Trance

Fig. 2. Feature images derived from song structure.

Various CNN architectures and parameters were explored to measure the performance of the derived feature images. TensorFlow with Keras in Python is used for generating the CNN. 50 trials were performed on the dataset, each time keeping random 50 samples for the test and the rest of the samples for training. Network architecture generated by ConvNet Drawer [18] is shown in Fig. 3 and parameters used are shown in Table IV. A drop-out layer used after third max-pooling layer with a parameter of 0.25 and a Gaussian noise layer with a parameter of 0.1. Results are given in Section V.

TABLE IV
PARAMETERS FOR CNN CLASSIFIER

Parameter	Value
Loss	Sparse Categorical Cross Entropy
Optimizer	Adam
Metrics	Accuracy
Batch size	32
Epochs	10
Validation split	%10
Activations	reLU (except last layer, softmax)

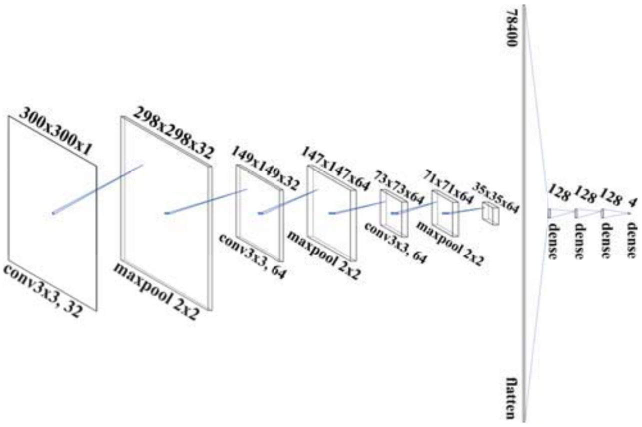


Fig. 3. CNN classifier architecture.

C. Voting Classifier

Another feature set we introduce is statistical data computed from generated CSV file for each genre containing all song structure information. All features mentioned here calculated genre-specific These features are as follows;

- **Average section count** (Eq. 1), computed by dividing the cumulated section count of all songs of a genre to the total song count of the genre.
- **Average tempo** (Eq. 2), computed by dividing the cumulative tempo value of all songs of a genre to the total song count of the genre.
- **Average utempo** (Eq. 3), same as tempo using utempo value computed in Librosa.
- **Average section length** (Eq. 4), computed by dividing the cumulated average section length of each song in a genre to the total song count of the genre.
- **Average repeating section count** (Eq. 5), computed by dividing the cumulation of repeating section counts in a genre to the total song count of the genre.
- **Average standard deviation of section lengths** (Eq. 6), computed by dividing the cumulation of standard deviation (between song section lengths) of each track in a genre to the total song count of the genre.

$$f_1 = \frac{\sum_{i=1}^{k} s_i}{k} \quad (1) \qquad f_2 = \frac{\sum_{i=1}^{k} t_i}{k} \quad (2) \qquad f_3 = \frac{\sum_{i=1}^{k} ut_i}{k} \quad (3)$$

$$f_4 = \frac{\sum_{i=1}^{k} (\frac{\sum_{j=1}^{m} l_j}{m})}{k} \quad (4) \qquad f_5 = \frac{\sum_{i=1}^{k} r_i}{k} \quad (5) \qquad f_6 = \frac{\sum_{i=1}^{k} \sigma_i}{k} \quad (6)$$

To calculate above features, a C# code for parsing the CSV file was written. Algorithm separated 50 random songs for testing and used the rest of the songs for calculating six features mentioned above. This procedure repeated for 100 times for all genres and all the statistical data were averaged. Results by genre can be seen in Table V.

TABLE V
FEATURE SETS BY GENRE (AVERAGE OF 100 TRIALS)

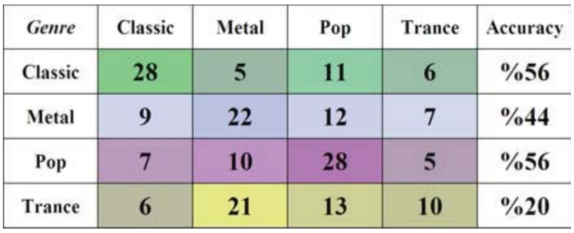
Features	Classic	Metal	Pop	Trance
Avg. section count	9,82	7,62	4,50	8,03
Avg. tempo	123,84	121,37	117,13	133,86
Avg. utempo	220,04	174,24	114,20	117,87
Avg. sect. length (seconds)	26,44	26,71	27,91	29,10
Avg. repeating sect. count	3,79	3,55	2,86	2,94
Avg. std. dev. of sect. len.	17,55	15,84	16,81	18,38

After calculation of this feature set for each genre, these features also calculated for each training sample while testing. Then, euclidean distance for each feature between training sample and the genre is calculated. Each least distance gets an upvote for the related genre. Maximum voted genre is accepted as the classificatio result. We can weight these votes according to the importance of a specifi feature. According to our observation, weighting section count, utempo, repeating section count and variance features with 1.25, tempo and section length features with 1.0 gave the best results. For the samples received the same vote with another genre/genres, classificatio is considered false.

$$class = argmax(v_{classic}, v_{metal}, v_{pop}, v_{trance}) \quad (7)$$

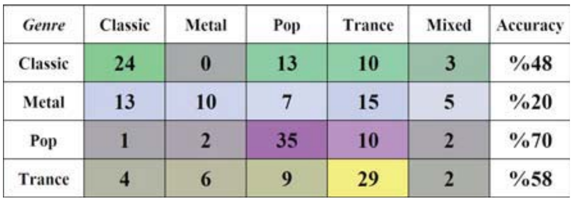
IV. RESULTS

Best accuracy with the CNN classificatio measured as %44 for total training samples. This means 88 samples out of 200 were classifie correctly solely using song structure images. Confusion matrix for each genre can be seen in Fig. 4. For Voting Classifie , average classificatio result of 100 trials was 46,81. This means 93,63 samples out of 200 were classifie correctly solely using six calculated features. Best attempt among this 100 trials also recorded and accuracy was %49. Confusion matrix for this best attempt can be seen in Fig 5.



Genre	Classic	Metal	Pop	Trance	Accuracy
Classic	28	5	11	6	%56
Metal	9	22	12	7	%44
Pop	7	10	28	5	%56
Trance	6	21	13	10	%20

Fig. 4. Confusion matrix for CNN Classifier.



Genre	Classic	Metal	Pop	Trance	Mixed	Accuracy
Classic	24	0	13	10	3	%48
Metal	13	10	7	15	5	%20
Pop	1	2	35	10	2	%70
Trance	4	6	9	29	2	%58

Fig. 5. Confusion matrix for Voting Classifier.

V. Conclusion

In this work, we presented seven novel high-level features derived from song structure analysis and measured the performance of these features with two different classifier using our own four-genre dataset. CNN classifie performed with %44 accuracy and weighted Voting Classifie performed %49 accuracy at best for four genres of music.

These results showed a meaningful improvement for classificatio solely using the introduced feature sets over random prediction. Thus, it can be concluded that these features has potential of use in different scenarios. For future works, performance of the combination of these novel features with various existing features needs to be examined using different datasets and genres.

References

[1] D. Pye, "Content-based methods for the management of digital music," 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings (Cat. No. 00CH37100), Vol. 4 IEEE, pp. 2437-2440, June 2000.

[2] G. Tzanetakis and P. Cook, "Music genre classificatio of audio signals," IEEE Trans. Speech and Audio Process., Vol. 10, no. 5, pp. 293–302, July 2002.

[3] T. Li, M. Ogihara and Q. Li, "A comparative study on content-based music genre classification " Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 282-289, July 2003.

[4] T. Lidy and A. Rauber, "Evaluation of feature extractors and psycho-acoustic transformations for music genre classification " ISMIR, pp. 34-41, September 2005.

[5] Z. Cataltepe, Y. Yaslan, and A. Sonmez, "Music genre classificatio using MIDI and audio features," EURASIP Journal on Advances in Signal Processing 2007, 1-8, 2007.

[6] A. Meng, P. Ahrendt, J. Larsen and L. K. Hansen, "Temporal feature integration for music genre classification " IEEE Transactions on Audio, Speech, and Language Processing 15.5, 1654-1664, 2007.

[7] I. Panagakis, E. Benetos and C. Kotropoulos, "Music genre classifica tion: A multilinear approach," ISMIR, pp. 583-588, 2008.

[8] C. H. Lee, J. L. Shih, K. M. Yu and H. S. Lin, "Automatic music genre classificatio based on modulation spectral analysis of spectral and cepstral features," IEEE Transactions on Multimedia 11.4, 670-682, 2009.

[9] Y. M. Costa, L. S. Oliveira, A. L. Koerich, F. Gouyon, and J. G. Martins, "Music genre classificatio using LBP textural features," Signal Processing 92.11, 2723-2737, 2012.

[10] L. Nanni, Y. M. Costa, A. Lumini, M. Y. Kim and S. R. Baek, "Combining visual and acoustic features for music genre classification " Expert Systems with Applications 45, 108-117, 2016.

[11] C. Senac and T. Pellegrini, "Music feature maps with convolutional neural networks for music genre classification " Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing, pp. 1-5, June 2017.

[12] H. Bahuleyan, "Music genre classificatio using machine learning techniques," arXiv preprint arXiv:1804.01149, 2018.

[13] Z. Durdag and P. Erdogmus, "Müzik türlerinin derin öğrenme ağları ile sınıflandırılması " Sakarya University Journal of Computer and Information Sciences 2/1, 53-60, April 2019.

[14] M. Defferrard, K. Benzi, P. Vandergheynst and X. Bresson, "Fma: A dataset for music analysis," arXiv preprint arXiv:1612.01840, 2016.

[15] B. McFee et al., "librosa: Audio and music signal analysis in python," Proceedings of the 14th python in science conference, Vol. 8., July 2015.

[16] B. McFee and D. Ellis, "Analyzing song structure with spectral clustering," ISMIR, pp. 405-410, 2014.

[17] Librosa [Online]. Available:librosa.github.io/librosa/auto_examples/plot_segmentation.html. [Accessed: June 3, 2020].

[18] ConvNet Drawer [Online]. Available:github.com/yu4u/convnet-drawer. [Accessed: June 3, 2020].