2022 17th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP) | 978-1-6654-5727-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ISAI-NLP56921.2022.9960254

Visual-based Musical Data Representation for Composer Classification

Somrudee Deepaisarn
Sirindhorn International Institute of Technology, Thammasat University
Pathum Thani, Thailand
somrudee@siit.tu.ac.th
Corresponding Author

Suphachok Buaruk
Sirindhorn International Institute of Technology, Thammasat University
Pathum Thani, Thailand
d6522300067@g.siit.tu.ac.th

Sirawit Chokphantavee
Sirindhorn International Institute of Technology, Thammasat University
Pathum Thani, Thailand
6222782250@g.siit.tu.ac.th

Sorawit Chokphantavee
Sirindhorn International Institute of Technology, Thammasat University
Pathum Thani, Thailand
6222782227@g.siit.tu.ac.th

Phuriphan Prathipasen
Sirindhorn International Institute of Technology, Thammasat University
Pathum Thani, Thailand
6222780619@g.siit.tu.ac.th

Virach Sornlertlamvanich
Faculty of Engineering, Thammasat University
Pathum Thani, Thailand
Faculty of Data Science, Musashino University
Tokyo, Japan
virach@gmail.com

***Abstract*—Automated classification for musical genres and composers is an artificial intelligence research challenge insofar as music lacks a rigidly defined structure and may result in varied interpretations by individuals. This research collected acoustic features from a sizable musical database to create an image dataset for formulating a classification model. Each image was constructed by combining pitch, temporal index length, and additional incorporated features of velocity, onset, duration, and a combination of the three. Incorporated features underwent Sigmoid scaling, creating a novel visual-based music representation. A deep learning framework, fast.ai, was used as the primary classification instrument for generated images. The results were that using velocity solely as an incorporated feature provides optimal performance, with an F1-score of 0.85 using the ResNet34 model. These findings offer preliminary insight into composer classification for heightening understanding of music composer signature characterizations.**

***Index Terms*—Music, Data Representation, Composer, Deep Learning, Artificial Intelligence**

I. Introduction

Traditionally, music theory professionals were known to be the only experts capable of identifying composers and genres from pieces of music. Gaining an understanding and interpretation of music is complicated, and can also be considered subjective for individual theorists. In recent decades, prospering artificial intelligence technology has begun to play significant roles in coping with these tasks, especially through classifying music into various categories. This is because music classification also serves as a foundation of other music-related applications, such as music recommendation [1] and generation systems [2], in order to construct more robust and advanced approaches.

Previous work in the area of music representation and classification usually involved mostly the signal domain, which is coherent with the music's end-product in the form of signals. For instance, signal processing-based and/or time series-based techniques utilizing one-dimensional temporal features are primarily applied to this kind of problem. Spectrogram-based analysis, on the other hand, creates a two-dimensional representation of musical signals to an extent that some of the original features are clearly presented. Dieleman and Schrauwen compared the performance of using spectrogram-based musical data against using audio forms as inputs for training convolutional neural networks for music tagging. While spectrograms gave slightly better tagging performance, the raw audio provided the ability to extract more detailed characteristics of music including frequency decomposition [3]. However, only a few studies view and present music in different aspects. The symbolic representation of music encoded within the MIDI format is one of the interesting ways. It is suggested that employing the symbolic representation of music is preferable due to its independence from the external environment noise [4]. The study of the symbolical music by Jain *et. al.* reports a 70% accuracy on six-composer classification using datasets prepared by transforming musical features into grayscale image [5]. Several studies in music classification have been carried out on different datasets, for example, [6], [7] and [8].

In this study, the concept of visual-based representation of symbolic musical data is introduced. We sampled music pieces from a large classical music dataset called the MAESTRO dataset [8]. Acoustic features were turned into grayscale and multi-channel red-green-blue (RGB) images to derive a

representation of music, with information retained on two-dimensional maps of features. Then, composer classification performance was used as an evaluation measure for such representations. The rest of this paper is organized as follows. Section II describes the method, including the dataset, data representation, composer classification models, and training and test data partitioning. Section III presents and discusses the experimental results. Section IV concludes the study and directions for future work.

II. Method

A. MAESTRO Dataset

The maestro-v3.0.0 [8] dataset with Musical Instrument Digital Interface (MIDI) file format was utilized to conduct this study. The dataset consists of 198.7 hours of 1276 piano pieces from 60 different virtuosic composers of classical music. The fine alignment method was applied to align recorded notes and acoustic audio with a precision of approximately 3 ms. A MIDI file describes the acoustic features of each music note, including pitch, velocity, onset, and duration. Where pitch is the frequency of the note, velocity is the rate of pressing a key on the piano, onset is the starting time for a note, and duration is the length of time between adjacent pitches.

B. Visual-based Representation of Music

The acoustic features provided in the MIDI file of the MAESTRO dataset contain detailed information on each note, which can then be converted into numerical values using the **`pretty_midi`** tool [9]. These acoustic features, including pitch and velocity, are extracted as the primary input features for the classification of composers [10]. A music note is visually represented on an image at a specified feature channel where a pixel intensity indicates its value for velocity, onset, or duration. Each feature channel is arranged in the shape of *(T, P)*, where *T* is the temporal index representing the ordinal timestamp of the note in a music piece, and *P* is the value for pitch ranging from 0 to 127. For image generation, the temporal index and pitch of a note were plotted on an image's horizontal and vertical axes, respectively. The original values, x, of velocity, onset, or duration were normalized using the sigmoid function as described in Equation (1). This contributed to the uniqueness of this work as the sigmoid function optimizes the dynamic range of the pixel intensity limited to a value between 0 and 1.

$$S(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

In the experiment, the grayscale single channel images were constructed from a single feature of velocity, onset, and duration, which are framed in a shape of $P \times T$, with T varying from 200-600 timestamps. These three features were also combined to create multi-channel RGB images. For example, the first, second, and third rows of Fig. 1 illustrate the velocity, duration, and onset frame of a 400 temporal index MIDI segment from a music piece, respectively. While the color image in the fourth row comprises three feature channels with the same pitch value.

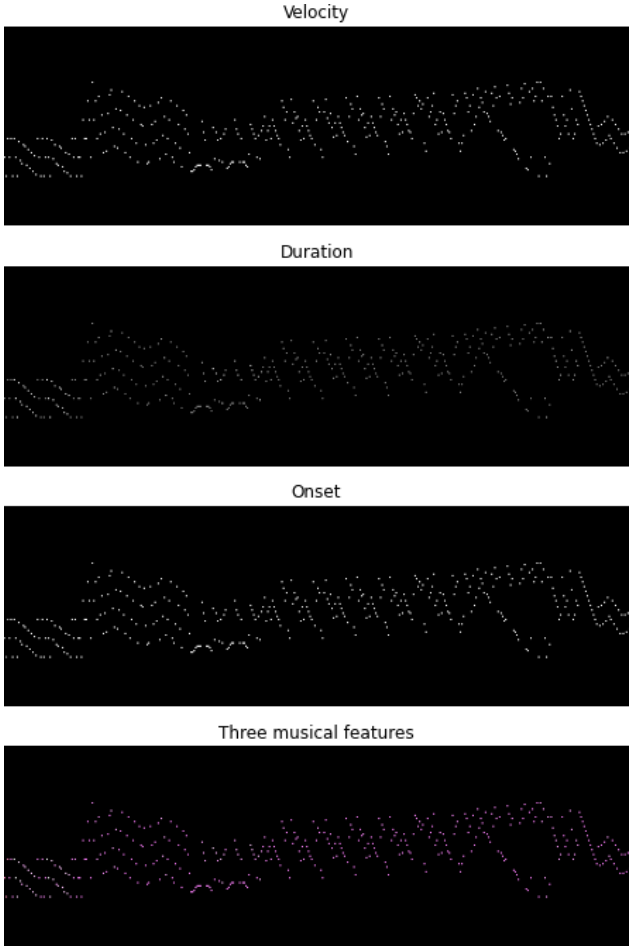

Fig. 1. Visual-based representation for musical data: Listed from top to bottom - The grayscale image generated with a single feature of velocity, duration, and onset, and the color image generated with the combination of all three features represented as the RGB channel intensity of each note with the vertical and horizontal axes representing the pitch and temporal index, respectively.

C. Composer Classification Models

The models and pipelines for music composer classification have experimented with varied parameters using a 5-fold cross-validation scheme. The deep learning framework used for training and prediction is described below. The model compiler parameters, i.e., type of deep learning model and learning rate, were optimized. Each experimental parameter related to the input data, such as the length of the temporal index and types of acoustic features, is elaborated in II-D.

FastAI is a PyTorch-based deep learning library that facilitates not only a state-of-the-art deep learning approach, which can be utilized swiftly and easily but also the APIs for customizing and engineering a deep learning model in low-level to bestow the experienced users with sufficient flexibility [11]. One of the most iconic attributes of FastAI, is the learning rate finder, which suggests the optimal value for the learning rate parameter following a sample training epoch,

making the hyper-parameter tuning simpler than ever before. There are many previous studies describing the role of FastAI for deep learning tasks, especially for image classification, for example, plant leaf disease recognition and classification [12].

In this study, pre-trained deep learning models, such as residual neural network (ResNet) [13] and EfficientNet [14], are utilized as the base for our models, which identifies the composer corresponding to the input images generated from the acoustic features of music pieces. Our model utilized the cross-entropy loss function with the Adam optimization algorithm. For each training epoch, the input was fed into the model as a batch of 16 instances. Lastly, the initial learning rate was computed for each experiment utilizing the **`lr_find`** function implemented in the FastAI framework.

D. Training and Test Data

This work classifies the virtuosic composers of the MAESTRO dataset based on acoustic features in Table I, which are transformed into the image representation of the musical data. Using the method described in II-B, the deep learning approaches were expected to be a tool for extracting patterns characterized by the image dataset.

TABLE I
EXPLANATION OF ACOUSTIC FEATURES

Acoustic features	Explanation
Velocity	How hard a piano key is struck
Onset	The beginning point of a note
Duration	How long a note is played

Originally, the MAESTRO dataset included 60 composers with 1,276 pieces of music. We only included the composer who had written more than 100 pieces for classification labels to prevent the issue of each composer having insufficient samples for model training and testing. With this condition, the input dataset is prepared, giving us five composers with a total of 809 pieces of music, which are divided into 70:30 for training:test sets yielding 566 and 243 pieces, respectively. Furthermore, we also extended our experiments to 14 composer classifications by filtering out the composer with less than 25 music pieces provided in this dataset, leaving us with 1,160 music pieces in total. Afterwards, these 1,160 pieces were split into training and test sets at 70:30.

After that, each piece from the training and test sets was divided into segments with the same temporal index length and generated an image array of size defined by temporal index length. For the five-composer experiment, the training and test sets contain 7,379 and 3,244 images produced from music segments of length 400 temporal indexes per image. There was no overlapping part in the music segments. In addition, the classification performance was investigated among the altered temporal index lengths of 200, 400, and 600, producing the image size variation.

In order to ensure the validity of the overall model performance, we split the music pieces of each composer into the training set and the test set first, then perform the segmentation of music pieces afterwards. The merit of our procedure is to prevent contamination between training and test data. Since there should exist some correlation among segments from the same music piece, which could make the model overly optimistic if trained and tested on randomly partitioned segments.

III. RESULT AND DISCUSSION

In this work, composer classification was performed on the visual-based musical data representation. Input data and classification models constituted of adjustable parameters are discussed in this section, including the based pre-trained model, length of the temporal index, and acoustic features used in the visual-based music data representation. The initial learning rate is dynamically calculated for each experiment by the built-in **`lr_find`** function provided by FastAI. The softmax activation function was used in the output layer with the Adam optimizer. The F1-score of classification performance on each parameter was investigated as shown in Table II. For the five-composer classification, the optimal model in terms of prediction performance and the computational cost was the ResNet34, with a temporal index length of 400 and the velocity as the pixel's grayscale, achieving a classification accuracy of 0.85. The experimental results are shown as the confusion matrix in Fig. 2. In this section, all experimental results are discussed.

TABLE II
CLASSIFICATION PERFORMANCE OF MODELS CONSTRUCTED BY VARIOUS FEATURES

#composer	models	features	length	F1-score
5	EfficientNet B7	Velocity	400	0.87
5	ResNet34	Velocity	200	0.85
5	**ResNet34**	**Velocity**	**400**	**0.85**
5	ResNet34	Velocity	600	0.82
5	ResNet34	onset	400	0.78
5	ResNet34	duration	400	0.78
5	ResNet34	RGB	400	0.72
14	ResNet34	Velocity	400	0.68

A. Deep Learning Models

The experimenting models in this study consist of two deep learning models, including ResNet34 and EfficientNet-B7, which serve as a base for our models utilizing the transfer learning technique. From Table II, the F1-score for five-composer classification suggests that the velocity is the most efficient feature where EfficientNet-B7 performs slightly better than ResNet34. However, the training session of EfficientNet-B7 is seven times the period required to train ResNet34, given the same training dataset. Therefore, the following experiments conducted in this paper were primarily based on ResNet34 since it provided almost the same F1-score but was significantly less computationally expensive.

B. Temporal Index Lengths

This part investigates the effects of temporal index lengths on the classification of images generated from acoustic features. This idea of segmenting a music piece was previously presented by Q. Kong *et al.*, achieving an accuracy of about 0.65 from 30-second clip-wised classification [10]. In our work, the length of the temporal index was varied from 200 to 600 in order to create the datasets from different sizes of image-represented music segments. The based pre-trained model and acoustic features were set to be ResNet34 and velocity, respectively. Overall, the index length did not significantly affect the F1-score in these measures. As indicated by Table II, the F1-score of the models are 0.85, 0.85, and 0.82 for the training dataset of the temporal index, equal to 200, 400, and 600, respectively.

C. Acoustic Features

In this section, F1-score evaluated metrics were the performance indicator of composer classification computed from the 3,244 images of the test dataset. According to Kong *et. al.*, the combination between pitch, velocity, and onset frame was used, achieving the composer classification accuracy of about 0.65 [10]. For our work, the acoustic features used for image generation instead consisted of the velocity, onset, and duration of each pitch. Moreover, these three acoustic features were combined, resulting in the three-channel RGB color image dataset that was then utilized as another dataset for our experiment. Each of these four image datasets was employed to train and fine-tune a deep learning model, in this case, the ResNet34, which achieved the F1-score of 0.85, 0.78, 0.78, and 0.72 as a result of the velocity, onset, duration, and combined features, respectively. As exemplified by the implementation, the acoustic feature that gave the highest F1-score was the velocity on its own. This phenomenon is logically sound in the musical aspect since the vertical dimension of the image already provides the pitch information, which is one of the essential acoustic features, and the horizontal axis supplies another vital information, which is the arrangement of note and their duration. Therefore, in the view of the pianists, the only thing left to perform this piece is the dynamic, in this case, the velocity. To further elaborate, each piano sheet consists of three predominant groups of notations, including pitch-related notation, time-related notation, and style-related notation [15]. In terms of pitch-related representation such as staff, clef, sharp, and flat, the MIDI integer encoded pitch can entail the combination of this information, and we portray them onto the vertical dimension of the generated image. As for time-related notation, namely notes and rests, the insight of each note duration and rest duration can be depicted by the horizontal dimension of the image data. Lastly, the style-related notation, which is dynamic and accent, can be directly represented by solely one quantity in MIDI, namely the velocity. Following this rationale, it is undoubtedly reasonable to achieve a higher F1-score when only the velocity feature is utilized for encoding the image yielding the style-related that fulfills all three clusters of music notation. In contrast, the onset and duration of notes cannot convey such detail, worsening the matter; they impart solely the redundant insight that can already be obtained from the arrangement of notes on the image representation. Regarding the combination of three acoustic features, the classification performance using the generated RGB image unexpectedly did not catch up to using the grayscale velocity values on its own. The redundancy of the data from onset and duration may likely confuse the classification model and hence lessen the capability of the model. Note that using the sigmoid function to normalize the pixel intensity overcomes the effect of extreme value appearing in an acoustic feature when the conventional `MinMaxScaler` normalization is applied. This normalization enhances the quality of images in terms of the distinction of pixel intensity so as to improve the classification performance.

D. Extended Multi-class Classification

Besides adjusting the based pre-trained model, temporal index length, and acoustic features, we also investigated the effect of extending the number of classes from 5 composers to 14 composers. As demonstrated in Table II, the F1-Score for 14 composers classification dropped drastically compared to the five-composer classification with the same model parameters (using velocity for grayscale intensity, temporal index length of 400, and ResNet34 as a based pre-trained model). To justify the phenomenon, this inferior result may occur due to a higher degree of imbalance in the dataset and insufficiency of data. For the dataset constructed for the 14-composer classification, the number of music composed by a composer ranges from 26 to 201 musical pieces. In contrast, the individual composers in the 5-composer classification have at least 100 musical pieces.

E. Interpretation of Results

The confusion matrix obtained from the results of five-composer classification using ResNet34 as a based pre-trained model, a velocity-based acoustic feature with a temporal index length of 400, is presented in Fig 2. As depicted in the confusion matrix, notable misclassification arises between the renowned name Ludwig Van Beethoven and Franz Schubert. In this case, our model classifies Beethoven's compositions incorrectly as the work of Schubert a noticeable number of times. The possible underlying reason for such a phenomenon stemmed from the fact that both Beethoven and Schubert dwelled not only in the same city of origin but also shared overlapping time frames. Furthermore, as stated in [16], [17], the admiration and reverence of Schubert toward Beethoven's compositions such as the Fifth Symphony and the C major Mass had a remarkable influence on several compositions of his own, for example, the B flat major Sonata, Op. 36 and the Grand Duo in C major, Op. 140. Therefore, it seems logically sound that one may misclassify these music compositions and so as the computational models. In summary, the above discussion demonstrated the efficiency of our model in capturing the important features and patterns of each specific composer underlying their music pieces.

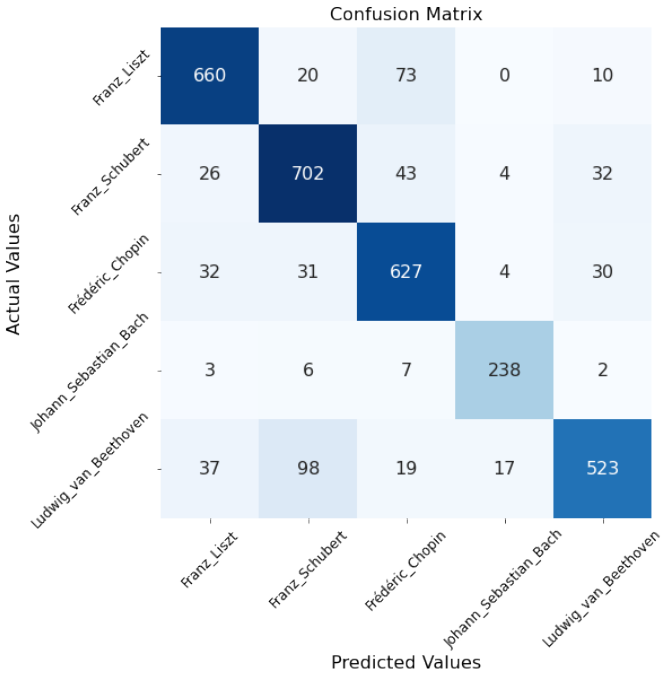

Fig. 2. Confusion matrix of the five-composer classification performance using the ResNet34 model, experimented on velocity incorporated images with temporal index length of 400 in the training and test datasets. Chopin, Schubert, Beethoven, Bach, and Liszt are among the composers included in the model evaluation, yielding 7,379 and 3,244 segments for training and testing, respectively.

IV. Conclusion and Future Work

The experiments in this study reveal the potential of acoustic features extracted from music pieces to be represented visually as images. In this work, we separated the experiment into three major parts, including the investigation of the performance of deep learning models based on ResNet34 and EfficientNet-B7, the evaluation of classification performance concerning the influence of temporal index length, and the assessment of composer classification on the grayscale images formed by different acoustic features, i.e., velocity, onset, duration, and the RGB color images created by the combination of the three features. Our model and extraction method yielded the highest F1-score of 0.87 by utilizing EfficientNet-B7 and 0.85 by using ResNet34, with the velocity as the only feature contributing to the pixel intensity of the generated images and the temporal index length of 400. In addition, when scaling the pixel intensity representing the musical features on an image dataset, the sigmoid normalization method gave rise to a superior classification performance compared to conventional normalization. For future work, the procured information might apply to a broader genre of music as well as the music era classification. Furthermore, the extracted musical features and their representation may also be studied to understand the musical composers' signatures toward a unique music generation technique.

Acknowledgment

The authors gratefully acknowledge the financial support provided by the Thammasat University Research fund under the TSRI, Contract No. TUFF19/2564 and TUFF24/2565, for the project of "AI Ready City Networking in RUN", based on the RUN Digital Cluster collaboration scheme. The authors would like to express special thanks to Maneesha Perera for outlining the background of music theory.

References

[1] F. Fessahaye, L. Perez, T. Zhan, R. Zhang, C. Fossier, R. Markarian, C. Chiu, J. Zhan, L. Gewali, and P. Oh, "T-recsys: A novel music recommendation system using deep learning," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1–6, 2019.

[2] H. H. Mao, T. Shin, and G. Cottrell, "Deepj: Style-specific music generation," in *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, pp. 377–382, 2018.

[3] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6964–6968, IEEE, 2014.

[4] S. Kim, H. Lee, S. Park, J. Lee, and K. Choi, "Deep composer classification using symbolic representation," *arXiv preprint arXiv:2010.00823*, 2020.

[5] S. Jain, A. Smit, and T. Yngesjö, "Analysis and classification of symbolic western classical music by composer," *Preprint.[Online]. Available: http://cs229. stanford. edu/proj2019aut/data/assignment_308832_raw/26583519. pdf*, 2019.

[6] Z. Cataltepe, Y. Yaslan, and A. Sonmez, "Music genre classification using midi and audio features," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–8, 2007.

[7] D. Herremans, D. Martens, and K. Sörensen, *Composer Classification Models for Music-Theory Building*, pp. 369–392. Springer, 10 2015.

[8] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C.-Z. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck, "Enabling factorized piano music modeling and generation with the MAESTRO dataset," in *International Conference on Learning Representations*, 2019.

[9] C. Raffel and D. P. Ellis, "Intuitive analysis, creation and manipulation of midi data with pretty_midi," in *15th international society for music information retrieval conference late breaking and demo papers*, pp. 84–93, 2014.

[10] Q. Kong, K. Choi, and Y. Wang, "Large-scale midi-based composer classification," *arXiv preprint arXiv:2010.14805*, 2020.

[11] J. Howard and S. Gugger, "Fastai: a layered api for deep learning," *Information*, vol. 11, no. 2, p. 108, 2020.

[12] A. Chakraborty, D. Kumer, and K. Deeba, "Plant leaf disease recognition using fastai image classification," in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 1624–1630, IEEE, 2021.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[14] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, pp. 6105–6114, PMLR, 2019.

[15] C. Schmidt-Jones, *Reading Music: Common Notation*. University Press of Florida, 2009.

[16] W. Nohl and F. H. Martens, "Beethoven's and schubert's personal relations," *The Musical Quarterly*, vol. 14, no. 4, pp. 553–562, 1928.

[17] M. Solomon, "Schubert and beethoven," *19th-Century Music*, vol. 3, no. 2, pp. 114–125, 1979.