

# Fast Recognition using Bar-based Data Mapping for Gamelan Music Genre Classification

KHAFIIZH HASTUTI<sup>1</sup>, PULUNG NURTANTIO ANDONO<sup>1</sup>, ARRY MAULANA SYARIF<sup>1</sup>,  
 AZHARI AZHARI<sup>2</sup>

<sup>1</sup> Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang, Indonesia  
 (e-mail: afis@dsn.dinus.ac.id, pulung.nurtantio.andono@dsn.dinus.ac.id, arry.maulana@dsn.dinus.ac.id)

<sup>2</sup> Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia  
 (e-mail: arisn@ugm.ac.id)

Corresponding author: Khafiizh Hastuti (e-mail: afis@dsn.dinus.ac.id).

⋮ **ABSTRACT** This research aims to develop a *gamelan* music genre classifier based on the musical mode system determined based on the dominant notes in a certain order. Only experts can discriminate the musical mode system of compositions. The Feed Forward Neural Networks method was used to classify *gamelan* compositions into three musical mode systems. The challenge is to recognize the musical mode system of compositions between the initial melody without having to analyze the entire melody using a small amount of data for the dataset. Instead of conducting a melodic extraction from audio signal data, the text-based skeletal melody data, which is a form of extracted melodic features, are used for the dataset. Unique corpuses are controlled based on the cardinality of the one-to-many relationship, and a data mapping technique based on the bars is used to increase the number of corpuses. The results show that the proposed method is suitable to solve the specified problems, where the accuracy in recognizing the class of unseen compositions between the initial melody achieves at 86.7%.

⋮ **KEYWORDS** Multi-class classification; Fast recognition; Bar-based data mapping; Feedforward Neural Networks; *Gamelan* music genre classification.

## I. INTRODUCTION

**M**USIC genre classification is a part of music information retrieval research (MIR) which retrieves information of music to measure musical similarity [1]. A music genre classifier is to predict style of compositions based on existing music characteristics. This research aims to develop a music genre classification system for Javanese traditional music known as *gamelan* based on the musical mode system determined based on the dominant notes in a certain order. In reality, only experts can discriminate the musical mode system of a composition. Notes arrangement based on *gamelan* rules is melodic patterns used by experts to discriminate the musical mode system of a composition. This became a motivation to develop a *gamelan* music genre classification system.

The fast recognition terminology used in this article

refers to the ability of a computer program to quickly recognize the musical genre by using the beginning parts of the melody. Bar, which is a musical element to determine the time segment in a composition, is used as an instrument to measure the ability of the program to quickly recognize music genres using a number of bars. The challenge increases when dealing with a limited number of corpuses.

Machine learning approach with the Feed Forward neural networks (FFNN) method has been proved in solving music genre classification problems [2-4]. A Feed Forward Neural Network classifier was developed to recognize a composition's musical mode system of *gamelan* at the initial melody without having to analyze the entire melody. The network was trained using a dataset consisting of a small number of compositions collected from three classes of the musical mode system. If the compositions are cut for a data

segmentation within a short duration as works by [3, 5], the number of corpuses will remain as small as the number of compositions. In this condition, the networks will be difficult to learn the patterns in a small dataset. Thus, the entire melody in the composition should be organized and used to increase the number of corpuses.

A bar-based data mapping to keep the entire melody being used to train the networks was proposed in order to increase the number of corpuses to solve the small number of corpus problems.

## II. RELATED WORKS

A Feed Forward Neural Network music genre classifier using audio data as the input for two classification process was developed by [6]. The first classification was to extract audio data to recognize the notes, and the second classification was conducted using the extracted notes to recognize six music genres. The method proposed by [6] is to classify songs into genres of blues, electronic, folk, country, pop, rock so it is questionable regarding the use of notes information as the only feature to classify musical genres, and unfortunately there is no information of the number of songs used as the dataset. A dataset containing 1000 songs collected from GTZAN music collection is considered as limited size data to train a FFNN classifier with a target which consists of 10 music genres [7]. If the dataset was proportionally distributed, each class was trained using 100 songs. While a FFNN for a multi-class music genre classification using an open-source audio processing software framework called Marsyas (Music Analysis, Retrieval, and Synthesis for Audio Signals) was developed by [2, 3]. The FFNN architecture developed by [2] contains 16 units for the input layer, 10 units for the hidden layer and four music genres as outputs, and the dataset consisting of 400 samples is divided into 200 samples for training, 100 samples for validation and 100 samples for testing. In contrast, despite having the same number of classes for output, the network developed by [3] uses 100 samples divided into 70:30 for training and testing without samples for validation, which is rare, and the FFNN architecture contains 135 units for the input layer and 10 units for the hidden layer. The performance of classifier developed by [2] is improved using fixed-size momentum in order to overcome limitations in backpropagation training, and to increase speedup over standard momentum. This method uses additional information, where weights values and bias are initially randomized and iteratively changed until error is reduced. Furthermore, it is claimed that accuracy of fixed-size momentum neural networks outperforms standard neural networks and standard neural networks using momentum.

The Feed Forward Neural Network method is not only restricted in the music genre classification, but it is also used to develop a music genre classifier which involves emotion and semantic as the target [4, 8-10]. A FFNN music genre classification which involves mood-based classification commonly uses energy, melody and timbre as features to

represent the moods. Furthermore, comparing to SVM which was also used by [9] in the same task of the classification, FFNN achieves better accuracy both in the genre and mood classification [8]. The Convolutional Neural Network (CNN) method was used by [10] to classify emotion contained in a melody, where the number of songs in the dataset was more than used by [8]. A classifier based on FFNN was developed by [4] to learn semantic features for music genre classification, where the extracted audio data is selected to identify complex patterns which is then mapped to the output layer. Instead of using a data mapping technique, a feature extraction for a music genre classification by compacting features as a feature subset was proposed to reduce the computation process [11].

A deep learning method using audio data was proposed by [12, 13] for music genre classification. Audio data is used as a dataset and extracted using Mel-Frequency Cepstral Coefficients (MFCCs), then the extracted data is sent to a CNN classifier to be learned [12]. Furthermore, it is claimed that more than three convolutional layers tend to be trapped at local minima, whereas less than three layers are not sufficient for a music genre classification. Deep belief network (DBNs) on Discrete Fourier Transforms (DFTs) audio data was used to develop a melodic feature extraction system; DBNs supports multi-layer probabilistic data representations learning as a non-linear problem [13].

The music genre classification for traditional music has been developed by [14-16], while [17, 18] developed a classifier related with the types of traditional instruments. Korean traditional music genre classification was developed using several classification algorithms, where the Support Vector Machine (SVM) outperforms multi-layer perception, random forest and decision tree [14]. While a classifier using k-Nearest Neighbor (k-NN) method was developed by [15] for Indian traditional music classification. In contrast, the music genre classification for Bangla music, traditional Bangladeshi music, was developed using FFNN instead of CNN or Long-Short Term Memory (LSTM) for low-cost computation, and FFNN has better accuracy than logistic regression, linear regression, SVM and k-NN algorithm [16]. The number of instruments of the Colombian Andean string music was counted using a FFNN classifier, where the input is audio data containing music recording [17], while [18] developed a FFNN classifier to recognize Persian music instruments.

A music genre classifier is to predict the style or genre of composition based on the characteristics of existing music. This type of research involves selecting compositions that will be converted into short time segments for representation [5], known as melodic feature extraction. Melodic feature extraction is to characterize a melody by reducing the dimension of the melody, in which some melodic elements that can represent relevant information are selected as features.

A sequential pattern mining algorithm called Apriori based on Functions in sequence (AFiS) was proposed by [19] to formulate rules for the melodic features of *gamelan* music.

AFiS algorithm works by partitioning the melody and using a sliding window technique to weight the previous note and the next note according to the order of positions in the sequence data. Rules are constructed for the skeletal melody of *gamelan* music known as *balungan* (Javanese: skeleton). A skeletal melody is a form of extracted *gamelan* melodic features, which contains selected notes extracted from all notes in a composition. It is used by musicians as a guide for playing musical instruments during their performances. Skeletal melody data was also used by [20-21] to formulate skeletal melody compositions based on grammar method, including automatic composition generation [22], and was also used as a dataset in developing the musical mode systems classifier proposed in this research.

### III. PROPOSED METHOD

A multi-class musical mode classification was developed for *gamelan* music. The challenge for the classifier is to recognize a composition's musical mode system at the initial melody without having to analyze the entire melody. *Gamelan* consists of two musical scale systems, namely *slendro* and *pelog*. The number and tuning of notes in these systems are different. *Slendro* consists of five notes, namely 1, 2, 3, 5 and 6. While *pelog* consists of seven notes, namely 1, 2, 3, 4, 5, 6 and 7. There is a dot note, an addition note used to represent silent moment. The *slendro* and *pelog* musical scale systems have their respective musical mode system that function to determine the characteristic or style of a composition. The system works based on the dominant notes in a certain order. The musical mode system in *slendro* is *manyura*, *nem* and *sanga*, while in *pelog* is *barang*, *lima* and *nem*.

The experiment in this research was limited to the development of a classifier to discriminate compositions of *slendro* into three classes. The classifier was built using a FFNN method and pattern recognition technique. Instead of using audio signal extraction to determine melodic features, skeletal melody data from a *gamelan* sheet music collection were used as the dataset. A skeletal melody is an abstraction from a melody. It consists of key notes and musical elements used as a guide for musicians in playing the instruments. A notes sequence in a skeletal melody is distributed in *gatra*, a group of notes that functions as bars in Western music. A *gatra* (bar) consists of four beats, and each beat can contain one or more notes. Figure 1 shows an example of a composition which consists of skeletal melody (a) and melody (b).

A skeletal melody is similar to the chords in Western music. A melodic abstraction for a chord detection was implemented in the works of HARMONET, a system of chorale harmonization in the style of J.S. Bach, in which a chord skeleton based on eighth and sixteenth notes was identified for selecting notes that could be removed to reduce chords conceived as harmonies into harmonic skeleton [23]. The melodic features selection was also conducted for the skeletal melody, but it is also to set the vector length apart from selecting the melody feature. The melodic features

selection was conducted by removing some of notes and musical elements, while the rest are stored as melodic selected features. This technique was proposed by [24] in developing *Gendhing Scientific Pitch Notation* (GSPN), a human-computer based representation of *gamelan* music, in which *gendhing* is a Javanese term which means song. The skeletal melody data were used to train a FFNN classifier with three classes of the musical mode system, namely *manyura*, *nem* and *sanga* as output.



Figure 1. Skeletal melody and melody in a *gamelan* composition

The method in developing the classifier consists of four phases, which are data preparation, data mapping, binary representation, implementation and evaluation. MATLAB 2019 was used for the programming in all the four phases, but the traditional technique was combined in the initial step of the data preparation phase by typing the data in a text document format before manipulated using MATLAB 2019.

### IV. DATA PREPARATION

A melody skeletal data collection from three classes of the *manyura*, *nem* and *sanga* musical mode systems. Each class consists of 35 compositions, so 105 compositions are used as the dataset. Figure 2 shows a melody skeletal example.



Figure 2. Melody skeletal example

Data preparation was conducted in advance by removing all marks, such as curves and circles, including all the dot marks that represents high or low notes if any. The marks removal was conducted since the network was designed to learn melodic patterns based on notes in form of note numbers and the dot note that represents silent moment. The beats in most skeletal melodies contain one note, but in some skeletal melodies, some beats are found containing more than one note. Beat extraction was conducted to set the same length vector for all beats. The first note in a beat is the dominant note, while the successor note is the ornamentation note. Post-sequence truncation technique which chops a number of elements in the beginning of a vector was used for a beat extraction. All the beats which contain more than one note are extracted by keeping the first note and deleting the rest started from the second note.

The data preparation was conducted using a traditional technique by manually typing the compositions data in a text document format. All the dotted notes which represent silent moment were converted to the number of 0 in order to support a computation process [19, 22, 24]. Furthermore, the notes sequence in a melody skeletal is concatenated in one row, and each note was separated by commas.

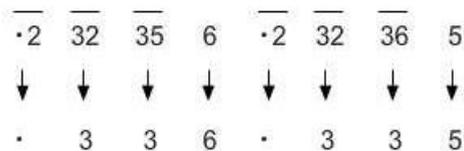


Figure 3. Marks removal and beats extraction example

Figure 3 shows an illustration of the initial step of the data preparation which uses the first line of a composition entitled *Ladrang Jalak Ijo*. The note sequence at the top is the original, and the one at the bottom is the results of the data preparation, where all marks are removed and beats containing more than one note are extracted.

In the text document format, the compositions data of the class of *manyura*, *nem* and *sanga* were inputted in three separated text documents, namely *manyura.txt*, *nem.txt* and *sanga.txt*. In each text document, each composition was inputted in different rows. So, each text document contains 35 rows which represents 35 compositions.

The rest of this article use two compositions of the *nem* class which are *Ladrang Alas Kobong* and *Ladrang Binar* as dummy data for a brief algorithm explanation. In the experiment, all the procedures explanation were also implemented to all compositions in all classes. In the following example, the text document of *nem.txt*, in which the first row contains the notes sequence of *Ladrang Alas Kobong*, and the second row contains the notes sequence of *Ladrang Binar*, looks as follows:

```
0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6, 0, 2, 0, 3, 0, 5, 0,
3, 0, 5, 0, 3, 0, 1, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5,
0, 6, 5, 5, 0, 0, 5, 5, 6, 5, 1, 2, 1, 6, 5, 3, 2, 3, 0, 5, 0, 5, 0,
5, 2, 3, 0, 0, 2, 3, 5, 5, 6, 5, 0, 0, 2, 1, 0, 0, 2, 1, 3, 5, 3, 2,
0, 1, 2, 6
```

```
0, 2, 0, 1, 0, 2, 0, 6, 0, 3, 0, 6, 0, 3, 0, 2, 0, 3, 0, 1, 0, 3, 0,
2, 5, 6, 5, 3, 2, 1, 2, 6, 0, 2, 0, 1, 0, 2, 0, 6, 0, 3, 0, 1, 0, 5,
0, 6, 0, 5, 0, 6, 0, 2, 0, 1, 0, 2, 0, 6, 0, 5, 0, 3, 0, 5, 0, 3, 0,
5, 0, 6, 0, 3, 0, 5, 0, 3, 0, 2, 0, 3, 0, 1, 0, 3, 0, 2, 5, 6, 5, 3,
2, 1, 2, 6
```

Next, the text document which contains compositions data was loaded into the Matlab program, and converted them into a sequence of vectors. The following is the code snippets for creating a sequence of vectors of the *nem* class with *k* is the number of rows in the text document which is equal to the number of compositions and is equal to 35 compositions of the *nem* class.

```
% load .txt format data
classOne = "nem.txt";
nemData = fileread (classOne);
% split data per row resulting 1 x k cell.
nem = regexp (nemData, '\r\n|\r|\n', 'split');
% transpose data resulting k x 1 cell.
nem = transpose (nem);
% create a sequence of vectors for each row
nem = cellfun (@str2num, nem, 'UniformOutput',
false);
% Results
nem{1} = 0 5 0 3 0 5 0 6 0 5 0 3 0 5 0 6 0 2
0 3 ... 6
nem{2} = 0 2 0 1 0 2 0 6 0 3 0 6 0 3 0 2 0 3
0 1 ... 6
```

## V. DATA MAPPING

The dataset which contains 105 compositions is relatively small to train the networks in discriminating the musical mode systems at the initial melody. In this condition, cutting the melody within a short initial duration for input is not recommended since the number of corpuses remains the same. The entire melody in a composition should be organized and used to increase the number of corpuses. Melody is organized in melody lines, bars and beats. A data mapping based on beats tends to produce many duplicate corpuses, and it is difficult to control the data cardinality. The more duplicate corpuses in the dataset will confuse the networks. While a data mapping based on melody lines is less effective in increasing the number of corpuses. Therefore, a data mapping based on bars is suitable to control the unique corpuses and to increase the number of corpuses. This method solves the small number of dataset problems for a fast recognition task.

The notes in the melody skeletal are distributed in bars, where a bar consists of four beats or four notes after data preparation. The length of the sequences of vectors were then reshape by four in order to distribute notes based on the bar length or 1-bar mapping. Continuing the previous examples, the following is the code snippet for reshaping the sequences of vectors of the *nem* class by four with *k* represents the number of the data segmentation in each composition and *p*

represents the number of the data segmentation of all the compositions after concatenation.

```
[rows columns] = size (nem);
for i = 1:rows
% transpose data in each compositions resulting k x 1 cell
nem{i} = transpose(nem {i});
% reshape data in each composition by four resulting 4 x k.
nemOne{i} = reshape(nem{i},4,[]);
end
% results
nemOne{1} =
0 0 0 0 0 0 0 0 0 ... 5
5 5 5 5 2 5 5 1 5 5 ... 5
0 0 0 0 0 0 0 0 0 ... 6
3 6 3 6 3 3 3 6 3 6 ... 5
nemOne{2} =
0 0 0 0 0 0 5 2 0 0 ... 0
2 2 3 3 3 3 6 1 2 2 ... 3
0 0 0 0 0 0 5 2 0 0 ... 0
1 6 6 2 1 2 3 6 1 6 ... 2
% concate cells resulting 4 x p cell
nemOne = horzcat(nemOne {:});
% transpose data resulting p x 4 cell
nemOne = transpose (nemOne);
nemOne =
0 5 0 3
0 5 0 6
0 5 0 3
....
2 1 2 6
```

The data mapping determines the short initial duration or a number of notes at the initial melody which are sufficient for the network to discriminate between musical mode systems, or how fast the classifier can discriminate the musical mode systems based on the number of notes in the data segmentation. Data mapping is designed by mapping the cardinality to one-to-many relationship, where a class can have more than one corpus, but a corpus must be linked to one class. Mapping was conducted based on bars, where the network learns to discriminate the musical mode system of *gamelan* compositions by analyzing note sequence pattern in bars. In a 1-bar mapping, a composition consisting of eight bars will contribute eight corpuses if they are all different from each other.

The data mapping was further conducted by reshaping each sequence of vector by  $n$  where  $n \in \{1, 2, 4\}$  with the value of 1 is for 1-bar mapping where each segmentation containing four notes, the value of 2 is for 2-bar mapping where each segmentation containing eight notes, and the

value of 4 is for 4-bar mapping where each segmentation containing 16 notes. The following are examples of 1-bar, 2-bars and 4-bars mapping for a composition played in the class *nem* entitled *Ladrang Alas Kobong* with  $n$  represents the number of bars, in which each bar contains four notes.

Data preparation:

```
0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6, 0, 2, 0, 3, 0, 5, 0,
3, 0, 5, 0, 3, 0, 1, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5,
0, 6, 5, 5, 0, 0, 5, 5, 6, 5, 1, 2, 1, 6, 5, 3, 2, 3, 0, 5, 0, 5, 0,
5, 2, 3, 0, 0, 2, 3, 5, 5, 6, 5, 0, 0, 2, 1, 0, 0, 2, 1, 3, 5, 3, 2,
0, 1, 2, 6
```

1-bar mapping with  $n = 1$ :

```
(0, 5, 0, 3)
(0, 5, 0, 6)
(0, 5, 0, 3)
...
```

```
(0, 1, 2, 6)
```

2-bars mapping with  $n = 2$ :

```
(0, 5, 0, 3, 0, 5, 0, 6)
(0, 5, 0, 3, 0, 5, 0, 6)
(0, 2, 0, 3, 0, 5, 0, 3)
...
```

```
(3, 5, 3, 2, 0, 1, 2, 6)
```

4-bars mapping with  $n = 4$ :

```
(0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6)
(0, 2, 0, 3, 0, 5, 0, 3, 0, 5, 0, 3, 0, 1, 0, 6)
(0, 5, 0, 3, 0, 5, 0, 6, 0, 5, 0, 3, 0, 5, 0, 6)
...
(0, 0, 2, 1, 0, 0, 2, 1, 3, 5, 3, 2, 0, 1, 2, 6)
```

The unique corpuses are selected by removing all duplicate data of  $n$ -bars. All  $n$ -bar mapping data from all compositions are grouped by class. Furthermore, the duplicate data found in the same class or in different classes are removed. This will result unique corpuses in each group. The similarity between compositions is one of the characteristics of *gamelan* music [25]. Therefore, many of the same bars are found in a composition, different compositions, including compositions from different musical mode systems. The similarity of *gamelan* compositions becomes a problem for the network in learning the melodic pattern. The network will confuse to discriminate if multiple inputs are found in more than one class. Increasing the number of bars in mapping data is a solution to get unique corpuses on each class.

The quality of the data was controlled by unique corpuses of data mapping and the distribution proportion of data for training and evaluation taking into account the classifier's task of discriminating musical mode systems at the initial bars of the composition. Analysis on the dataset shows that the number of bars or 1-bar mapping collected from 35 compositions in the class *manyura* is 720, and there are 146 bars remaining after removing duplicate bars. Whereas in the



The binary representation was conducted to all compositions data in 2-bars and 4-bars mapping. Further the results of the binary representation conducted using MATLAB were then exported into Microsoft Excel document format based on the corpus division described in the Implementation and Evaluation section. The exported data is a public dataset called *Gamelan Musical Genre Classification* (GMGC) which can be downloaded through [www.gamelanresearch.com](http://www.gamelanresearch.com) or in the link listed after the Conclusion and Future Works section.

## VII. IMPLEMENTATION & EVALUATION

*Gamelan* music genre classification was developed using the FFNN method. The dataset consists of 105 *gamelan* compositions of the *slendro* musical scale system. The compositions were collected from three classes of the *manyura*, *nem* and *sanga* musical mode systems, where each class consists of 35 compositions. A total of 720 bars were collected from the class *manyura*, 784 bars were collected from the class *nem* and 764 bars were collected from the class *sanga*. Melodic pattern based on 2-bars mapping produces 360 corpuses from 720 bars in the class *manyura*, 392 corpuses from 784 bars in the class *nem* and 382 corpuses from 764 bars in the class *sanga*. Meanwhile 4-bars mapping produces 180, 196 and 191 corpuses respectively.

The task of the classifier is to discriminate the musical mode systems at the initial bars of the composition. The corpus which contains the first initial bars of the compositions were divided for training and unseen data evaluation. Each class contributes a total of 35 these corpuses. The corpus which contains the first initial bars of the compositions were organized separately. All of the corpus which contains the first initial bars of the compositions were separated. After this separation and duplicate bars removal, 2-bars mapping produces 158 unique corpuses in the class *manyura*, 119 unique corpuses in the class *nem* and 163 unique corpuses in the class *sanga*. Meanwhile 4-bars mapping produces 144, 149 and 159 unique corpuses respectively.

The corpus which contains the first initial bars of the compositions were and divided into 70% for sending back to training data and 30% for unseen data evaluation. So, a total of 25 corpus per class was sent back to the training data. This resulted 2-bars mapping consisting of 183 corpuses in the class *manyura*, 144 corpuses in the class *nem*, and 188 corpuses in the class *sanga*. While 4-bars mapping consisted of 169, 174, 184 corpuses respectively. The remaining 30 corpuses which contains the first initial bars of the compositions, both in both 2-bars and 4-bars mapping, were used for unseen data evaluation

The input vector length for 2-bars mapping is 48 bits, and 3 bits for the target. Data of 2-bars mapping were transposed into a 48 x 515 matrix for input with a 3 x 515 matrix for the target. While 4-bars mapping contains 96 bits length for the input and 3 bits for the target. Data of 4-bars mapping were transposed into a 96 x 527 matrix for input with a 3 x 527 matrix for the target. Furthermore, the corpus was randomly

divided into 80% for training data, 10% for validation data and 10% for testing data. Continuing to the unseen data, in the 2-bars mapping, the input data was transposed into a 48 x 30 matrix and the target data was transposed into a 3 x 30 matrix. While the 4-bars mapping consisted of a 96 x 30 matrix for input with a 3 x 30 matrix for the target. Table 6 shows the corpus division for training, validation and testing data in 2-bars mapping, while Table 7 shows the corpus division in 4-bars mapping.

**Table 6. Corpus division in 2-bars mapping**

Class	Tr.	Val.	Test	Total
<i>Manyura</i>	146	18	18	183
<i>Nem</i>	115	14	14	144
<i>Sanga</i>	150	19	19	188
	412	52	52	515

**Table 7. Corpus division in 4-bars mapping**

Class	Tr.	Val.	Test	Total
<i>Manyura</i>	135	17	17	169
<i>Nem</i>	139	17	17	174
<i>Sanga</i>	147	18	18	184
	421	53	53	527

Two different FFNN classifiers with three layers and back-propagation learning algorithm was developed. The input layer for the network which consists of 2-bars mapping is 48 units, while 4-bars mapping is 96 units. Both of the networks consist of three classes as output. The number of hidden layers was determined by trying several different numbers until the best number was obtained. The network which uses 2-bars mapping has the best number of hidden layers of 30 units, while 4-bars mapping has 20 units. Figure 4 and Figure 5 shows the FFNN architecture comparison between 2-bars and 4-bars mapping.

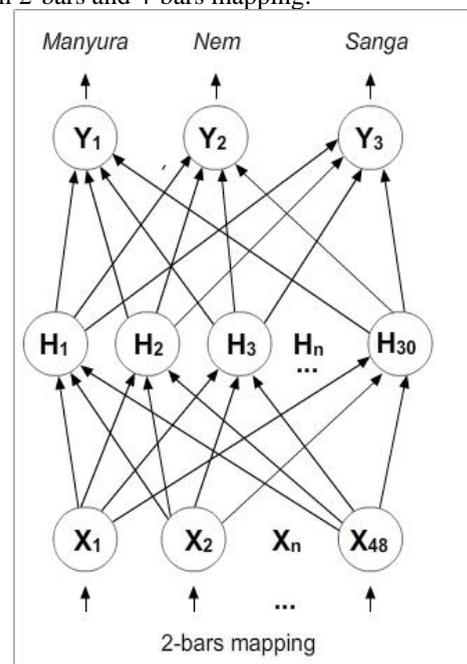


Figure 4. Networks architecture for 2-bars

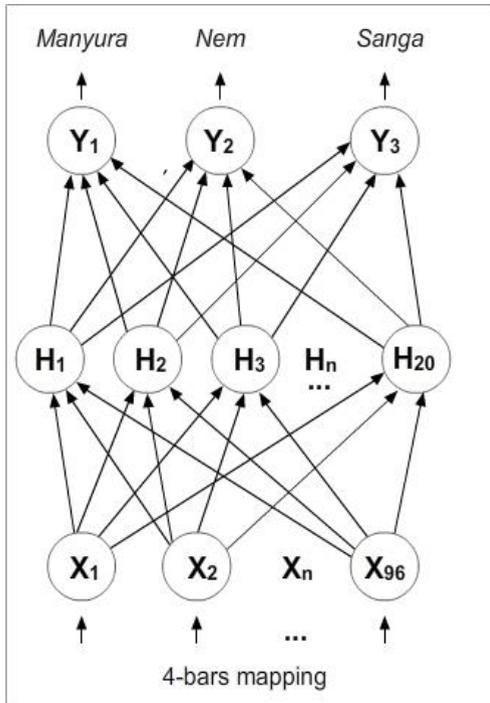


Figure 5. Networks architecture for 4-bars

The experimental results of the network which uses 2-bars mapping reaches the best validation performance value of 0.29588 at epoch 9, while confusion matrix evaluation shows the accuracy of the network which uses 2-bars mapping reaches 65.2% for training, 65.4% for validation and 61.5% for testing. Overall, the accuracy reached 68.7%, where 354 out of 515 corpses were correctly discriminated (shown in Figure 6).

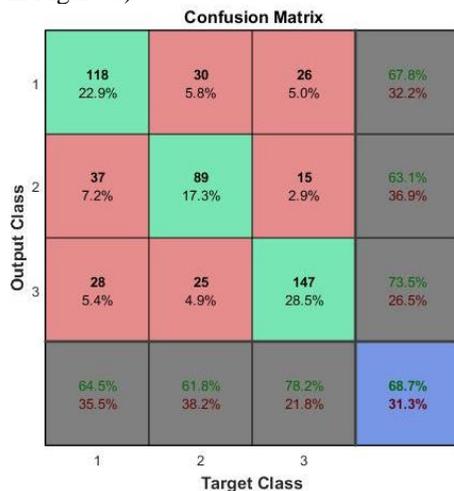


Figure 6. Confusion matrix of the network which uses 2-bars mapping

The experimental results of the network which uses 4-bars mapping reaches the best validation performance value of 0.25922 at epoch 9, while confusion matrix evaluation shows the accuracy of the network containing 4-bars mapping reached 76.5% for training, 73.6% for validation

and, 77.4% for testing. Overall, the accuracy reached 76.3%, where 402 out of 527 corpses were correctly discriminated (shown in Figure 7).

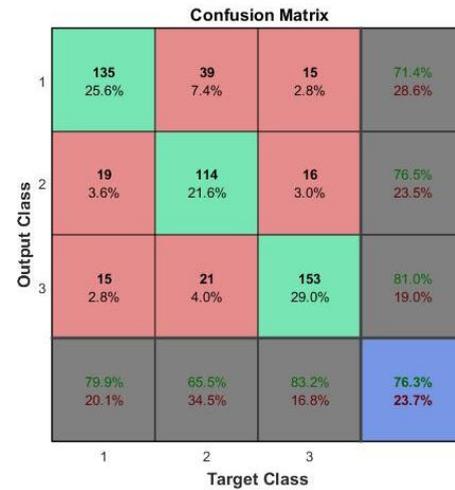


Figure 7. Confusion matrix of the network which uses 4-bars mapping

Evaluation was conducted using unseen data for both of the network containing 2-bars and 4-bars mapping. Unseen data, which were randomly selected, consist of 30 corpses containing the first initial bars of the compositions. The experimental results show that the accuracy of the two networks is increasing. The accuracy of the network containing 2-bars mapping reached 73.3%, where 22 out of 30 corpses were correctly discriminated. While the accuracy of the network containing 4-bars mapping reached 86.7%, where 26 out of 30 corpses were correctly discriminated.

Table 8. Accuracy comparison between 2-bars and 4-bars mapping

Data Mapping	Training	Evaluation
2-bars mapping	68.7%	73.3%
4-bars mapping	76.3%	86.7%

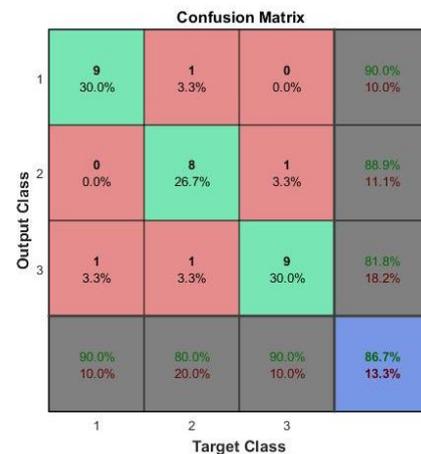


Figure 8. Confusion matrix of evaluation (unseen) data of the network which uses 4-bars mapping

Table 8 shows the accuracy comparison on training and evaluation (unseen) data between the networks which use 2-bars and 4-bars mapping, and Figure 8 shows the results of confusion matrix of evaluation data for 4-bars mapping.

### VIII. CONCLUSION & FUTURE WORKS

A classifier to discriminate *gamelan* music genre was developed using FFNN method. The task of the classifier is to learn melodic patterns in order to discriminate compositions into three classes of musical mode systems namely *manyura*, *nem* and *sanga*. The challenge is to classify between the initial melody without having to analyze the whole melody using a small amount of data. Skeletal melody of *gamelan* compositions that are forms of feature extraction data that are readily available were used as the input. 105 compositions were collected from three classes, where each class consists of 35 compositions. One-to-many relationship cardinality was used to control unique corpus, where a class can have more than one corpus, but a corpus must be linked to one class. Data mapping was used to increase number of unique corpus, where 2-bars and 4-bars mapping were determined to be learned by two different networks.

The experimental results show that 4-bars mapping has a greater number of unique corpus and a better proportions of dataset division than 2-bars mapping. 4-bars mapping results 527 corpuses, while 2-bars mapping results 515 corpuses. The number of corpus in each class in 4-bars mapping is on average better distributed than 2-bars mapping. The difference between the smallest and largest number of corpus in 4-bars mapping is 15 corpuses, while in 2-bars mapping is 39 corpuses. A number of the initial bars of the compositions were used as the unseen data to evaluate the accuracy of the networks. The accuracy of the network containing 2-bars and 4-bars mapping based on the unseen data evaluation are higher than their training.

4-bars mapping accuracy outperforms 2-bars mapping. On the other hand, the network requires four bars, twice the length of two bars, to discriminate the musical mode system of *gamelan* compositions. Increasing the number of *gamelan* compositions for dataset may increase the accuracy and shorten the duration in classifying. Overall, the FFNN method using a bar-based data mapping is suitable for a fast recognition of the *gamelan* music genre classification, in which the networks are trained using a small amount of data.

The proposed method is different with others musical genre classification methods which has been proposed in the previous works by other researchers. A bar-based mapping to organize the entire melody for increasing the number of corpuses from a small number of data is used to train the networks to recognize the musical mode system of compositions between the initial melody without having to analyze the entire melody. The data from a sheet music collection are used to train the networks rather than data from an audio signal extraction. The proposed method gives advantages in learning a pattern from the original source, and the learning results can be implemented for a music genre classification using text-based data or audio signal data.

The proposed method can only be applied to 4-bar data conditions starting from the first bar, and it remains to be tested for its ability to receive 4-bar data starting from the second bar or the next bar. The use of audio data including the audio feature extraction into skeletal melody data also will be conducted in the future works, and also the performance improvement of the network developed in this research to classify the musical mode system using the skeletal melody extracted from audio data.

### References

- [1] J. Ens, P. Pasquier, "Quantifying musical style: Ranking symbolic music based on similarity to a style," *Proceedings of the International Symposium on Music Information Retrieval*, Delft, Netherlands, November 4-6, 2019, pp. 870-877.
- [2] N. Prabhu, A. Asnodkar, R. Kenkre, "Music genre classification using improved artificial neural network with fixed size momentum," *International Journal of Computer Applications*, vol. 101, no. 14, pp. 25-30, 2014. <https://doi.org/10.5120/17756-8857>.
- [3] M. Haggblade, Y. Hong, K. Kao, *Music Genre Classification*, 2011, [Online]. Available at: <http://cs229.stanford.edu/proj2011/HaggbladeHongKao-MusicGenreClassification.pdf>.
- [4] D. Imran, H. Wadiwala, M.A. Tahir, M. Rafi, "Semantic feature extraction using feed-forward neural network for music genre classification," *Asian Journal of Engineering, Sciences & Technology*, vol. 2, no. 1, pp. 1-6, 2017.
- [5] N. Pelchat, C.M. Gelowitz, "Neural network music genre classification," *Proceedings of the 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE)*, Edmonton, AB, Canada, May 5-8, 2019, pp. 1-4. <https://doi.org/10.1109/CCECE.2019.8861555>.
- [6] A.I. Tamboli, R.D. Kokate, "An effective optimization-based neural network for musical note recognition," *Journal of Intelligent Systems*, vol. 28, no. 1, pp. 173-183, 2019. <https://doi.org/10.1515/jisys-2017-0038>.
- [7] G. Campobello, D. Dell'Aquila, M. Russo, A. Segreto, "Neuro-genetic programming for multigenre classification of music content," *Applied Soft Computing Journal*, vol. 94, pp. 1-13, 2020. <https://doi.org/10.1016/j.asoc.2020.106488>.
- [8] A. Shakya, B. Gurung, M.S. Thapa, M. Rai, B. Joshi, *Music Classification Based on Genre and Mood*, in: J. Mandal, P. Dutta, S. Mukhopadhyay (eds), Computational Intelligence, Communications, and Business Analytics. CICBA 2017. Communications in Computer and Information Science, vol 776, Springer, Singapore, 2017, pp. 168-183. [https://doi.org/10.1007/978-981-10-6430-2\\_14](https://doi.org/10.1007/978-981-10-6430-2_14).
- [9] Y.-S. Seo, J.-H. Huh, "Automatic emotion-based music classification for supporting intelligent IoT applications," *Electronics*, vol. 8, no. 2, pp. 1-20, 2019. <https://doi.org/10.3390/electronics8020164>.
- [10] B. Xie, J.C. Kim, C.H. Park, "Musical emotion recognition with spectral feature extraction based on a sinusoidal model with model-based and deep-learning approaches," *Applied Sciences*, vol. 10, no. 3, pp. 1-11, 2020. <https://doi.org/10.3390/app10030902>.
- [11] J. Yoon, H. Lim, D.-W. Kim, "Music genre classification using feature subset search," *International Journal of Machine Learning and Computing*, vol. 6, no. 2, pp. 134-138, 2016. <https://doi.org/10.18178/ijmlc.2016.6.2.587>.
- [12] T.L.H. Li, A.B. Chan, A.H.W. Chun, "Automatic musical pattern feature extraction using convolutional neural network," *Proceedings of the International Multi Conference of Engineers and Computer Scientist 2010*, Hongkong, March 17-19, 2010, pp. 546-550.
- [13] P. Hamel, D. Eck, "Learning features from music audio with deep belief networks," *Proceedings of the 11<sup>th</sup> International Society for Music Information Retrieval Conference (ISMIR 2010)*, Utrecht, Netherlands, August 9-13, 2010, pp. 339-344.
- [14] J. Lee, M. Lee, D. Jang, K. Yoon, "Korean traditional music genre classification using sample and MIDI phrases," *KSII Transactions on Internet and Information Systems*, vol. 12, no. 4, pp. 1869-1886, 2018. <https://doi.org/10.3837/tiis.2018.04.026>.

- [15] S. Jothilakshmi, N. Kathiresan, "Automatic music genre classification for Indian music," *Proceedings of the 2012 International Conference on Software and Computer Applications (ICSCA 2012)*, Singapore, June 9-10, 2012, pp. 55-59.
- [16] M.A. Al Mamun, I. Kadir, A.S.A. Rabby, A. Al Azmi, "Bangla music genre classification using neural network," *Proceedings of the 8<sup>th</sup> International Conference System Modeling and Advancement in Research Trends (SMART)*, Moradabad, India, November 22-23, 2019, pp. 397-403. <https://doi.org/10.1109/SMART46866.2019.9117400>.
- [17] S. Grollmisch, E. Cano, F.M. Angel, G.L. Gil, "Ensemble size classification in Colombian Andean string music recordings," *Proceedings of the 14th International Symposium of Computer Music Multidisciplinary Research*, Marseille, France, October 14-18, 2019, pp. 565-576.
- [18] S. M. H. Mousavi, V. B. S. Prasath, S. M. H. Mousavi, "Persian classical music instrument recognition (PCMIR) using a novel Persian music database," *Proceedings of the 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, Mashhad, Iran, October 24-25, 2019, pp. 122-130. <https://doi.org/10.1109/ICCKE48569.2019.8965166>.
- [19] K. Hastuti, A. Azhari, A. Musdholifah, R. Supanggih, "Building melodic feature knowledge of gamelan music using apriori based on functions in sequence (AFiS) algorithm," *International Review on Computers and Software*, vol. 11, no. 12, pp. 1127-1137, 2016. <https://doi.org/10.15866/irecos.v11i12.10841>.
- [20] J. Becker, A. Becker, "A grammar of the musical genre srepegan," *Asian Music*, vol. 14, no. 1 pp. 30-73, 1982. <https://doi.org/10.2307/834043>.
- [21] D.W. Hughes, "Deep structure and surface structure in Javanese music: A grammar of gending lampah," *Ethnomusicology*, vol. 32, no. 1, pp. 23-74, 1988. <https://doi.org/10.2307/852225>.
- [22] K. Hastuti, A. Azhari, A. Musdholifah, R. Supanggih, "Rule-based and genetic algorithm for automatic gamelan music composition," *International Review on Modelling and Simulations*, vol. 10, no. 3, pp. 202-212, 2017. <https://doi.org/10.15866/iremos.v10i3.11479>.
- [23] H. Hild, J. Feulner, W. Menzel, "HARMONET: A neural net for harmonizing chorales in the style of J.S. Bach," *Proceedings of the Advances in Neural Information Processing Systems 4 (NIPS)*, Denver, Colorado, USA, December 2-5, 1991, pp. 267-274.
- [24] A.M. Syarif, A. Azhari, S. Suprpto, K. Hastuti, "Human and computation-based musical representation for gamelan music," *Malaysian Journal of Music*, vol. 9, pp. 82-100, 2020. <https://doi.org/10.37134/mjm.vol9.7.2020>.
- [25] Martopangrawit, *Pengetahuan Karawitan II*, ASKI Surakarta, Indonesia, 1972, 59 p. (in: Indonesian)
- [26] P.M. Todd, "A connectionist approach to algorithmic composition," *Computer Music Journal*, vol. 13, no. 4, 1989, pp. 27-43. <https://doi.org/10.2307/3679551>.
- [27] F. Colombo, S.P. Muscinelli, A. Seeholzer, J. Brea, W. Gerstner, "Algorithmic composition of melodies with deep recurrent neural networks," *Proceedings of the 1st Conference on Computer Simulation of Musical Creativity*, University of Huddersfield, UK, June 17-19, 2016, pp. 2-12.

#### Dataset link:

The dataset of *Gamelan Musical Genre Classification (GMGC)* can be downloaded in this link: <https://drive.google.com/drive/folders/1504nL8y50tjJiEtlwt4DQI9TwKsMQnT2?usp=sharing>.



**KHAFIIZH HASTUTI** received her doctoral degree from the Computer Science Doctoral Program of Universitas Gadjah Mada, Indonesia. She is currently teaching at Department of Computer Science Universitas Dian Nuswantoro. Her research interests are algorithmic composition, music information retrieval, software engineering and project management.



**PULUNG NURTANTIO ANDONO** received his doctoral degree from Institut Teknologi Sepuluh November, Indonesia. He is currently teaching at Faculty of Computer Science Universitas Dian Nuswantoro, Indonesia. His research interests are machine learning, multimedia programming, computer vision and software engineering.



**ARRY MAULANA SYARIF** received his Master degree from Universitas Dian Nuswantoro. He is currently teaching at Faculty of Computer Science Universitas Dian Nuswantoro, Indonesia. His research interests are algorithmic composition, music information retrieval, artificial intelligent, machine learning and multimedia programming.



**AZHARI AZHARI** received his doctoral degree from the Computer Science Doctoral Program of Universitas Gadjah Mada, Indonesia. He is currently teaching at Department of Computer Science and Electronics Universitas Gadjah Mada. His research interests are Intelligent agent, algorithmic composition, music information retrieval, software engineering and project management.

...