Gamelan Demung Music Transcription Based on STFT Using Deep Learning

Andi Rokhman Hermawan
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
andi11@mhs.ee.its.ac.id

Eko Mulyanto Yuniarno
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
ekomulyanto@ee.its.ac.id

Diah Puspito Wulandari
Department of Electrical Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
diah@ee.its.ac.id

Abstract—**Learning to play a gamelan instrument would be easier when there's a musical notation guide. The process of converting a musical signal into a notation guide is called transcription. In this paper, we would like to transcript the gamelan music especially the Demung instrument using the Deep Learning method. Each Demung's note from 6-low until 1-high would be converted to the time-frequency domain using STFT (*Short-Time Fourier Transform*). Then, those data will be treated as an input for the multilayers perceptron. The training method is a single label of each notation. The output returned by the model is a music roll transcription.**

Index Terms—**Gamelan, Notation Recognition, Deep-Learning, Automatic Music Transcription.**

I. INTRODUCTION

One of Indonesia's traditional music instrument is Gamelan. The Gamelan music is a culture and a heritage that need to be preserved. This cultural asset has existed for centuries in the Indonesia. The Gamelan instrument is not only famous in the Indonesia, but also in international countries [1].

The current issue in this globalized world regarding the Gamelan music is lack of devootes as well as the reducing of the gamelan artisan. Learning to play Gamelan music is one of a way to preserve the art and culture. Music transcription could be very helpful for a beginner who wants to study gamelan music. Generally, only artists, gamelan teacher or people that have a lot of experience in the gamelan field are capable of transcribing the gamelan music. Gamelan teacher teach their student is dependent on memory. Because they teach them them trough oral instead of according the standard notation [2].

The automatic music transcription is a branch of music information retrieval science that could help a person to study and learn the music with a help of technology. Because the automatic music transcription technology is a conversion process of a song or a digital audio music signal into a symbolic notation, such as a score or a MIDI file [3].

We would like to explain the Gamelan instrument that will be used in this research. The Gamelan consists of 15 instuments and it has monophonic character [4]. One of the Gamelan instrument type is *Balungan* group. The *Balungan* group is a percussion instrument and played using a mallet. There are 3 instuments in the Balungan group. Demung is one of the *Balungan* music instruments that made of copper

metal. The other *Balungan* music instrument are Peking and Saron [5]. For the physical size, the Demung has the biggest metal bar among all of them. Hence, it produces the lowest frequency as well.

Gamelan is an unique eastern musical instrument. Compared to the western musical instrument, the same Demung instrument has a slightly different frequency, resonance, and amplitude. It is because the production of Gamelan is hand-made by the artisans [6] and the tuning practice isn't based on mathematical purity of interval vibration ratio [7]. Therefore, music transcription for the gamelan instrument is interesting and challenging.

In this research, we will perform the transcription process on a Demung instrument. The appearance of Demung instrument can be seen in the Fig. 1. Demung has the same shape as Saron or Peking, but the size is bigger. The instrument type that would be used is The *Slendro* type. The *Balungan* instruments is separated into two type of tuning system, the *Pelog* and the *Slendro*. The *Slendro* has a pentatonic tuning, whereas the *Pelog* has a heptatonic tuning [8]. The pentatonic tuning doesn't have the fourth tone and and the seventh tone.

Fig. 1. The Demung, Slendro type

The research on automatic music transcription has been performed by several researchers. Generally there are 2 type of methods to achieve AMT (*Automatic Music Transcription*), the features engineering and machine learning. The research about music transcription using STFT (*Short-Time Fourier Transform*) for Saron Slendro and Peking Slendro instrument

[9] can be categorized as feature engineering AMT. The STFT is doing Fourier Transform iteratively in a single sound wave. Then, as a result, it returns a spectrogram that has the information of time-frequency. The result of that research was signal envelope from separation process the music signals into some channels of music notation by the fundamental frequency range of each music notation.

There was another research about Music transcription to detect Saron instrument using HMM (*Hidden Markov Model*) [10]. The method is using HMM probability value of observation, the initial probability, and transition probabilities. The result is 95% performance for the single instrument.

In the machine learning field, there was a research about music transcription using *Backpropagation Neural Network* in 2007 [11]. The instrument is piano and their research shows 98.6% accuracy when generating notation of music. The STFT has been used as a feature for their neural network and has been able to detect iteration of notes and the onset position.

There was another research using Backpropagation Neural Network as well for gamelan instruments [12]. Their gamelan instruments are gambang, kenong, gong and saron. The research was not classifying the musical notes, but the musical instrument. The features for their research are the Short Time Energy (STE) that marks the violence of the voice at a short time and the Zero Crossing Rate (ZCR) denotes a sequential sample on a digital signal. Their result of the evaluation was reached 82.5% accuracy.

Furthermore, the multi-label and multi-instrument research was also performed for training the neural network [13]. The instruments is Peking, Demung, and Saron. The average accuracy result on their research was 96.58%. The method also involves the STFT as a feature for the neural network. Based on those 2 successful results, this research is aiming to use the STFT and Deep Neural network on the music transcription. But, the training for this method will still use the single label for each note and a single instrument, the Demung itself.

The spectrogram result from STFT has been used as the input for the neural network by many researchers. Because the spectrogram captures the information about time and frequency in a single package. Raw audio can be used as the input as well. But the data only contains the information of alternation of amplitude over time. There are two categories of STFT, the standard or Non-Overlapped STFT, and Overlapped-STFT. The main difference in the overlapped version is using hop value to determine the start of Fourier Transform computation instead of using the end of the window's length.

The data in this research consists of 450 pieces of Demung sound from 6-low key to 1-high key and recorded in the 44,100 rates. So it would have 7 labels or categories of data. Each data is a single Demung note. It has various lengths from 0.25 seconds until 2.5 seconds. The distribution of each key is not equal as well. The most data is on Demung 2-key which consists of 126 data and the least data is on Demung 1-key which consists of 39 data. In the Fundamental Frequency perspective, the Demung 6-low key to Demung 1-high has the following frequencies [13]. The Demung 6-low note is 231 Hz,

Demung 1 note is 267 Hz, Demung 2 note is 307 Hz, Demung 3 note is 349 Hz, Demung 5 note is 402 Hz, Demung 6 note is 463 Hz, and Demung 1-high note is 533 Hz.

II. METHODS

We divide this research into two parts, the first part is training the model and the second part is creating the music roll representation. The first part of this research is explained by the block diagram in Fig. 2. The second part is testing the model that explained by the Fig. 3, the process is used in the result and discussion of section III. Each parts in those chart will be explained in this section.
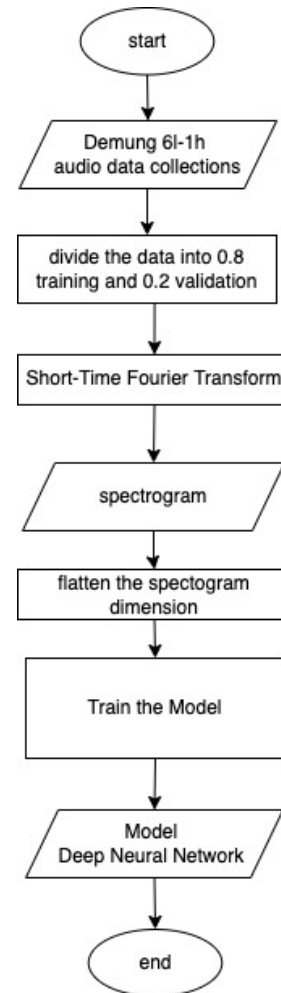

Fig. 2. Block Diagram for training the Model

A. Data Collections

As explained in the previous section, the input data for building the model is 450 pieces. We would like to divide it into training data and validation data. Those data will be used in the training process of the deep neural network. To divide it correctly, we must categorize it by the note first. This is because we need to have exactly 20% validation data in each
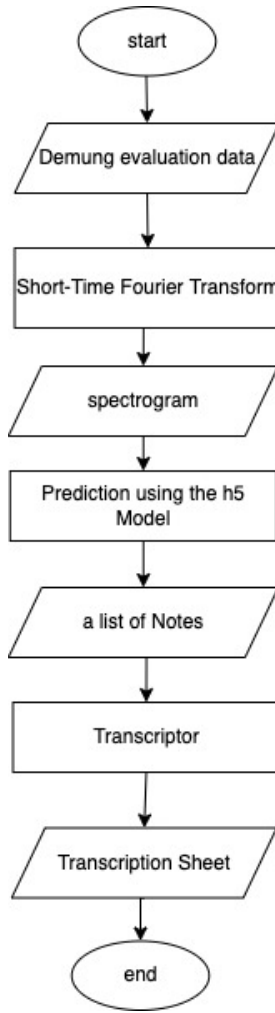
Fig. 3. Block diagram for testing the model

TABLE I. DATA COLLECTION

note	total	training	validation
6 low	42	33	9
1	39	31	8
2	126	100	26
3	63	50	13
5	68	54	14
6	71	56	15
1 high	45	36	9
total	*454*	*360*	*94*

Demung note. When it didn't categorized by the key, it has possibility that one of the keys doesn't have a validation data.

The result of the separated data collection is as shown in Table I. The data for training is 360 pieces and the data for validation is 94 pieces. Some notes have larger validation data e.g the Demung 2 note, but it's already balanced to 20% for the other notes as well. Those data are the time-domain audio data and will be converted to the time-frequency domain data using STFT.
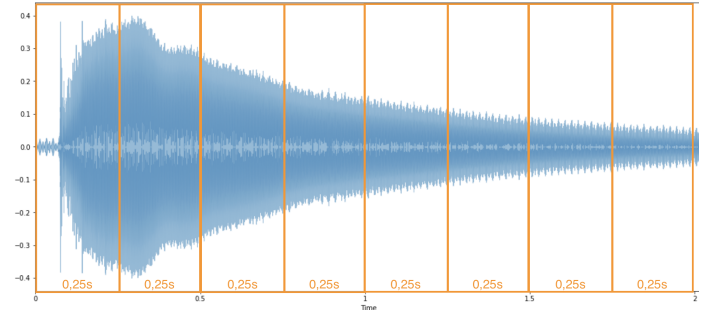
B. STFT Configuration

Fig. 4. The demung segment

The STFT (*Short-Time Fourier Transfer*) is a method to convert the time domain signal into a time-frequency signal. This section will explain the STFT part both in the   Fig. 2 and   Fig. 3 . The intuition of STFT is to FFT the signal in each exact length iteratively. The STFT function can be seen in the equation (1). The result of that function depends on the *m* or the frame and the k or *frequency* because the output of STFT is a spectral matrix with frequency bins and frames.

The function sums all the samples, the *N* is the number of frame size. The *x(n)* is the signal that present in the current frame. There, the *mH* is the starting sample of the current frame. The *H* is the hop size. The *w(n)* is the windowing function and the last part of the equation is the same as the Discreete Fourier Transform function. In other words, it is multiplying by a pure tone that has a frequency given by *k* divided by *N*. It is taking the signal, then decomposing it and projecting it onto pure tone with frequency with *k* divided by *N*.

$$S(m, k) = \sum_{n=0}^{N-1} x(n + mH) \cdot w(n) \cdot e^{-i2\pi n \frac{k}{N}} \qquad (1)$$

The frequency sampling to be used in this STFT is 22,010 Hz. It means there are 22,010 data points from the raw audio signal in each second of the audio. The frequency sampling selection is the important part before choosing another parameter for the STFT configuration.

The STFT in this method is the overlapped version. The window length value is 2,048, and the hop is 128. From the window length, the max frequency is known by the equation (2). For example, the result of the frequency range or the frequency bins is 0 to 1,025.

$$max_{frequencyBin} = \frac{windowlength}{2} + 1 \qquad (2)$$

A segment is a length of signal that will be processed using STFT. We use 4 segments each mninute. So there would be 4 (n) times STFT in each second. The length of the audio signal for the STFT is the division of frequency sampling and segment. For this research, the length is 5,502. Hence, the

input of the STFT is 5,502 samples or equal to 0.25 seconds of sounds. The input can be seen in the equation (3). For example, there will be 4 n in the 1 second input. The first input will be the audio length from 0 to 5,502 data points and the 4th n is from 16,506 to 22,008 data points.

$$STFT_{n\ input} = audio[sample \times (n-1), sample \times n] \quad (3)$$

The number of samples, window length, and hop is correlated with the size of frames. To get the value of the frames we can take a look at equation (4). The higher and lower is determining the characteristic of the result. The higher the value of the frames, the higher the frequency resolution but the time resolution is lower. Otherwise, the lower the value of the frames the lower the frequency resolution but the time resolution is higher. for example, the frame size of this research is 27. It is considered as low frame size because on this one instrument music transcription, there is a need for higher time resolution.

$$framesize = \frac{sample - window}{hop + 1} \quad (4)$$

Each segment is a single input data for the Deep Neural Network. On the 2 second audio of a single signal in Figure 4, the Deep Neural Network would have 8 inputs. As a result, the output of the STFT in this research is a 2-dimensional matrix. The output of the STFT is a 2-dimensional matrix with the output as described in equation (5). For example, the shape for this configuration is (1,025; 27)

$$STFT_{shape\ output} = (max_{frequency}, frames) \quad (5)$$

C. Training The Deep Neural Network

We would like to explain the input as well as the output of the model first before training the Deep Neural Network, explaining the architecture, accuracy, and loss. The input is a spectrogram but the standard multi-layer perceptron deep neural network only accepts one-dimensional data.

Flatten function is needed to convert it one-dimensional. Altough the flattened STFT in the research about synthetizer parameter tuning gave poor results due to the large input size [14]. In other music transcription, the flattened STFT is used in their research method and have a decent result [15].

The 2-dimensional matrix in this research with the shape of (1,025; 27) is now converted to a 1-dimensional matrix with the shape of (27,675; 0). The input for the test or evaluation data need to processed with the same way and have the same length.

The output of this neural network should represent 7 notes from Demung 6-low to the Demung 1-high. The 7 notes here

are represented with the array or list. The array has 7 lengths and the data is binary, the zero and one.

The first data is representing the Demung 6-low and the last data is representing Demung 1-high, e.g [0, 0, 0, 0, 0, 0, 1]. From the previous example, that output represent the last key which is the Demung 1-high.
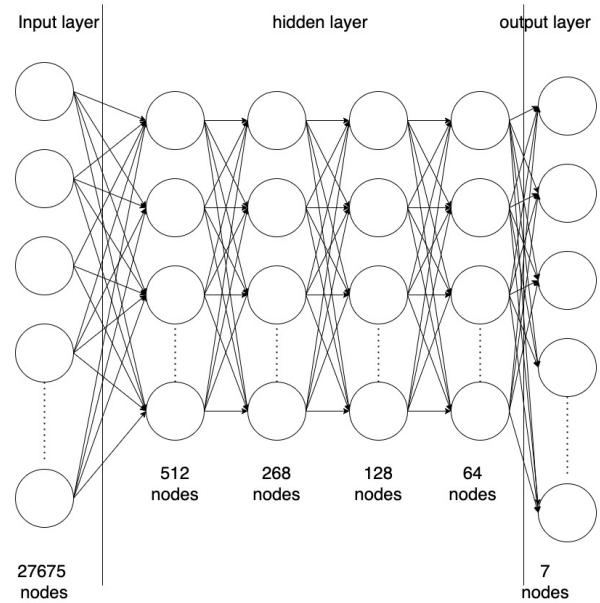

Fig. 5. Architecture of the neural network

Here's the architecture of this deep neural network, the input layer consists of 27,675. The hidden layer consists of 4 layers. The first layer is 512 nodes, the second layer is 265 nodes, the third layer is 128 nodes.

The activation function of those layers is ReLU (*Rectifier Linear Unit*). The output of ReLU is the input itself when the value is positive. otherwise the value is 0 when it's not positive. Given the amount of nodes in this deep neural network, the advantage of using ReLU is less computationally expensive [16]. This is because ReLU's function has a simple mathematical algorithm. ReLU is also rectifies and avoids vanishing gradient problem.

$$ReLU(x) = max(0, x) \quad (6)$$

The fourth layer is 64 nodes. Between the fourth and output layer, there's a Sigmoid function. The Sigmoid function is a form of logistic function that denoted by equation 7. The output of this function is a value in range 0 and 1. The function is good for classifier and it has smooth gradient for deep network [17].

$$sigmoid(x) = \frac{1}{1 + e^{-x}} \quad (7)$$

Outside the architecture configuration, there are some parameters for training the model e.g, epoch, loss algorithm, metrics. 1,000 epoch needed to train the model. The loss algorithm is mean squared error and the metrics that we use are accuracies and binary accuracy.

The hardware specification for training the model is a computer that has Intel Core i5-1038NG7, 16GB of RAM, and the computer doesn't have GPU. The training process is consuming around 20 minutes and the result is displayed in the following Fig. 6.
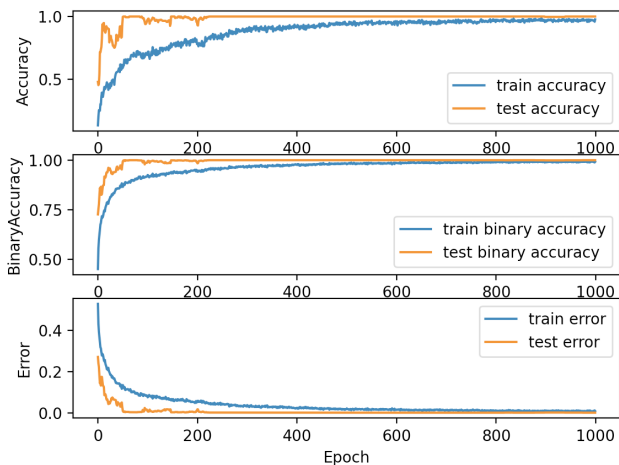

Fig. 6. Training Result

The training result is then stored in an *h5* file or a hierarchical file format as written by the Keras Python library. By using the h5 file format, the model of the deep neural network is reusable. Therefore, it can be loaded and used for the testing or evaluation.

The metric of the model is explained in the following. The training result for loss is 0.0079, the accuracy is 0.9765, the binary accuracy is 0.9920. However, the result when compared to validation data are the following. The validation accuracy is 0.9989, the validation binary accuracy is 0.9998, and the loss is 1.8145e-05.

D. Music Roll Representation

The length input, as well as the output is every 0.25 seconds and the result is rows of text about the prediction. To make the result to be easier to understand, those rows of text data need to be converted to a music roll.

The music roll builder or a transcriptor can be accomplished using a plotter. The output data will be mapped on the 2-dimensional plane. The x-axis is time and the y-axis is the value between 1 to 7 to represent the Demung 6-low to Demung 1-high. We can find the sample of this music roll representation in the next section, e.g Fig. 9.

III. RESULT AND DISCUSSION

The experiment was performed 3 times using the audio music signal of Demung Gamelan Slendro. The evaluation data is not included both in the training data or validation data. The first experiment will be tested using 1 note of acoustic signal. The second experiment will be using the semi-synthetic signal. The semi-synthetic signal is an artificial signal customed with several acoustic signals. Last, the model will be tested using a long acoustic signal of the Demung instrument.
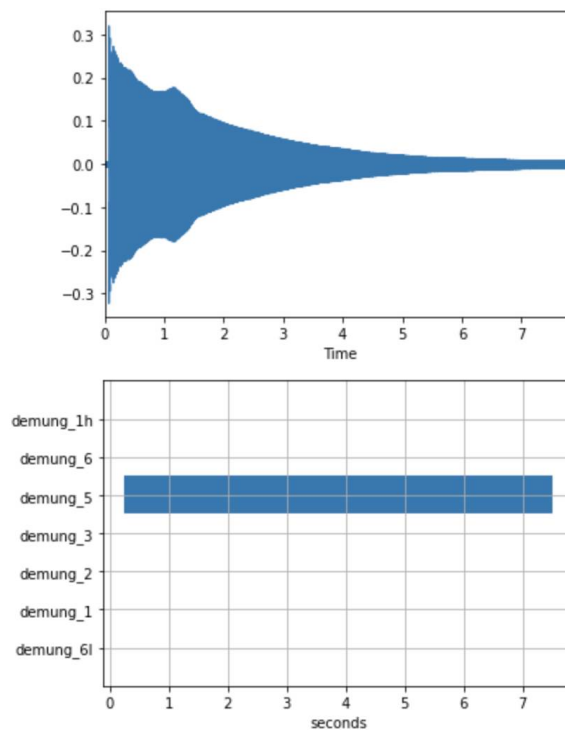

Fig. 7. Experiment 1: Demung 5

The first signal of this experiment is displayed in Fig. 7. It's an acoustic gamelan music audio consisting of 1 note only. The note is Demung 5 and it is *Slendro* type Gamelan instrument.

The result of the first transcription can be seen in Fig. 7 as well. On that figure, we found that the audio signal and the transcription result correctly display the Demung 5. The note error rate is 0%.

The second experiment is testing a semi-synthetic signal. The signal was built manually using 3 Demung audio signals that are not included in the training data as well as the validation data. Those 3 data are the Demung 6, Demung 2, Demung 5. The data spliced together resulting in 10 seconds in the signal length.

The transcription result is also consistent with the content of the second signal in Fig. 8. The model correctly transcript the notes of the Demung with 0% note error rate. It has Demung 6 and then switched to Demung 2 around 4-5 seconds, and then switched to Demung 5.
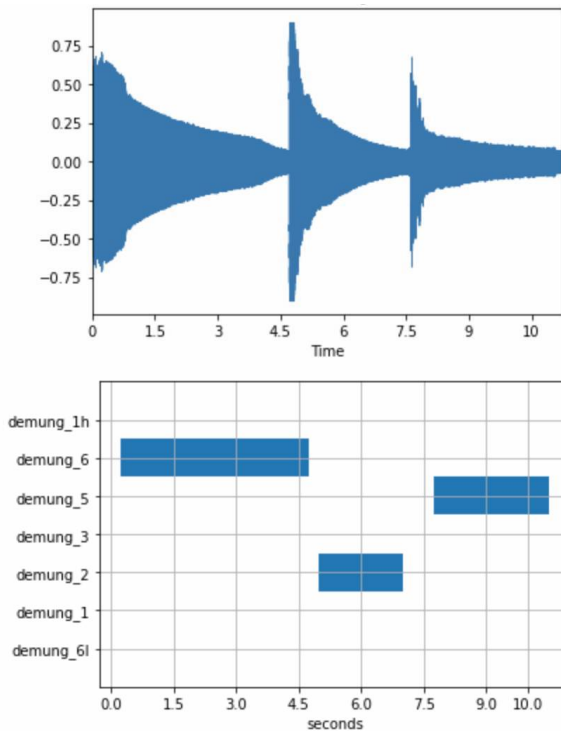
Fig. 8. Experiment 2: Demung 6, Demung 2, Demung 5

Fig. 9. Experiment 3: All Demung notes

The third experiment is a long acoustic gamelan sound consisting of Demung 6-low to 1-high. The music obtained from recorded music of a single Demung instrument that played alternately in 20 seconds. The result is available in the 9. The model correctly transcript the notes as well.

There is one insertion among 41 Demung keys that counted as error in the third experiment. At the time 19.5 seconds, the model predicted double key. the Demung 6-low and Demung 1. The value should be only Demung 1. Because of that, the note error rate is 2.4%. if we take a look at the envelope characteristic of the Demung instrument, it's consists of attakt, decay, sustain and release phase. The error might because of the Demung 6-low is still on the release phase.

IV. CONCLUSION

The research of this paper produces a transcription of gamelan audio using deep neural network based on the STFT value. The STFT has been proven to be a useful input for the deep neural network. This is an alternative method compared to the feature engineering method. The method in this paper eliminate the needs of filtering and thresholding. As comparisson in the [9] research, there is a need to determine initial fixed frequency range each notation and also determine a maximum amplitude value as a treshold value. Segmenting the audio by 0.25 times, and then computing the STFT 4 times every second is also resulting a good music transcription. However, there's still a downside to implementing the music transcription this way. The resolution of this music transcription is only 0.25 seconds. When there are notes played alternately faster than 0.25 seconds the prediction might not be correct. Other than
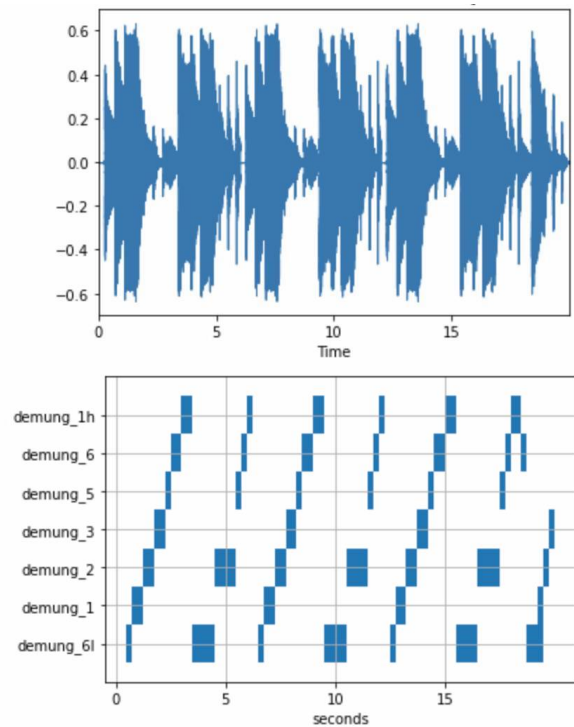
that, adding more Gamelan instruments one by one using this method is also needed for future work if the notation data is available.

REFERENCES

[1] M. A. Maula and H. Setiawan, "Spectrum identification of peking as a part of traditional instrument of gamelan," in *2018 8th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, 2018, pp. 72–77.

[2] Y. Suprapto, E. M. Yuniarno, and K. Fithri, "Gamelan notation generating using band pass filter for saron instrument," in *2019 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2019, pp. 1–6.

[3] M. D. Anssi Klapuri, *Signal Processing Methods for Music Transcription.* "Springer US", 2006.

[4] P. Yampolsky, "Javanese gamelan and the west by sumarsam," *Asian Music*, vol. 49, no. 2, pp. 158–162, 2018.

[5] Y. K. Suprapto, S. M. Ramadhan, and E. Pramunanto, "Gamelan simulator multiplatform application development," in *2018 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM)*, 2018, pp. 227–233.

[6] Y. Triwidyastuti, "Saron music onset detection on gamelan orchestra using hidden markov model for music transcription," in *Institut Teknologi Sepuluh Nopember*, 2013.

[7] A. Tjahyanto, D. P. Wulandari, Y. K. Suprapto, and M. H. Purnomo, "Gamelan instrument sound recognition using spectral and facial features of the first harmonic frequency," *Acoustical Science and Technology*, vol. 36, no. 1, pp. 12–23, 2015.

[8] G. Wendt and R. Bader, "Analysis and perception of javanese gamelan tunings," in *Computational Phonogram Archiving.* Springer, 2019, pp. 129–142.

[9] L. Fitria, Y. K. Suprapto, and M. H. Purnomo, "Music transcription of javanese gamelan using short time fourier transform (stft)," in *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, May 2015, pp. 279–284.

[10] Y. K. Suprapto and Y. Triwidyastuti, "Saron music transcription based on rhythmic information using hmm on gamelan orchestra," *Telkomnika*, vol. 13, no. 1, p. 103, 2015.

[11] Z. Guibin and L. Sheng, "Automatic transcription method for polyphonic music based on adaptive comb filter and neural network," in *2007 International Conference on Mechatronics and Automation*, Aug 2007, pp. 2592–2597.

[12] F. Firdausillah, D. Gilang Mahendra, J. Zeniarja, A. Luthfiarta, H. Agus Santoso, A. Nugraha, E. Yudi Hidayat, and A. Syukur, "Implementation of neural network backpropagation using audio feature extraction for classification of gamelan notes," in *2018 International Seminar on Application for Technology of Information and Communication*, 2018, pp. 570–574.

[13] D. Nurdiyah, Y. K. Suprapto, and E. M. Yuniarno, "Gamelan orchestra transcription using neural network," in *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2020, pp. 371–376.

[14] O. Barkan and D. Tsiris, "Deep synthesizer parameter estimation," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3887–3891.

[15] K. W. Cheuk, K. Agres, and D. Herremans, "The impact of audio input representations on neural network based music transcription," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–6.

[16] D. Stursa and P. Dolezel, "Comparison of relu and linear saturated activation functions in neural network for universal approximation," in *2019 22nd International Conference on Process Control (PC19)*, 2019, pp. 146–151.

[17] A. D. Rasamoelina, F. Adjailia, and P. Sinčák, "A review of activation function for artificial neural network," in *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2020, pp. 281–286.