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. 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.

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 or people that have a lot of experience in the gamelan field are capable of transcribing the gamelan music.

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 [1].

We would like to explain the Gamelan instrument that will be used in this research. 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. 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 handmade by the artisans [2]. Until now, there is no fixed parameter regarding the frequency value in each bar of the Demung. Therefore, music transcription for the gamelan instrument is interesting and challenging.

In this research, we will perform the transcription process on a Demung instrument, especially the *Slendro* type. The *Balungan* instruments is separated into two types, the *Pelog* and the *Slendro*. The *Slendro* is unique because it has a pentatonic scale (1,2,3,5,6), whereas the *Pelog* has a standard heptatonic scale (1,2,3,4,5,6,7) [3]

Figure 1. The Demung, Slendro type

The research on automatic music transcription has been performed by several researchers. Liza Fitria et al [4] researched music transcription using STFT (*Short-Time Fourier Transform*) in 2015. 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. In the machine learning field, [5] Zheng Guibin et al tried music transcription using *Backpropagation Neural Network* in 2007. 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.

The Backpropagation Neural Network research for the Gamelan notes was performed by Firdausillah et al as well [6]. Their feature 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.

Multi-label and multi-instrument research was also performed for training the neural network by Dewi Nurdiyah et al [7]. 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, we will also try to use the STFT and Deep Neural network as well 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 proven as the input for the neural network. 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 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 [7]. 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 Figure 2. The second part is testing the model that explained by the Figure 3, the process is used in the result and discussion of section III.

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

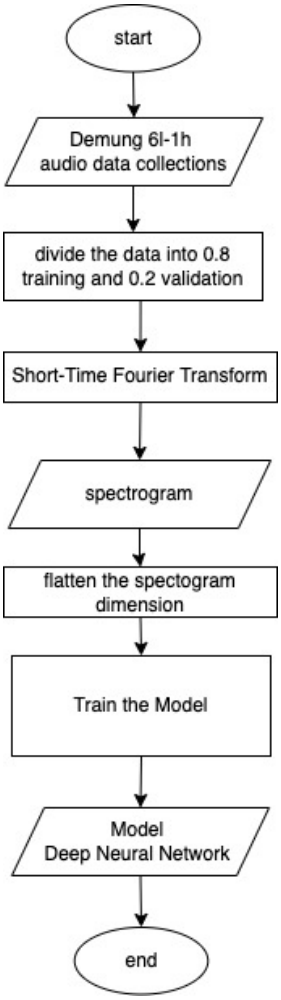

Figure 2. Block Diagram for training the Model

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***

Table I
DATA COLLECTION

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.

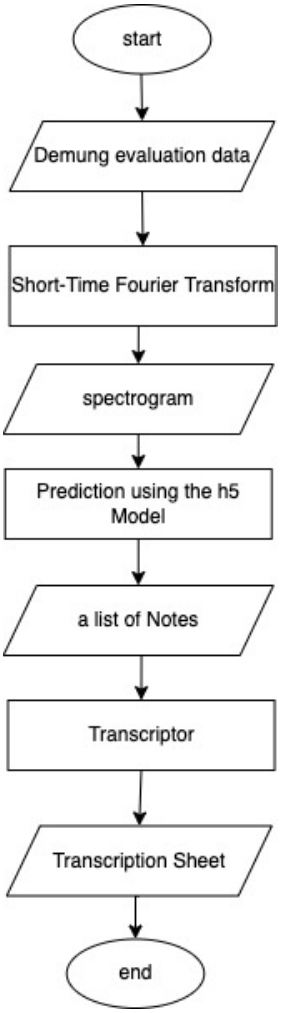

Figure 3. Block Diagram for testing the Model

B. STFT Configuration

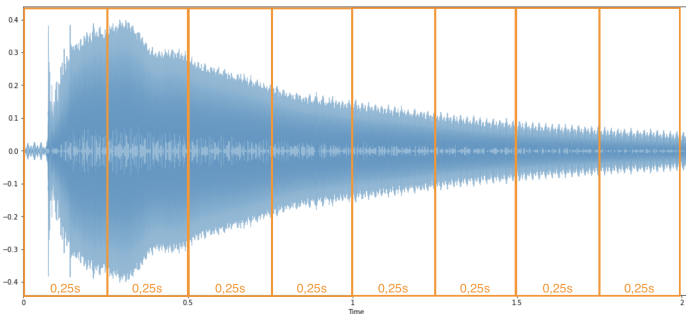

Figure 4. The Demung Segments

The STFT (Short-Time Fourier Transfer) is a method to convert the time domain signal into a time-frequency signal.The intuition of STFT is to FFT the signal in each exact length iteratively. It's explained 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*.

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

The frequency sampling to be used in this STFT is 22.010 Hz. It means there are 22.010 data point 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 2048, and the hop is 128. Those value is chosen because the window length value is 2048 because the result of the frequency range or the frequency bins is 0 to 1025 aligned with the equation (2).

$$max_{frequency} = \frac{windowlength}{2} + 1 \quad (2)$$

A segment is a length of signal that will be processed using STFT. We use 4 segments each mninute. So there would be 4 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 5502. Hence, the input of the STFT is 5502 samples or equal to 0,25second of sounds.

$$STFT_{input} = audio[start_n, 5502] \quad (3)$$

The output of the STFT is a 2-dimensional matrix with the output as described in equation (3). The value of frames is correlated with the number of samples, window length, and hop. To get the value of the frames we can take a look at equation (4). 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.

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

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 with the following shape (1025, 27).

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

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 deep neural network only accepts one-dimensional data.

Flatten function is needed to make it one-dimensional. So the input is following equation (6). The 2-dimensional matrix with the shape of (1025, 27) is now converted to a 1-dimensional matrix with the shape of (27675,). The input for the test or evaluation data need to processed with the same way and have the same length.

$$DNN_{input} = flatten(STFT_{out}) \tag{6}$$

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.

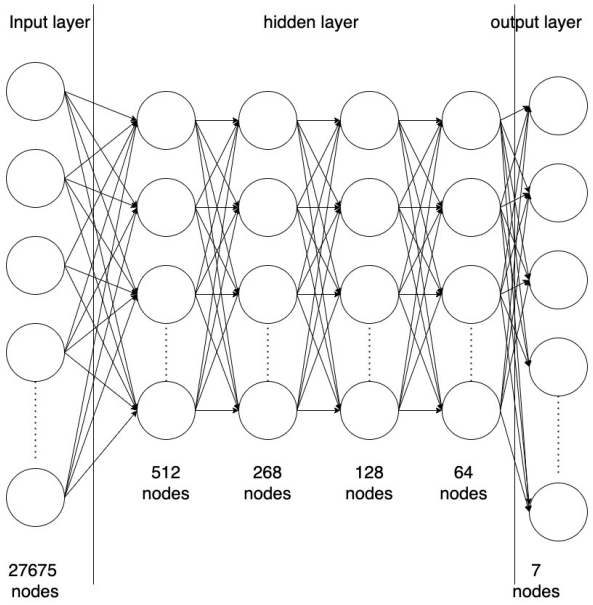

Figure 5. Architecture of The Neural Network

Here's the architecture of this deep neural network, the input layer consists of 27675. It's aligned with equation (6). 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 computational expensive because of the simple mathematical algorithm. ReLU is also rectifies and avoids vanishing gradient problem.

$$ReLU(x) = max(0, x) \tag{7}$$

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 8. The output of this function is a value in range 0 and 1. The function is good for classifier and it has smooth gradient.

$$sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{8}$$

Outside the architecture configuration, there are some parameters for training the model e.g, epoch, loss algorithm, metrics. 1000 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 without GPU. The training process is consuming around 20 minutes and the result is displayed in the following Figure 6.

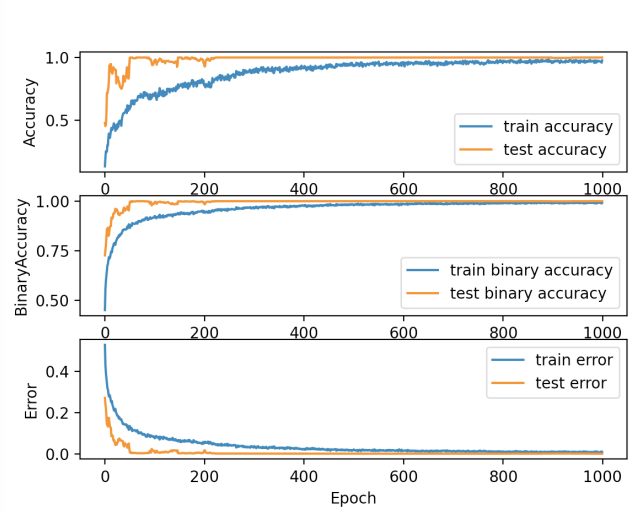

Figure 6. Training Result

The training result is then stored in an *h5* file or a hierarchical file format. By using the h5 file format, the model of the deep neural network is reusable therefore it can be 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 0,25s 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 Figure 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.

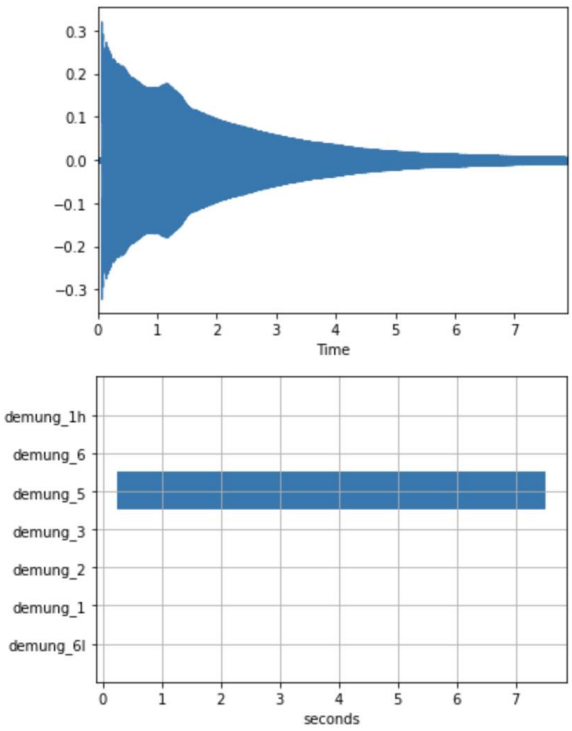

Figure 7. Experiment 1: Demung 5

The first signal of this experiment is displayed in Figure 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 Figure 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%.

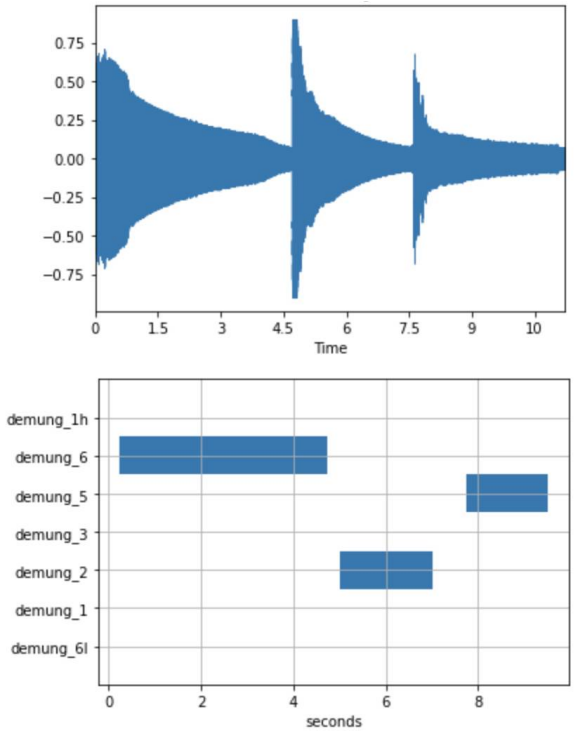

Figure 8. Experiment 2: Demung 6, Demung 2, Demung 5

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 Figure 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.

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 second, 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.

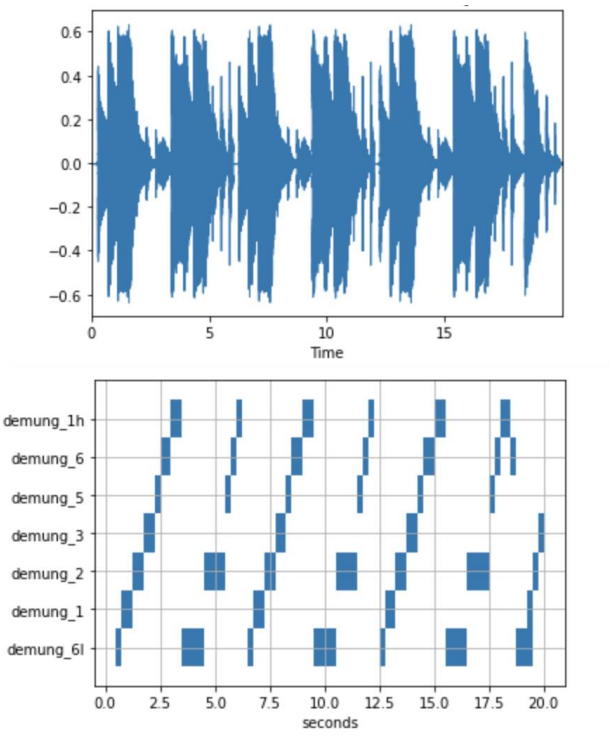

Figure 9. Experiment 3: All Demung notes

IV. Conclusion

The research of this paper produces a transcription of gamelan audio based on the STFT value. The STFT has proven to be a good input for the deep neural network. This is an alternative method compared to using a feature selection as an input for deep neural network. 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. D. Anssi Klapuri, *Signal Processing Methods for Music Transcription*. "Springer US", 2006.

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

[3] Y. K. Suprapto, "Ekstraksi suara saron berbasis spectral - density menggunakan filter multidimensi. institut teknologi sepuluh nopember," 2010.

[4] 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.

[5] 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.

[6] 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.

[7] 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.