Amplitude Spectrogram Prediction from Mel-Frequency Cepstrum Coefficients and Loudness Using Deep Neural Networks

Shoya Kawaguchi[1] and Daichi Kitamura[1]

[1]National Institute of Technology, Kagawa College
355 Chokushi, Takamatsu, Kagawa 761–8058, Japan

Abstract

Timbre conversion of musical instrument sounds utilizing deep neural networks (DNNs) has been the subject of extensive research and continues to elicit significant interest in the development of more advanced techniques. We aim to propose a novel algorithm for timbre conversion utilizing a variational autoencoder. However, this system must possess the capability of predicting the amplitude spectrogram from the mel-frequency cepstrum coefficient (MFCC) and loudness. The present research aims to build a DNN-based decoder that utilizes the MFCC and loudness as inputs to predict the amplitude spectrogram. Experiments using a musical instrument sound dataset indicate that a decoder incorporating bidirectional long short-term memory yields accurate predictions of amplitude spectrograms.

1. Introduction

The generation of musical instrument sounds and the conversion of timbre have been the focus of extensive research in recent years. Various techniques based on deep neural networks (DNNs) have been proposed to tackle this problem. For example, differentiable digital signal processing [1] is a synthesizer of musical instrument sounds, which utilizes multiple sinusoidal waves and filtered noise, whose parameters are generated by pre-trained DNNs. Another approach involves the use of a variational autoencoder (VAE) [2] to generate musical instrument sounds [3, 4, 5]. In this method, VAE is used to extract disentangled latent features of pitch and timbre.

Similar to the aforementioned approach, this paper also focuses on a VAE-based system for the timbre conversion of musical instruments. Currently, we are in the process of developing a new timbre conversion system, as depicted in Fig. 1. In this system, we propose to incorporate the three fundamental elements of musical instrument sounds: pitch, timbre, and volume. To represent these features, we employ traditional, well-established features: note number as a parameter of pitch, mel-frequency cepstrum coefficient (MFCC) [6] as a parameter of timbre, and loudness as a parameter of volume. The timbre conversion of an input sound is achieved by modifying only the MFCC based on the pre-trained VAE. We anticipate that the proposed system will enable the interpolation of timbre across multiple types of musical instruments, thus contributing to the advancement of new art and music.

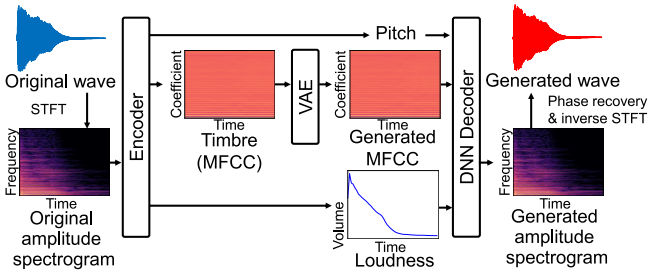

Figure 1: Process flow of proposed timbre conversion system

The proposed system (Fig. 1) requires a decoding process that involves the calculation of an amplitude spectrogram from the pitch, manipulated timbre, and volume. This decoder cannot be realized in an analytical manner. To solve this problem, in this paper, we propose the utilization of three DNN architectures and evaluate the suitable DNN for predicting amplitude spectrograms.

2. Proposed system and its DNN decoder

2.1 Overview of proposed system

As depicted in Fig. 1, the proposed system first computes an amplitude spectrogram of the input sound via a short-time Fourier transform (STFT). Subsequently, the pitch, MFCC, and loudness are extracted. While the pitch can be extracted using various techniques, including DNN-based methods such as [7], the MFCC and loudness are obtained through deterministic calculations. In the proposed system, it is assumed that the MFCC is a reliable feature that represents the timbre of the sound and is disentangled from the pitch and volume. To train the timbre-embedded latent space, the MFCCs of various from multiple musical instruments are input into the VAE. This training enables us to manipulate only the timbre of the sounds after the training of the entire proposed system. Finally, the pitch, manipulated MFCC,

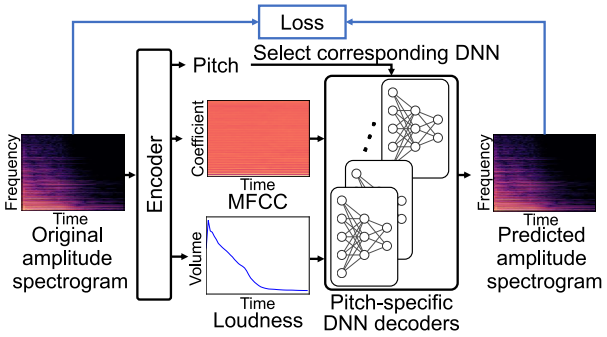

Figure 2: Training process flow of proposed DNN-based timbre decoder

and loudness are decoded to the amplitude spectrogram, and the waveform of the generated sound is obtained via inverse STFT and phase recovery techniques.

2.2 Motivations and contributions of this paper

In the proposed system, an output of the VAE (generated MFCC) must be decoded into an amplitude spectrogram using pitch and loudness. Since MFCC is a dimensionality-reduced feature, this decoding cannot be achieved through linear operations or analytical manners. To circumvent this issue, we employ a DNN as the decoder of the proposed system, namely, the DNN decoder predicts an amplitude spectrogram of synthesized sound from the inputted pitch, MFCC, and loudness. In this paper, we investigate a suitable architecture for this DNN decoder and evaluate its accuracy. The evaluation of the entire system of Fig. 1 is our future work.

The procedure for training the DNN decoder is depicted in Fig. 2. As the decoder, multiple DNNs are independently prepared and trained for each sound note number (from C3 to B5). Thus, the estimated pitch is utilized to select the pitch-specific DNN, and MFCC and loudness are inputted into the selected one. Thanks to this pitchwise model training, a generalization capability for pitch is not required for each DNN, resulting in a high prediction performance.

In this paper, we conduct experimental investigations to determine the suitable DNN architecture for the DNN decoder. The performance of multilayer perception (MLP), bidirectional gated recurrent unit (BiGRU) [8], and bidirectional long short-term memory (BiLSTM) [9] are compared. BiGRU and BiLSTM are referred to as bidirectional recurrent neural networks (BiRNNs) [10], which can effectively capture latent structures of time-series data.

2.3 Architecture of DNN decoders

Let $\boldsymbol{Y} \in \mathbb{R}_{\geq 0}^{I \times J}$ and y_{ij} be the amplitude spectrogram of an input sound and its element at frequency i and time j, respectively, as obtained by STFT. The loudness of this signal is computed by $v_j = \sum_{i=1}^{I}(y_{ij} + \varepsilon)$, where ε represents a small value to avoid zero-division. The amplitude spectrogram is

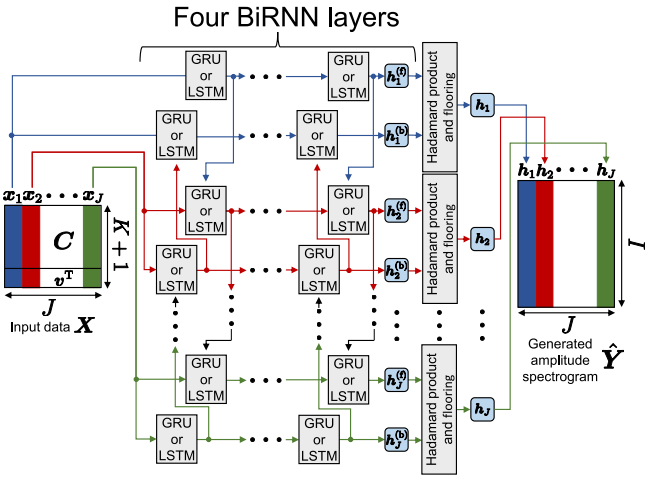

Figure 3: Architecture of BiRNN used as DNN decoder

normalized using the loudness as $\overline{y}_{ij} = y_{ij}/v_j$. Then, the MFCC is computed with the normalized power spectrogram $\boldsymbol{P} \in \mathbb{R}_{\geq 0}^{I \times J}$, whose element is $\overline{y}_{ij}^2$. This MFCC is denoted as $\boldsymbol{C} \in \mathbb{R}^{K \times J}$, where K is the order of MFCC, which is equal to the number of filters in the bandpass filter called the mel filter bank used in the MFCC computation. The input data of the DNN decoder is defined as

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{v}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{(K+1) \times J}, \tag{1}$$

where $\boldsymbol{v} = [v_1, v_2, \cdots, v_J]^{\mathrm{T}} \in \mathbb{R}^J$.

For the training of MLP, we vectorize $\boldsymbol{X}$ and input the vector into the first layer. The label (reference of the DNN prediction) is defined as $\boldsymbol{y} \in \mathbb{R}_{\geq 0}^{IJ}$, a vectorized version of the original amplitude spectrogram $\boldsymbol{Y}$. This MLP consists of three fully connected hidden layers with 1024–512–512 dimensions, and each hidden layer has a rectified linear unit as the activation function.

The network architecture of BiGRU or BiLSTM is illustrated in Fig. 3. The column vectors of $\boldsymbol{X}$, denoted as $\boldsymbol{x}_j \in \mathbb{R}^{K+1}$, are input into the first BiRNN layer. This bidirectional computation is applied four times, acquiring the forward output vectors $\boldsymbol{h}_1^{(\mathrm{f})}, \boldsymbol{h}_2^{(\mathrm{f})}, \cdots, \boldsymbol{h}_J^{(\mathrm{f})} \in \mathbb{R}_{\geq 0}^{I}$ and the backward output vectors $\boldsymbol{h}_J^{(\mathrm{b})}, \boldsymbol{h}_{J-1}^{(\mathrm{b})}, \cdots, \boldsymbol{h}_1^{(\mathrm{b})} \in \mathbb{R}_{\geq 0}^{I}$. The dimension of each vector is increased from $K+1$ to I at the first BiRNN layer and is maintained throughout the remaining layers. Finally, the predicted amplitude spectrogram is composed as

$$\hat{\boldsymbol{Y}} = [\boldsymbol{h}_1\ \boldsymbol{h}_2\ \cdots\ \boldsymbol{h}_J] \in \mathbb{R}_{\geq \varepsilon}^{I \times J}, \tag{2}$$

$$\boldsymbol{h}_j = \max\left(\boldsymbol{h}_j^{(\mathrm{f})} \odot \boldsymbol{h}_j^{(\mathrm{b})}, \varepsilon\right)\ \forall j, \tag{3}$$

where $\odot$ denotes the Hadamard product and $\max(\cdot,\cdot)$ returns the maximum value of inputs in each element.

All the DNNs are trained by minimizing the mean squared error $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2$, where $\hat{\boldsymbol{y}} \in \mathbb{R}_{\geq \varepsilon}^{IJ}$ is a vectorized version of the

Table 1: Experimental conditions

Window and shift lengths in STFT	64/32 ms
Window function in STFT	Hann window
Maximum frequency of mel filter bank	8 kHz
Minimum frequency of mel filter bank	0 kHz
Number of mel-filters (K)	64
Flooring value ε	2.0×10^{-7}

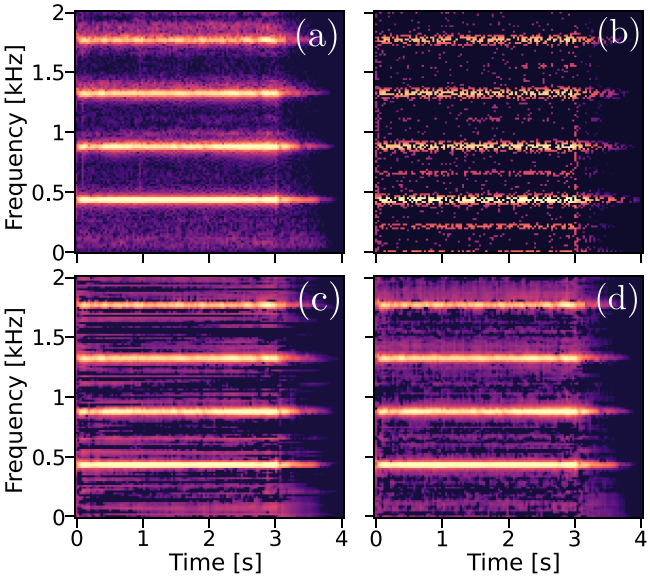

Figure 4: Example of spectrograms for test data (acoustic flute sound): (a) original, (b) predicted by MLP, (c) predicted by BiGRU, and (d) predicted by BiLSTM

predicted amplitude spectrogram $\hat{\boldsymbol{Y}}$ and $\|\cdot\|_2$ is the L_2 norm.

3. Experimental evaluation of DNN decoders

3.1 Dataset and conditions

To evaluate the performance of the DNN decoder, we conducted an experiment on amplitude spectrogram prediction using the neural audio synthesis (Nsynth) dataset [11]. Nsynth is an audio dataset comprising four-second-long signals of various musical instrument sounds and consists of 305,979 signals. These signals were split into 289,205 training, 12,678 validation, and 4,096 test data. The other experimental conditions are shown in Table 1. As evaluation criteria, we used amplitude relative squared error (ARSE) and MFCC relative squared error (MRSE), defined as

$$\mathrm{ARSE} = 10\log_{10}\frac{\sum_{j=1}^{J}\sum_{i=1}^{I}(y_{ij}-\hat{y}_{ij})^2}{\sum_{j=1}^{J}\sum_{i=1}^{I}y_{ij}^2}\ [\mathrm{dB}], \quad (4)$$

$$\mathrm{MRSE} = 10\log_{10}\frac{\sum_{j=1}^{J}\sum_{k=2}^{14}(c_{kj}-\hat{c}_{kj})^2}{\sum_{j=1}^{J}\sum_{k=2}^{14}c_{kj}^2}\ [\mathrm{dB}], \quad (5)$$

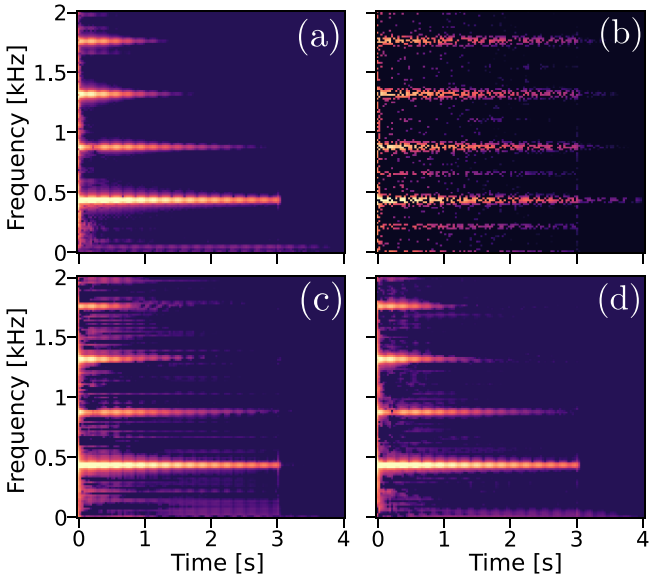

Figure 5: Example of spectrograms for test data (synth. keyboard sound): (a) original, (b) predicted by MLP, (c) predicted by BiGRU, and (d) predicted by BiLSTM

respectively, where $\hat{y}_{ij}$ and c_{kj} are the elements of $\hat{\boldsymbol{Y}}$ and $\boldsymbol{C}$, respectively, and $\hat{c}_{kj}$ is the MFCC calculated from the predicted amplitude spectrogram $\hat{\boldsymbol{Y}}$. Small values of ARSE and MRSE indicate accurate prediction performance. It should be noted that for calculating MRSE, we only used the MFCCs from $k = 2$ to 14, as these dimensions specifically encompass the timbre characteristics.

3.2 Results

Figs. 4, 5, and 6 show examples of original and predicted amplitude spectrograms. These results confirm that the MLP consistently fails to predict the amplitude spectrogram: the predicted results contain numerous spectral holes, resulting in artificial distortions. Conversely, BiGRU and BiLSTM accurately predict the harmonic structures and temporal transitions within the original amplitude spectrogram. This is due to the recurrent architecture in BiGRU and BiLSTM, which effectively captures the time-series structures of the input data. Also, Fig. 7 shows ARSE and MRSE results averaged over all test data for each musical instrument. In both scores, BiLSTM consistently provides the best result for all the instruments. The results for mallets are inferior as compared to other musical instruments. This is likely due to the fact that mallets are typically classified as percussion instruments, and their sounds are characterized by complex harmonic structures and time transitions, as shown in Fig. 6. From these results, we found that BiLSTM provides satisfactory performance in predicting the amplitude spectrogram and, thus, is a preferred choice for the DNN decoder in the proposed system.

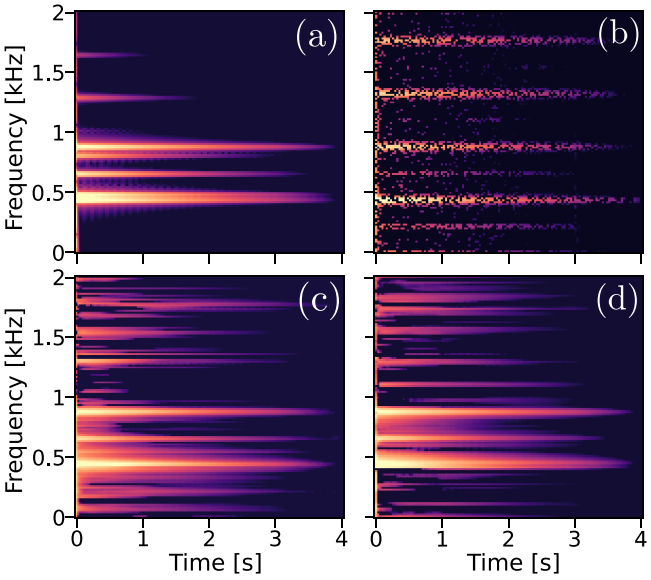

Figure 6: Example of spectrograms for test data (acoustic mallet sound): (a) original, (b) predicted by MLP, (c) predicted by BiGRU, and (d) predicted by BiLSTM

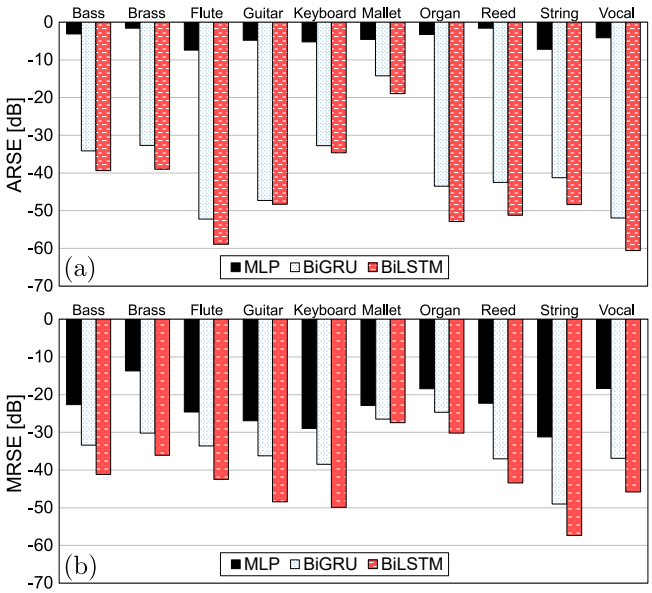

Figure 7: (a) ARSE and (b) MRSE results averaged over test data for each musical instrument

4. Conclusions

We examined the efficacy of predicting amplitude spectrograms from MFCC and loudness using DNNs. Experiments using the Nsynth dataset showed that BiLSTM consistently outperforms MLP and BiGRU models. In future work, we intend to construct the proposed sound generation system (Fig. 1) utilizing the BiLSTM-based DNN decoder.

Acknowledgment

This work was partially supported by JSPS KAKENHI Grant number 22H03652 and Ono Charitable Trust for Acoustics.

References

[1] J. Engel, L. Hantrakul, C. Gu, and A. Roberts, "Differentiable Digital Signal Processing," *in Proc. ICLR*, 2020.

[2] D. P. Kingma, and M. Welling, "Auto-Encoder Variational Bayes", *in Proc. ICLR*, 2014.

[3] Y. J. Luo, K. Agres, and D. Herremans, "Learning disentangled representations of timbre and pitch for musical instrument sounds using Gaussian mixture variational autoencoders," *in Proc. ISMIR*, pp 746–753, 2019.

[4] Y. J. Luo, K. W. Cheuk, T. Nakano, M. Goto, and D. Herremans, "Unsupervised disentanglement of pitch and timbre for isolated musical instrument sounds," *in Proc. ISMIR*, pp 700–707, 2020.

[5] K. Tanaka, R. Nishikimi, Y. Bando, K. Yoshii, and S. Morishima, "Pitch-timbre disentanglement of musical instrument sounds based on VAE-based metric learning," *in Proc. ICASSP*, pp. 111–115, 2021.

[6] F. Zheng, G. Zhang, and Z. Song, "Comparison of different implementations of MFCC," *J. Comput. Sci. Technol.*, vol. 16, pp. 582–589, 2001.

[7] J. W. Kim, J. Salamon, P. Li, and J. P. Bello, "CREPE: A convolutional representation for pitch estimation," *in Proc. ICASSP*, pp. 161–165, 2018.

[8] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," arXiv: 1406.1078, 2014.

[9] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, 2016.

[10] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997.

[11] J. Engel, C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi, "Neural audio synthesis of musical notes with WaveNet autoencoders," *in Proc. ICML*, pp. 1068–1077, 2017.