Improved Architecture for High-resolution Piano Transcription to Efficiently Capture Acoustic Characteristics of Music Signals

Jinyi Mi, Sehun Kim and Tomoki Toda
Nagoya University, Japan
E-mail: {mi.jinyi, kim.sehun}@g.sp.m.is.nagoya-u.ac.jp, tomoki@icts.nagoya-u.ac.jp

***Abstract*—Automatic music transcription (AMT), aiming to convert musical signals into musical notation, is one of the important tasks in music information retrieval. Recently, previous works have applied high-resolution labels, i.e., the continuous onset and offset times of piano notes, as training targets, achieving substantial improvements in transcription performance. However, there still remain some issues to be addressed, e.g., the harmonics of notes are sometimes recognized as false positive notes, and the size of AMT model tends to be larger to improve the transcription performance. To address these issues, we propose an improved high-resolution piano transcription model to well capture specific acoustic characteristics of music signals. First, we employ the Constant-Q Transform as the input representation to better adapt to musical signals. Moreover, we have designed two architectures: the first is based on a convolutional recurrent neural network (CRNN) with dilated convolution, and the second is an encoder-decoder architecture that combines CRNN with a non-autoregressive Transformer decoder. We conduct systematic experiments for our models. Compared to the high-resolution AMT system used as a baseline, our models effectively achieve 1) consistent improvement in note-level metrics, and 2) the significant smaller model size, which shed lights on future work.**

I. Introduction

Automatic music transcription (AMT) has gained considerable research interest in the fields of music signal processing and music information retrieval (MIR) for several decades [1]. The object of AMT is to convert acoustic musical signals into some form of musical notation [2], such as piano rolls, sheet music, and Musical Instrument Digital Interface (MIDI), to improve the time-consuming process of manual music transcription. AMT has been applied in automatic annotation of musical information [3], musical education through automatic instrument tutoring [4], [5], and musicological analysis [6]. In addition, a successful AMT system is also useful for the other tasks in MIR, such as beat tracking [7], chord recognition [8], and performance classification [9].

Piano transcription is a crucial task of AMT, which typically transcribes piano recordings into a series of note events with pitches, onset/offset timings, and velocities. This task is particularly challenging due to its inherent polyphonic nature, i.e., the multiple pitches are usually in the same frame, thereby causing a complex interaction and overlap of harmonics. To address this issue, previous methods, taking into account the piano's acoustic property that the note energy decays after an onset, have mainly focused on adapting models to notes with varying amplitude and harmonics. Inspired by image classification tasks [10], convolutional neural network (CNN)-based methods [11], [12] have been proposed to regress harmonic structures as acoustic representations for audio. On the other hand, recurrent neural network (RNN) methods, such as long short-term memory (LSTM) [13] and gated recurrent unit (GRU) [14], have been applied to capture the medium- and long-range dependencies between notes. Recently, drawing on advancements in speech recognition tasks, convolutional recurrent neural network (CRNN), which combines a CNN acoustic model with an RNN-based sequential model, has become popular for polyphonic piano transcription. Particularly, Kong et al. [15] proposed a high-resolution piano transcription system by regressing precise onset and offset times of notes at arbitrary time resolution, achieving effective performance in piano transcription.

However, there are several limitations of the high-resolution system [15]. First, the harmonics of notes are usually recognized as false positive notes. A potential reason is that the frequency components are calculated with the short-time Fourier transform (STFT), whereas the frequencies that have been chosen to make up the music scale are geometrically spaced, thus yielding components that do not map efficiently to musical frequencies. The second limitation is excessive time and resource consumption due to the large model size. This motivates us to build lightweight architectures for getting around resource constraints and achieve the higher accuracy of transcription.

With this in mind, this work applies an alternative front-end called the Constant-Q Transform (CQT) [16] instead of the STFT to achieve better simulation of the frequencies in music signals. Moreover, we have designed two architectures: the first is based on a CRNN with dilated convolution to well capture a harmonic structure of music signals on CQT feature, and the second is an encoder-decoder architecture that integrates CRNN with a non-autoregressive Transformer (NR-Transformer) decoder. Compared to the high-resolution system, we show that our systems effectively achieve consistent improvement in note-level metrics. Specifically, the proposed

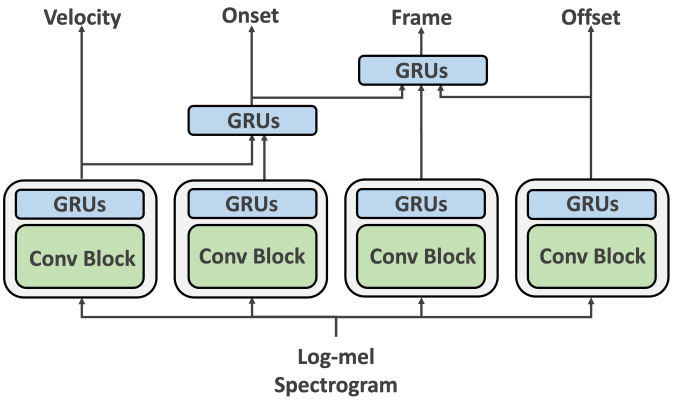


Fig. 1. The high-resolution model.

architectures use significantly fewer parameters of 2.7 million and 0.9 million, respectively, while the high-resolution system uses 20 million parameters. This demonstrates that our systems can achieve the ideal transcription performance without excessive resource consumption.

II. Related work

Since the high-resolution system can be seen as the step stone of our work, we here give a detailed overview of the high-resolution system in this section.

A. Onset and offset times detection

Onsets and offsets represent the beginning and ending of a piano note event, carrying rich information about the piano notes. [17] proposed a dual-objective system for onsets and offsets that conditions the detection of onsets to predict frame-wise outputs. For each piano note, [17] labeled only several consecutive frames of an onset or offset as 1, while other frames was labeled as 0, thereby limiting the transcription resolution. When the precise onset or offset time shifts within a frame, the information about these changes is lost after quantization. Additionally, this labeling method cannot handle cases where the precise onset or offset time is on the boundary between two frames. In light of this, [15] proposed a new labeling method to address the issue that transcription resolutions are limited by the hop size between adjacent frames, by predicting the continuous onset and offset times of piano notes. A new training target was applied to represent the time difference between the center of a frame and its nearest onset or offset times of a note. The process of encoding the time difference into the training targets can be formulated as

$$\begin{cases} g\left(\Delta_i\right) = 1 - \frac{|\Delta_i|}{J\Delta}, |i| \leq J \\ g\left(\Delta_i\right) = 0, |i| > J, \end{cases} \tag{1}$$

where Δ and Δ_i are the frame hop size time and the time difference, respectively. i is the index of a frame, where negative and positive i values indicate the previous and future frame indexes of an onset or offset. J is a hyperparameter that is used to control the sharpness of the targets, i.e., the smaller the J, the sharper the targets $g(\Delta_i)$.

B. High-resolution piano transcription system

Fig. 1 shows the architecture of the high-resolution model that we use as the baseline. The log-mel spectrogram with a shape of $T \times F$ calculated from STFT spectrum is the input feature, where T is the number of frames, and F is the number of mel frequency bins. The frame, onset, offset, and velocity tasks share the same acoustic module. This acoustic module, consisting of a convolution block with several convolutional layers and a bidirectional gated recurrent unit (biGRU) layer, is able to extract spectral and temporal information from the log-mel spectrogram. Then, a fully connected layer is applied to output the results of the acoustic model with a shape of $T \times K$, where K is the number of pitch classes. The prediction outputs of velocities are concatenated with outputs of the onsets from the acoustic model. This concatenated data is then fed into a biGRU layer to calculate the final onset predictions. Similarly, the predicted onsets and offsets are used as conditional information to predict frame-wise outputs in the same way.

III. Proposed method

At the core of our approach is the idea of effectively simulating the frequencies of music signals to improve the performance of the high-resolution system. We have designed two architectures: the first is based on a CRNN with dilated convolution (See Section III-A), while the second combines a CRNN with a NR-Transformer decoder in an encoder-decoder architecture (See Section III-B).

A. Improved CRNN Model for the High-Resolution System

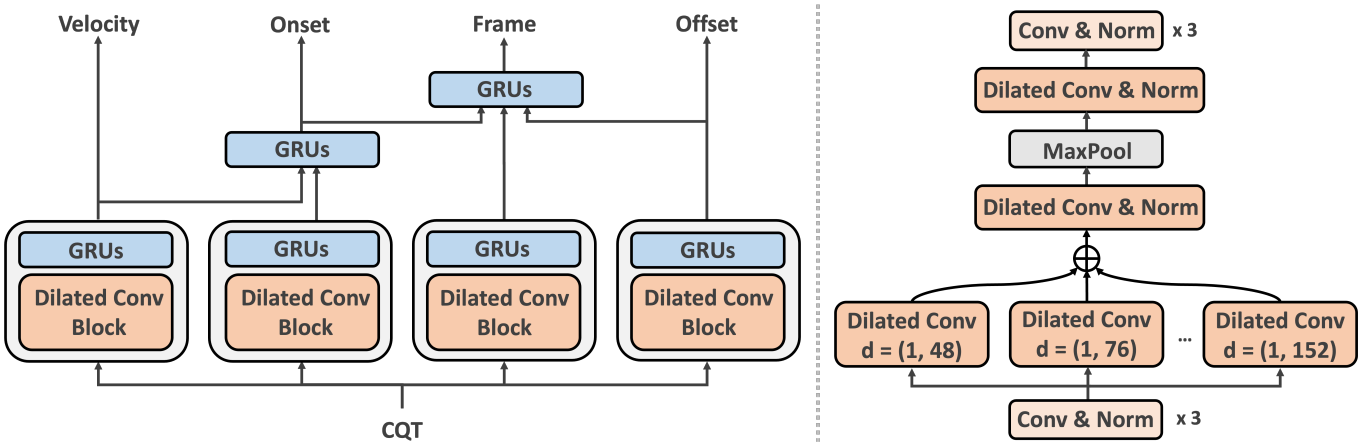
Fig. 2(a) shows the overall architecture of our proposed CRNN model for improved high-resolution system. The CQT spectrogram is applied as input instead of the log-mel spectrogram. Onset, offset, frame and velocity tasks share the same acoustic model that includes a dilated convolution block and a biGRU layer. The output of the velocity serves as conditional information for the onset, while the outputs of both onset and offset condition the frame. We denote this model as HRplus.

1) Inputs and Outputs: To better simulate the frequencies in music signals, we use CQT as the input representation. Unlike the traditional Fourier transform or the STFT which employ a linear frequency scale, the CQT utilizes a logarithmic frequency scale that closely approximates musical frequencies. As described in [18], the distance d_k between the fundamental frequency f_0 and the k-th overtone is given by

$$\begin{aligned} d_k &= \log_{2^{1/Q}}\left(k \cdot f_0\right) - \log_{2^{1/Q}}\left(f_0\right) \\ &= Q \cdot \log_2(k), \end{aligned} \tag{2}$$

where Q is the constant factor of the filter banks that indicates the number of frequency bins per octave.

The outputs of models consist of frame, onset, offset, and velocity events, where onsets and offsets are represented as continuous events, as described in Section II-A. Then, we post-process note-wise events into a set that contains onset time,



(a) Overall structure of HRplus.

(b) Dilated Conv Block.

Fig. 2. Illustration of HRplus model architecture. Norm denotes instance normalization with a ReLu activation, d denotes the dilation rate.

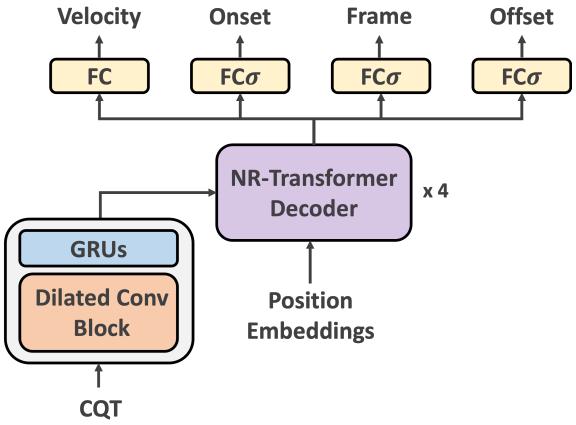


Fig. 3. Illustration of HRplus-hybrid model architecture. FC denotes a fully connected layer, σ denotes a sigmoid function.

offset time, and velocities, using the algorithm of the high-resolution system. This post-process proceeds in the following steps:

Step 1. Note onset detection. If the value of an onset event in a frame exceeds the onset threshold, and this value is a local maximum, this frame is detected to contain an onset. Then, the precise onset time is calculated.

Step 2. Velocity scaling. MIDI files use integers between 0 and 127 to represent the velocity of notes. Since we normalize the dynamic range of velocities from $[0, 127]$ to $[0, 1]$ during model training, the predicted velocities are scaled back to the $[0, 127]$ range if an onset is detected in Step 1.

Step 3. Note offset detection. For the onset detected in Step 1, an offset is detected if the offset prediction output exceeds the offset threshold, or if any frame prediction outputs fall below the frame threshold. Then, the precise offset time is calculated. Additionally, when consecutive onsets of the same pitch are detected, the previous onsets are truncated by adding offsets.

2) Acoustic Model for CQT Input Representation: Considering the challenge of capturing and analyzing multi-scale frequency information scattered in the CQT, we use a dilated convolution block and a biGRU layer as the acoustic model, inspired by HPPNet-sp [19]. The detail of dilated convolution block is shown in Fig. 2(b). The first three convolution layers, each with a kernel size of 7×7 and equipped with instance normalization and ReLU activation, extract local information from the CQT input. Subsequently, eight dilated convolution layers with kernel sizes of 1×3 apply different dilation rates of 48, 76, 96, 111, 124, 135, 144, and 152, respectively, to capture the harmonic series. These dilation rates, calculated by Eq. (2) with $Q = 48$, correspond to the intervals between adjacent harmonics in a harmonic series, i.e., d_2, d_3, ..., d_9 in Eq. (2). The outputs of the eight dilated convolution layers are combined and then fed into a dilated convolution layer with a dilation rate of 48 and a kernel size of 1×3, equipped with instance normalization and ReLU activation. Subsequently, a max-pooling layer with a pooling size of 4 along the frequency axis reduces the frequency bins to match the number of pitch classes. Finally, a dilated convolution layer with a kernel size of 1×3, a dilation rate of 12, instance normalization, and ReLU activation is applied, followed by three convolution layers with kernel sizes of 5×1.

B. Hybrid CRNN-Transformer Encoder-Decoder Model

We have designed an encoder-decoder architecture integrating the acoustic model of HRplus with the NR-Transformer decoder, as shown in Fig. 3. We denote this model as HRplus-hybrid. The motivations for designing this encoder-decoder architecture are derived from two main considerations: 1) the acoustic model of HRplus could serve exclusively as the encoder of HRplus-hybrid, so as to focus on extracting features from the input and transforming them into a intermediate high-dimensional representation; and 2) the strength of Transformer decoder in capturing long-term dependencies would be leveraged, thanks to which these intermediate representations could be effectively decoded to more accurate outputs even if reducing the model size. The decoder employs the NR-

Transformer decoder, which includes four decoder blocks stacked in series. Each NR-Transformer decoder block includes a self-attention module, a cross-attention module, and a feed-forward module. The outputs of the encoder and a trainable positional embedding are used to calculate the cross-attention. Finally, a fully connected layer outputs the predictions for onset, offset, frame, and velocity.

C. Loss functions

We use a similar loss calculation as the high-resolution system for the proposed models, with the total loss L_{total} being the sum of the losses for onset L_{on}, offset L_{off}, frame L_{fr}, and velocity L_{vel} as

$$L_{total} = L_{on} + L_{off} + L_{fr} + L_{vel}. \tag{3}$$

We denote the continuous target and prediction of onset and offset as g_{on}, $\hat{g}_{on}$, g_{off}, and $\hat{g}_{off}$, respectively, where they are regarded as the probability of a binary variable (i.e., from 0 to 1), and the binarized target and prediction of frame as b_{fr} and $\hat{b}_{fr}$ (i.e., 0 or 1 for the target and from 0 to 1 for the prediction). The onset loss L_{on}, offset loss L_{off}, and frame loss L_{fr} are represented as

$$L_{on} = \sum_{t=1}^{T}\sum_{k=1}^{K} L_{bce}(g_{on}(t,k), \hat{g}_{on}(t,k)), \tag{4}$$

$$L_{off} = \sum_{t=1}^{T}\sum_{k=1}^{K} L_{bce}(g_{off}(t,k), \hat{g}_{off}(t,k)), \tag{5}$$

$$L_{fr} = \sum_{t=1}^{T}\sum_{k=1}^{K} L_{bce}(b_{fr}(t,k), \hat{b}_{fr}(t,k)). \tag{6}$$

L_{bce} is a binary cross-entropy loss function defined as

$$L_{bce}(y, \hat{y}) = -y \cdot log(\hat{y}) - (1-y) \cdot log(1-\hat{y}), \tag{7}$$

where y denotes target and $\hat{y}$ denotes prediction. Since we predict the velocity only where the onset is detected, the velocity loss L_{vel} is represented as

$$L_{vel} = \sum_{t=1}^{T}\sum_{k=1}^{K} b_{on}(t,k) \cdot L_{bce}(b_{vel}(t,k), \hat{b}_{vel}(t,k)), \tag{8}$$

where b_{vel} and $\hat{b}_{vel}$ are the binarized target and prediction of velocity, $b_{on} \in \{0,1\}^{T\times K}$ indicates the presence or absence of note onsets.

IV. Experimental Evaluations

A. Datasets

To evaluate the performance of our systems on the piano transcription task, we used the MIDI and Audio Edited for Synchronous Tracks and Organization (MAESTRO) [20] dataset, a large-scale piano dataset containing about 200 hours of paired CD-quality audio recordings and MIDI files from ten years of the International Piano-e-Competition. These audio recordings and MIDI files are aligned with around 3 ms accuracy and sliced into individual musical pieces, which are annotated with composer, title, and year of performance. Virtuoso pianists performed on Yamaha Disklaviers, concert-quality acoustic grand pianos, integrated with a high-precision MIDI capture and playback system. To compare against the baselines, we trained and evaluated on MAESTRO v2 and MAESTRO v3 datasets, respectively. We used the train/validation/test split following by the official configuration of MAESTRO dataset without any extensions or augmentations. The total duration of each split in hours are 161.3/19.4/20.5 in MAESTRO v2 and 159.2/19.4/20.0 in MAESTRO v3, respectively.

B. Experimental Setup

We used PyTorch [21] to implement our systems. The audio recordings were split into 20-second pieces and resampled to 16 kHz so that all the frequencies of the piano could be covered. Then, we down-mixed the audio into a single channel and converted it to a CQT using the nnAudio toolkit [22] with a hop length of 320 points, 48 bins per octave, resulting in a total of 352 frequency bins. For high-resolution label for onset and offset, we set the hyperparameter $J = 5$, i.e., each onset or offset affects the regression values of $2 \times J = 10$ frames.

All the models we proposed were trained with the Adam optimizer [23] with a batch size of 2 and a learning rate of 6×10^{-4}. We used the ReduceLROnPlateau scheduler in PyTorch for learning rate scheduling, employing its default parameters. The best models were determined by the performance on the validation set. At inference, the velocity threshold was set to 0, while all other thresholds were set to 0.4. The outputs were converted to MIDI events as described in Sections III-A1.

C. Baselines

Since the proposed models are optimized based on the high-resolution system, and the acoustic model is inspired by HPPNet-sp, we use both High-resolution [15] and HPPNet-sp [19] as baselines. The results are shown in Table I and Table II. High-resolution applies the high-resolution labels for note onset and offset to construct the piano transcription system. HPPNet-sp uses dilated convolution and frequency-grouped LSTM to model the harmonic structure and pitch-invariance over time in piano transcription.

D. Evaluation Metrics

For evaluation metrics of the piano transcription systems, we used note-level metrics involving the standard precision, recall, and F1 score. These metrics match each predicted note with a ground truth note, considering onset times, pitches, and optional offset times and velocities. We used the mir_eval library [24] for all metric calculations. Following the default configuration of mir_eval, we applied a 50 ms tolerance for note onsets, a 20% offset ratio for note offsets, and a tolerance of 0.1 for velocities.

E. Experimental Results

Table I shows the evaluation results for the MAESTRO v2. For the MAESTRO v2 dataset, HRplus significantly outperforms High-resolution on all metrics. Besides, HRplus uses

TABLE I
TRANSCRIPTION RESULTS EVALUATED ON THE MAESTRO V2 DATASET
(P: PRECISION, R: RECALL, **BOLD**: BEST SCORE)

Model	Params	Onset			Onset & Offset			Onset, Offset & Velocity		
		P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
High-resolution [15]	20M	98.17	95.35	96.72	83.68	81.32	82.47	82.10	79.80	80.92
HRplus	**2.7M**	**98.96**	**95.81**	**97.34**	**86.05**	**83.36**	**84.67**	**84.21**	**81.59**	**82.86**

TABLE II
TRANSCRIPTION RESULTS EVALUATED ON THE MAESTRO V3 DATASET
(P: PRECISION, R: RECALL, **BOLD**: BEST SCORE, UNDERLINE: SECOND BEST SCORE)

Model	Params	Onset			Onset & Offset			Onset, Offset & Velocity		
		P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
High-resolution [reproduced]	20M	98.22	95.26	96.69	83.33	80.86	82.06	81.68	79.28	80.44
HPPNet-sp [19]	<u>1.2M</u>	98.45	**95.95**	<u>97.18</u>	84.88	82.76	83.80	83.29	<u>81.24</u>	82.24
HRplus	2.7M	**99.01**	<u>95.86</u>	**97.39**	**86.14**	**83.46**	**84.76**	**84.31**	**81.69**	**82.96**
HRplus-hybrid	**0.9M**	<u>98.68</u>	94.92	96.73	<u>85.98</u>	<u>82.77</u>	<u>84.32</u>	<u>83.91</u>	80.79	<u>82.30</u>

2.7 million parameters, markedly fewer than the 20 million parameters used by High-resolution. This demonstrates that our model architecture is more lightweight and cost-efficient.

To provide a more comprehensive comparison between our proposed models and the baselines, we train and evaluate High-resolution on the MAESTRO v3 dataset. The results are shown in Table II. From the baseline perspective, we observe that HPPNet-sp outperforms High-resolution in note-level metrics. Meanwhile, HPPNet-sp also has a significantly smaller model size compared to High-resolution. When comparing our proposed models with the baselines, it is observed that HRplus wins in eight out of nine test metrics, especially in the F1 score, which is significantly better than all other systems. Moreover, we find that even with the smallest model size of only 0.9 million parameters, HRplus-hybrid still achieves the second best results in six out of nine test metrics. It outperforms all the baselines in F1 score for Onset & Offset and Onset, Offset & Velocity. Specifically, HRplus-hybrid surpasses the High-resolution system in both F1 score and Precision across all metrics. This demonstrates our proposed models, combining the advantages of High-resolution and HPPNet, are very effective in improving AMT performance as well as reducing resource consumption.

We specifically analyze the model size among different systems. First, since the dilated convolution can significantly reduce the number of parameters, the model size of HRplus is greatly reduced compared to that of High-resolution. Second, in contrast to the approach taken by HRplus which utilizes different GRU+dilated conv block for individual outputs, HRplus-hybrid shares one GRU+dilated conv block across all outputs, thus achieving the smallest model size.

In addition, we note that the results of HR-hybrid are slightly inferior to those of HRplus. A potential reason is that HRplus-hybrid does not utilize conditional information in its prediction outputs. Conversely, the use of conditional information, especially onset information, has significantly enhanced the final transcription performance, as evidenced in the studies of [15], [17], [19].

V. CONCLUSIONS

In this paper, we utilize the CQT as the input representation instead of the STFT to better adapt to musical signals, thereby improving the transcription performance of the high-resolution system. We design two architectures: the first is based on a CRNN with dilated convolution, and the second is an encoder-decoder architecture that integrates CRNN with a NR-Transformer decoder. The experimental results demonstrate that our proposed methods can effectively achieve consistent improvements in note-level metrics without any extensions or augmentations. Specifically, the proposed architectures use only 2.7 million and 0.9 million parameters, respectively, compared with the 20 million used by the high-resolution system. Therefore, our systems can achieve the ideal transcription without excessive resource consumption. In the future, we will extend our models to other instruments and introduce transfer learning methods to enhance transcription performance.

ACKNOWLEDGMENT

This work was partly supported by JST CREST JPMJCR19A3, Japan. In addition, this work was also financially supported by JST SPRING, Grant Number JPMJSP2125. The author would like to take this opportunity to thank the "THERS Make New Standards Program for the Next Generation Researchers."

References

[1] E. Benetos, S. Dixon, Z. Duan, and S. Ewert, "Automatic music transcription: An overview," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 20–30, 2018.

[2] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, "Automatic music transcription: Challenges and future directions," *Journal of Intelligent Information Systems*, vol. 41, pp. 407–434, 2013.

[3] Y.-S. Huang and Y.-H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 1180–1188.

[4] E. Cano, C. Dittmar, J. Abeßer, C. Kehling, and S. Grollmisch, "Music technology and education," *Springer Handbook of Systematic Musicology*, pp. 855–871, 2018.

[5] P. Dunbar-Hall, "Music transcription as pedagogy: Discussion of a cross-disciplinary approach to teacher preparation," in *Celebrating Musical Communities: Proceedings of the 40th Anniversary National Conference, Perth 6th-8th July 2007*, Australian Society for Music Education Nedlands, WA, 2007, pp. 89–93.

[6] A. Klapuri and M. Davy, "Signal processing methods for music transcription," 2007.

[7] R. Vogl, M. Dorfer, G. Widmer, and P. Knees, "Drum transcription via joint beat and drum modeling using convolutional recurrent neural networks.," in *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, 2017, pp. 150–157.

[8] Y. Wu and W. Li, "Automatic audio chord recognition with midi-trained deep feature and blstm-crf sequence decoding model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 355–366, 2018.

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

[10] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[11] R. Kelz, M. Dorfer, F. Korzeniowski, S. Böck, A. Arzt, and G. Widmer, "On the potential of simple framewise approaches to piano transcription," *arXiv preprint arXiv:1612.05153*, 2016.

[12] R. Kelz, S. Böck, and C. Widnaer, "Multitask learning for polyphonic piano transcription, a case study," in *2019 International Workshop on Multilayer Music Representation and Processing (MMRP)*, IEEE, 2019, pp. 85–91.

[13] S. Böck and M. Schedl, "Polyphonic piano note transcription with recurrent neural networks," in *2012 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2012, pp. 121–124.

[14] M. A. Román, A. Pertusa, and J. Calvo-Zaragoza, "An end-to-end framework for audio-to-score music transcription on monophonic excerpts," in *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, 2018, pp. 34–41.

[15] Q. Kong, B. Li, X. Song, Y. Wan, and Y. Wang, "High-resolution piano transcription with pedals by regressing onset and offset times," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3707–3717, 2021.

[16] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[17] C. Hawthorne, E. Elsen, J. Song, *et al.*, "Onsets and frames: Dual-objective piano transcription," in *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, 2018, pp. 50–57.

[18] W. Wei, P. Li, Y. Yu, and W. Li, "Harmof0: Logarithmic scale dilated convolution for pitch estimation," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, 2022, pp. 1–6.

[19] W. Wei, P. Li, Y. Yu, and W. Li, "Hppnet: Modeling the harmonic structure and pitch invariance in piano transcription," in *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, 2022.

[20] C. Hawthorne, A. Stasyuk, A. Roberts, *et al.*, "Enabling factorized piano music modeling and generation with the MAESTRO dataset," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=r1lYRjC9F7.

[21] A. Paszke, S. Gross, F. Massa, *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[22] K. W. Cheuk, H. Anderson, K. Agres, and D. Herremans, "Nnaudio: An on-the-fly gpu audio to spectrogram conversion toolbox using 1d convolutional neural networks," *IEEE Access*, vol. 8, pp. 161 981–162 003, 2020.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[24] C. Raffel, B. McFee, E. J. Humphrey, *et al.*, "Mir_eval: A transparent implementation of common mir metrics.," in *Proceedings of the International Society for Music Information Retrieval (ISMIR)*, vol. 10, 2014, p. 2014.