Reconstructing Human Expressiveness in Piano Performances with a Transformer Network

Jingjing Tang[1], Geraint Wiggins[1 2], and György Fazekas[1] *

[1] Center for Digital Music, Queen Mary University of London
[2] Vrije Universiteit Brussel
jingjing.tang@qmul.ac.uk

Abstract. Capturing intricate and subtle variations in human expressiveness in music performance using computational approaches is challenging. In this paper, we propose a novel approach for reconstructing human expressiveness in piano performance with a multi-layer bi-directional Transformer encoder. To address the needs for large amounts of accurately captured and score-aligned performance data in training neural networks, we use transcribed scores obtained from an existing transcription model to train our model. We integrate pianist identities to control the sampling process and explore the ability of our system to model variations in expressiveness for different pianists. The system is evaluated through statistical analysis of generated expressive performances and a listening test. Overall, the results suggest that our method achieves state-of-the-art in generating human-like piano performances from transcribed scores, while fully and consistently reconstructing human expressiveness poses further challenges. Our codes are released at https://github.com/BetsyTang/RHEPP-Transformer.

Keywords: music generation, expressive music performance, transformer model

1 Introduction

An expressive music performance goes beyond playing the notes in the score correctly. Following annotations in music sheets, performers interpret the music with different degrees of expressive control including articulation and dynamics to express emotions and provide an individual rendition of the music, resulting in different performance styles [6]. A common way of rendering expressive performances with computational models is to meaningfully tune the velocity and timing of notes in the score to reconstruct

* This work is supported by the UKRI Centre for Doctoral Training in Artificial Intelligence and Music. J.Tang is a research student supported jointly by the China Scholarship Council and Queen Mary University of London. G. Wiggins received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen".We would like to thank Lele Liu, Jiawen Huang and the reviewers for their valuable feedback to improve our work.

human expressiveness [2]. Generally modelling human expressiveness requires capturing the differences between scores and human performances in expressive features including tempo, timing, dynamics, and so on. Learning the subtle nuances in expression among individual pianists demands the model to learn much smaller perceivable differences within those expressive features.

In recent years, deep learning (DL) models have shown promising results in music generation and representation learning. In particular, the Transformer architecture has gained popularity due to its ability to capture long-range dependencies and contextual information in sequential data. This capability positions the Transformer as a potential solution for modeling performance actions such as adjusting tempo and loudness, and capturing a performer's structural interpretation of music. However, while many studies have successfully applied Transformer architecture to algorithmic music composition [4, 9, 10, 11] and representation learning for symbolic music [5, 24], few works pay attention to modeling human performance expressiveness independently. In the field of expressive performance rendering (EPR), recent studies have achieved convincing results for the purpose of reconstructing general human expressiveness and controlling style using DL architectures including Recurrent Neural Network [12], Graph Neural Network [13] and conditional Variational Autoencoder [21]. These models require large-scale accurate alignments of well-annotated music scores and performances. However, due to the limited quality and size of the currently available datasets, including the Vienna 4x22 Piano Corpus [8] and ASAP [7], these systems still have difficulty dealing with playing techniques such as pedalling and trills, recovering expressiveness overarching longer passages of music, as well as modeling the performance style of individual players.

In this paper, we propose a novel approach for reconstructing human expressiveness with a multi-layer bi-directional Transformer encoder. Training a Transformer model for this task demands large amounts of accurately recorded and score-aligned performance data, which is not currently readily available. A recently released performance-to-score transcription system [15] and the transcribed expressive piano performance dataset ATEPP [25] allow us to use transcribed scores and performances to train our model. Using transcribed scores in the EPR task can be beneficial when the canonical score is not representative enough. For example, jazz performances rely heavily on improvisation, making it difficult to align canonical scores with performances. Even in classical music, ornaments such as trills may not be explicitly notated in canonical scores, which poses problems for the alignment process. Moreover, the reconstruction of human expressiveness from transcribed scores can support research in musical style transfer, particularly when people aim to change a performance by one pianist into the style of another. Considering this, we investigate the ability of our system to model the expressiveness for individual pianists and evaluate it through statistical analysis of the generated performances and a listening test comparing our model to state-of-the-art expressive performance rendering systems.

The rest of this paper is organized as follows: Section 2 describes the methodology detailing the dataset, the process of feature extraction and the model architecture. Section 3 introduces the experiment setting-ups for training our model. Section 4 presents the results of quantitative analysis and the listening test as well as discussions upon the results, and finally, Section 5 concludes the paper.

2 Methodology

2.1 Problem Definition

Expressive performance rendering (EPR) is commonly defined as *the task of generating human-like performances with music sheets as input*. Most existing work [12, 13, 21] proposes systems using recorded performances and canonical scores to solve the problem. All of these systems require alignment between the canonical scores and performances, which is limited in accuracy given the available datasets and alignment algorithms. With the purpose of reconstructing human expressiveness given a composition, we reformulate the task by relaxing the requirement for using conventional music sheets as input, in order to take advantage of the recent performance-to-score transcription algorithms [15] and large transcribed performance datasets [25]. We will provide more details about the transcription algorithm and the dataset used in this work in Sections 2.2 and 2.3. As shown in Fig. 1, the EPR task, in our definition, is to take the transcribed scores as input and reconstruct human expressiveness by generating expressive performances that are similar to the transcribed human performances.

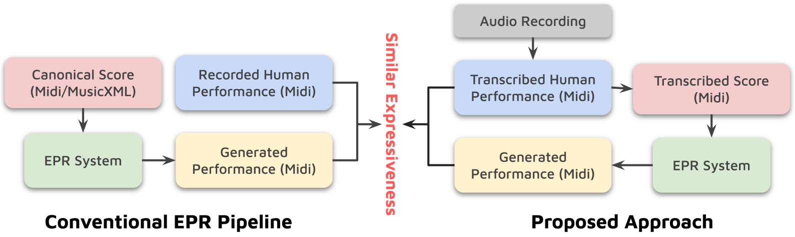
Canonical Score (Midi/MusicXML)
Recorded Human Performance (Midi)
EPR System
Generated Performance (Midi)
Conventional EPR Pipeline
Similar Expressiveness
Audio Recording
Transcribed Human Performance (Midi)
Transcribed Score (Midi)
Generated Performance (Midi)
EPR System
Proposed Approach

Fig. 1. Comparison of the conventional expressive performance rendering (EPR) pipeline with our proposed method

2.2 Dataset

The recently released ATEPP dataset [25] provides high-quality transcribed piano performances by world-renowned pianists. According to a listening test conducted by Zhang et al., the transcribed performance MIDIs reliably retain the expressiveness of performers. The dataset includes multiple performances of the same composition by different pianists, allowing comparison in expressiveness among different performers. However, since the ATEPP dataset has a highly skewed distribution of performers, rather than using the whole dataset, we use a subset [19] that balances the number of performances by six pianists: Alfred Brendel, Claudio Arrau, Daniel Barenboim, Friedrich Gulda, Sviatoslav Richter, and Wilhelm Kempff. Compositions in this subset are mainly composed by Beethoven with only two pieces by Mozart. Each of the compositions corresponds to at least one performance by each pianist. Table 1 presents statistics of the subset in comparison with datasets used by other EPR systems.

Table 1. Comparison of datasets used in different EPR systems. *NN stands for the number of notes. † denotes that the information not provided.

Systems	Performances	Pianists	Compositions	Composer(s)	Total NN*
VirtuosoNet [12]	1052	/†	226	16	3301K
Sketching-Internal [21]	356	/	34	1	/
Sketching-External [21]	116	/	23	10	/
Ours	457	6	36	2	1341K

2.3 Data Processing

Score Transcription Similarly to other EPR systems [12, 13, 21], our method requires note-to-note alignment between the input score MIDI and the output performance MIDI. Despite the convincing alignment results of the state-of-the-art algorithm proposed by Nakamura et al. [18], the algorithm shows difficulty in dealing with repeated sections as well as trills in classical piano music, which causes unexpected loss of information during the alignment process. Instead of using the original or manually edited scores of the compositions, we obtained the transcribed scores of the performances through a performance-to-score transcription algorithm proposed by Liu et al. [15]. The transcribed score midi data can be aligned with the performances at the note level without losing any structural generality in the music [23].

The transcription algorithm performs rhythm quantisation through a convolutional-recurrent neural network and a beat tracking algorithm to remove expressive variations in timing, velocity, and pedalling. While expressiveness regarding velocity and pedalling is certainly erased through the process, how much expressiveness is remained in timing is implicit and will be discussed further in Section 4. A further constraint of this algorithm is its inability to retrieve performance directives like dynamics, phrase markings, and beam directions set by the composer. As a result, we were limited to leveraging only the note-related features the algorithm offered.

Data Augmentation The transcribed scores are first scaled to the same length as the corresponding performances. We then augment the data by changing the tempo for both performances and the scores. For each pair of performance and score midis, the onset time, offset time and duration of each note are multiplied by a ratio $r_i \in [0.75, 1.25]$. In total, we have each pair augmented by multiplying 10 different ratios that are evenly spaced along the interval grid.

Table 2. Vocabulary size of the tokenized note-level features

Features	Pitch	Velocity	Duration	Position	Bar
Size	89	66	4609	1537	518

Feature Encoding Features related to performance expressiveness are extracted and tokenized to reduce the the dimensionality of the input space. Following the tokenisation method, OctupleMIDI, proposed by Zeng et al. [24], we encode the note-level features including pitch, velocity, duration, bar, and position. Table 2 shows the vocabulary size of our tokens for each feature. When using OctupleMIDI, the onset time of a note N_i is represented jointly by its bar number B_i and position number P_i, where $i = 1, 2, \ldots, n$ and n denotes the length of the note sequences. Given that we use a piano music dataset, we consider only pitches with numbers ranging from 21 to 109. The duration of notes is set to be linearly proportional to the token value D_i. All of the midi files have a resolution of 384 ticks per beat, and we default each bar to have 4 beats, resulting in $384 \times 4 = 1536$ different positions per bar. We calculate values of other two note-level performance features which are commonly used for capturing the expressiveness of piano performances [12, 20, 21] based on the tokens:

- *Inter-Onset Interval (IOI)*: the time interval between the onset time (OT) of the note N_i and that of the next note N_{i+1}:

$$IOI_i = \begin{cases} OT_{i+1} - OT_i,\ i = 1, 2, \ldots, n-1 \\ 0,\ i = n \end{cases} \tag{1}$$

  where $OT_i = B_i \times 1536 + P_i,\ i = 1, 2, \ldots, n$

- *Duration Deviation (DD)*: the difference between duration token values of a note in performance midi and score midi

$$DD_i = Dp_i - Ds_i, i = 1, 2, \ldots, n \tag{2}$$

  where Dp is the duration obtained from the performance midi and Ds is that from the score midi.

2.4 Generation with Transformer Encoder

Input and Output Features Input and output features are carefully designed to preserve the score content while allowing changes in the performance control of each note. The input features include pitch, velocity, duration, bar, position, and inter-onset interval from the score midis. As for the output, we infer values of three features including velocity, DD, and IOI in the performance midis. Following Eq. 1 and Eq. 2, we can calculate the predicted token values of duration, position, and bar for each note based on DD and IOI. Combined with the predicted token values for velocity, we can construct a performance MIDI file through detokenization.

Model Architecture Inspired by the MidiBert model proposed by Chou et al. [5], we design a multi-layer bi-directional Transformer encoder with 4 layers of multi-head self-attention where each has 4 heads and a hidden space dimension of 128. The pianist's identity is represented using a one-hot encoding embedding, which is then concatenated to the last hidden state before the final prediction, as shown in Fig 2. As velocity and timing in music are continuous variables, the interval between two token values is informative in representing the distinction of playing a note. Most transformers trained for music generation [9, 4, 11, 10] take different token values as independent

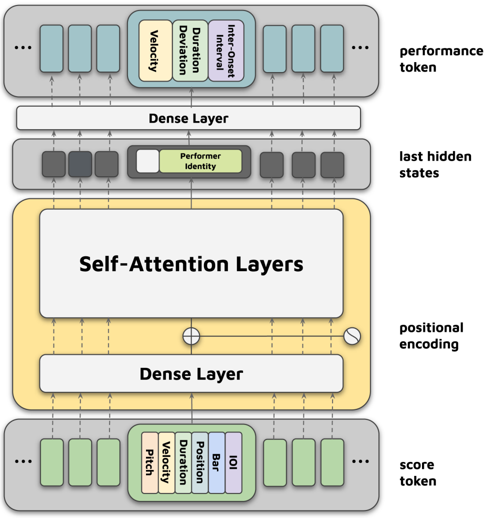

Fig. 2. Model architecture of the Transformer encoder

classes which makes this information implicit to the model. Our system instead uses the tokens without creating embeddings, and predicts the token values for different features through regression. In addition, we add activation functions after the inference layer to clamp the predicted values, ensuring that they fall into the ranges of different features.

Loss Design The losses $\mathcal{L}_v$, $\mathcal{L}_{dd}$, and $\mathcal{L}_{ioi}$ for velocity, DD, and IOI features are calculated respectively, following the loss function defined in Eq. 3 which represents the percentage of how much the predicted values y deviated from the target values $\hat{y}$. Masks are created to exclude loss calculation for padded tokens.

$$\mathcal{L}_{feature} = \sum_{i=0}^{n} l(y_i) m_i, \tag{3}$$

where m_i represents the loss mask for the i-th note and

$$l(y_i) = \begin{cases} \dfrac{|y_i - \hat{y}_i|}{|\hat{y}_i|}, \text{ if } \hat{y}_i \neq 0 \\ \alpha |y - \hat{y}_i|, \text{ if } \hat{y}_i = 0 \end{cases}$$

The parameter α regularizes the loss calculation when the target value is zero and is experimentally set to 0.001. The total loss is calculated by

$$\mathcal{L}_{total} = w_v \mathcal{L}_v + w_{dd} \mathcal{L}_{dd} + w_{ioi} \mathcal{L}_{ioi} \tag{4}$$

where weights are empirically initialized and assigned to each feature loss respectively.

2.5 Evaluation

The system is objectively evaluated through validation losses and statistical distributions of expressive parameters in generations, presented in Section 4.1. Additionally, we evaluate the perceived expressiveness of generated performances through a subjective listening test. As the aim of EPR task is to generate performances with human-like expressiveness [2], we assume that the more similar a model's output is to a human performance, the more effectively expressive it is. We recruit participants who have experience in playing musical instruments and who are engaged with classical music, and ask them to rate the presented samples by evaluating how expressive, natural, and human-like they are. The detailed experiment design and conditions and the results of the listening test are presented in Section 4.2.

3 Experimental Setup

We implement our model based on the PyTorch. We have a 8:1:1 data split in the number of piece and performance, and we cut or pad the token sequences into sequences of 1000 notes before inputting into our transformer. The model is trained with a batch size of 16 sequences for at most 400 epochs, using the Adam optimizer with an initial learning rate of 1e-4 and a weight decay rate of 1e-7. We update the learning rate using the cosine annealing warm restart scheduler [17] since it has been shown to result in faster convergence during training, compared with other learning rate scheduling strategies. If the validation loss does not improve for 30 consecutive epochs, we stop the training process early. The training converges in 2 days on two RTX A5000 GPUs.

Different vocabulary sizes of expressive features shown in Table 2 result in different degrees of complexity when modeling. Consequently, we observed unbalanced decrease in losses and overfitting across learning for different features with constant weights assigned to each feature loss. To balance training and reduce overfitting, we optimize the training process using the GradNorm algorithm proposed by Zhao et al. [3] to dynamically update weights based on gradients calculated at the end of each training epoch.

4 Results

4.1 Quantitative Evaluation

Quantitative methods for evaluating expressive performance rendering systems are limited. One approach [2] is to calculate the loss for each performance feature. Unlike existing approaches [13, 12, 21] where the features are not tokenised, our system computes the losses using the token values. Based on the feature encoding process and the loss design discussed in Section 2, we estimate the average prediction errors in MIDI quantised velocity value and seconds, shown in Table 3.

Although the results are not directly comparable to existing works because of the differences in feature extraction and loss design, they indicate that the transformer model could learn the patterns of expressive variations and reproduce them in the transcribed scores. However, the average errors at the note level in generations are still

Table 3. Loss and average prediction error in MIDI velocity value and seconds for note-level expressive features on the test dataset

Features	Loss	Average Error
Velocity	0.1267	±16.2048
Duration Deviation	0.6280	±0.0473s
Inter-Onset Interval	0.2389	±0.0183s

noticeable to human ears [16], and can affect the perceived expressiveness of the generated music in comparison to human performances.

Since the level of expressiveness regarding timing left in the transcribed scores is implicit as discussed in Section 2, we evaluate the ability of our system to reconstruct the expressiveness for individual pianists through the velocity distributions obtained from kernel density estimation [20, 26].

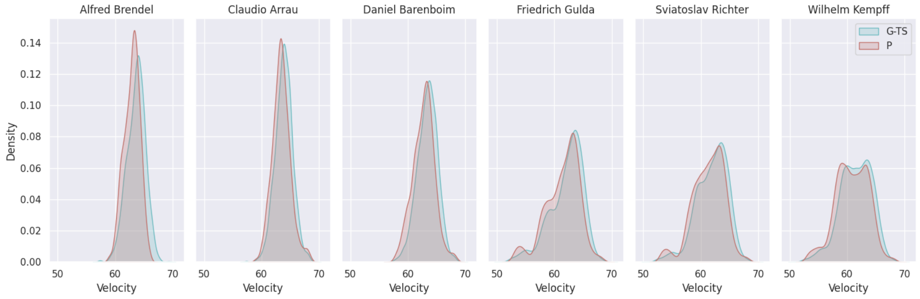

Fig. 3. Velocity distributions for the human performances (P) and the our generations (G-TS) on all pieces in the test set, grouped by different pianists.

As shown in Fig. 3, velocity distributions for each pianists are distinguishable, indicating different performing styles. However, performance recording environments may have impact on the transcribed velocity values [14] and contribute to differences of the distributions. The distributions of the generations based on transcribed scores (G-TS) and those of the human performances (P) have a high degree of overlap, providing evidence of learning individual expressiveness through the training.

4.2 Subjective Evaluation

A listening test was performed to evaluate the perceived expressiveness of our model's output. We recruited 19 people who had some level of music training through email. All participants have learned a musical instrument, while over half of our participants had been engaged with classical music for over 5 years. The participants completed the study anonymously.

The stimuli consisted of four 20s classical piano excerpts detailed in Table 4. For each excerpt, the human performance (**P**) was provided as a reference to be compared with four MIDI renderings: the generation based on the transcribed score (**G-TS**), the generation by the state-of-the-art VirtuosoNet [12] using the canonical score (**V**), a direct rendering of the transcribed score (**TS**), and finally the canonical score (**S**) without expression. The human performances were transcribed piano performance MIDIs from the ATEPP dataset [25] and were included as one of the stimuli as well. All the MIDIs were synthesised into audio recordings through GarageBand to ensure consistency in the listening experience. For each piano excerpt, six recordings, the reference plus 5 stimuli, were presented in the test [3].

Participants were asked to listen to five stimuli, and rate the degree of expressiveness for them on a 100-point scale by comparing each of them with the reference human performance. During the test, we explicitly ask participants to rate based on the expressive differences among the stimulus with more focus on the performance features such as the dynamics and tempo changes rather than the compositional content. We encouraged them to use the full scale, rating the best sample higher than 80 and the worst lower than 20. We adopt the MUSHRA framework [22] to conduct the test using the Go Listen platform [1].

Table 4. Compositions used for the listening test

Annotation	Composer	Composition
Piece **A**	Beethoven	*Piano Sonata No. 19 in G Minor, Op. 49 No. 1: II. Rondo (Allegro)*
Piece **B**	Beethoven	*Piano Sonata No. 7 in D Major, Op. 10 No. 3: III. Menuetto (Allegro)*
Piece **C**	Haydn	*Piano Sonata in C Major, Hob. XVI:48: II. Rondo (Presto)*
Piece **D**	Bach	*French Suite No. 5 in G, BWV 816: 7. Gigue*

In total, 380 ratings from the 19 listeners were collected. We filtered out raters who could not identify the difference in expressiveness between the anchor (**S**) and the reference (**P**). Fig 4 shows the mean opinion scores (MOS) and the results of Wilcoxon signed rank test for the differences between: (a) **TS** versus **S**, (b) **G-TS** versus **V**, (c) **P** versus **G-TS**, (d) **P** versus **V**, (e) **G-TS** versus **TS**.

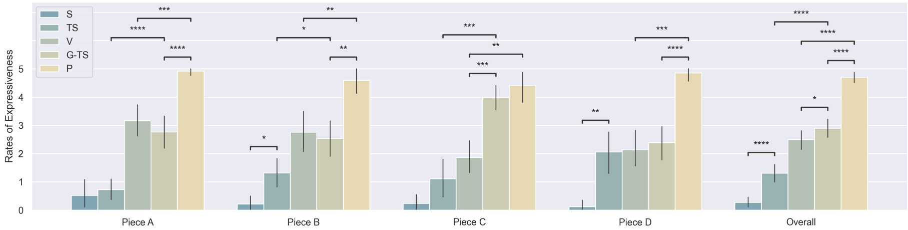

Fig. 4. Results of listening test. The mean opinion scores (converted to a 5-point scale) and 95% confidence intervals are presented for each test piece and the overall results. Wilcoxon signed-rank test are performed to test the significance of the differences. * ($0.01 < p < 0.05$), ** ($0.001 < p < 0.01$), *** ($0.0001 < p < 0.001$), **** ($p < 0.0001$)

[3] Listening samples are provided at `https://drive.google.com/drive/folders/1nfaZ23vr8xZHlyhTAAppK2hl-aHQPigP?usp=sharing`

According to the results, human performances (**P**) are significantly different from generations of our model (**G-TS**) and VirtuosoNet (**V**) in most situations. The outputs of our model (**G-TS**) are overall preferred over the performances produced by VirtuosoNet (**V**) significantly ($0.01 < p < 0.05$), receiving trivially lower (not significant) ratings for piece **A** and **B** but higher (significant for **C** and not significant for **D**) ratings for the compositions that never appear in the training dataset. Comparing with canonical scores (**S**), transcribed scores (**TS**) get significantly higher ratings from listeners. Ratings of the generations by our system (**G-TS**) are significantly higher than those of the direct audio rendering of transcribed scores (**TS**) for most pieces except **D**.

These results suggest that our system achieves the state-of-the-art and even outperforms the VirtuosoNet [12] in some cases, although neither of the systems can consistently generate the same level of expressiveness as human performances. On the other hand, while the transcribed scores (**TS**) could have more expressiveness than the canonical scores (**S**), the generations from the transcribed scores (**G-TS**) are perceptually more expressive than the transcribed scores (**TS**) in most cases, indicating the success of reconstructing human expressiveness. The success has also been proven by the overall difference ($0.01 < p < 0.05$) in MOS between our generations (**G-TS**) and generations from the VirtuosoNet (**V**).

4.3 Case Study: Comparison in Dynamics and Duration

Building on the promising results of our system in the listening test of Piece **C**, we conducted a more detailed analysis to compare the expressive variations in dynamics and duration among human performances, system-generated performances, and scores. Specifically, in Fig. 5, we present the fluctuations in velocity and duration across the note sequences. Compared with the VirtuosoNet generation (**V**), the generation of our system (**G-TS**) could capture both short-term and long-term velocity variations better.

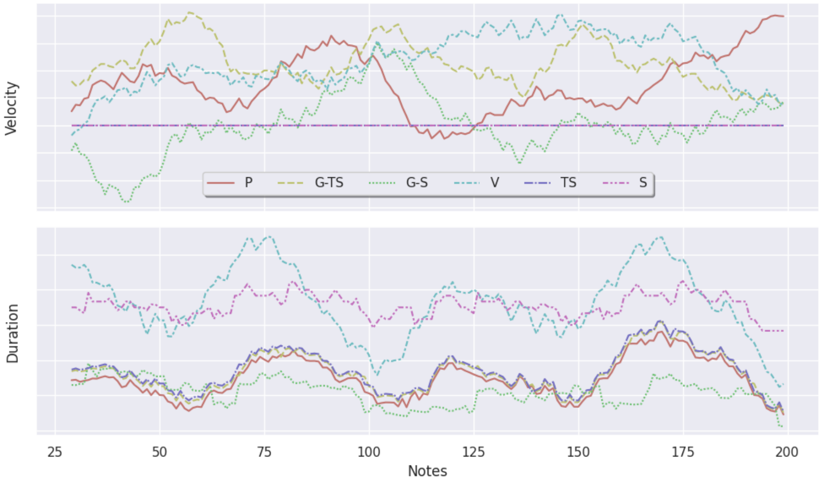

Fig. 5. Standardized and smoothed velocity and duration changes across note sequences from *Piano Sonata in C Major, Hob. XVI:48: II. Rondo (Presto)* for enhanced trend comparison. G-S represents the generation of our system based on the canonical scores.

Even when inputting the unseen canonical score, the generation of our system (**G-S**) outperforms the other model in terms of reconstructing velocity variations. Meanwhile, the strong similarity between duration changes in the human performance (**P**) and transcribed score (**TS**) suggest that the transcription algorithm [15] alters the timing information of the notes cautiously with only limited modification of the duration. Therefore, the reconstruction of the expressive variations in timing through our system could be restricted. The limitation is also demonstrated by the duration changes of our system's generation based on the canonical score (**G-S**).

5 Conclusion

This paper presents a novel method for reconstructing human expressiveness in classical piano performances. Our expressive performance rendering system consists of a Transformer encoder trained on transcribed scores and performances. The quantitative evaluation and listening test show that the proposed method succeed in generating human-like expressive variations, especially for dynamics. Moreover, our method could be used for modeling the differences in expressiveness among individual pianists.

In future work, we will train our system with a mixture of the canonical scores and transcribed scores to create a more robust system. We will further improve the capacity of our system on modeling individual performance styles possibly through contrastive learning. In addition, we will consider a separate system to model pedalling techniques in performances or try to integrate the pedalling information into the current feature encoding.

References

1. Barry, D., Zhang, Q., Sun, P.W., and Hines, A.: Go Listen: An End-to-End Online Listening Test Platform. Journal of Open Research Software (2021)
2. Cancino-Chacón, C.E., Grachten, M., Goebl, W., and Widmer, G.: Computational models of expressive music performance: A comprehensive and critical review. Frontiers in Digital Humanities 5, 25 (2018)
3. Chen, Z., Badrinarayanan, V., Lee, C.-Y., and Rabinovich, A.: GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. In: Dy, J., and Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, pp. 794–803. PMLR (2018)
4. Choi, K., Hawthorne, C., Simon, I., Dinculescu, M., and Engel, J.: Encoding Musical Style with Transformer Autoencoders. In: III, H.D., and Singh, A. (eds.) Proceedings of the 37th International Conference on Machine Learning. Proceedings of Machine Learning Research, pp. 1899–1908. PMLR (2020)
5. Chou, Y.-H., Chen, I., Chang, C.-J., Ching, J., Yang, Y.-H., *et al.*: MidiBERT-piano: Large-scale pre-training for symbolic music understanding. arXiv preprint arXiv:2107.05223 (2021)
6. Dai, S., Zhang, Z., and Xia, G.G.: Music Style Transfer: A Position Paper. arXiv:1803.06841 [cs, eess] (2018)
7. Foscarin, F., Mcleod, A., Rigaux, P., Jacquemard, F., and Sakai, M.: ASAP: a dataset of aligned scores and performances for piano transcription. In: Proceedings of the 21st International Society for Music Information Retrieval Conference, pp. 534–541 (2020)
8. Goebl, W.: Melody lead in piano performance: Expressive device or artifact? The Journal of the Acoustical Society of America 110(1), 563–572 (2001)

9. Hsiao, W.-Y., Liu, J.-Y., Yeh, Y.-C., and Yang, Y.-H.: Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 178–186 (2021)
10. Huang, C.-Z.A., Vaswani, A., Uszkoreit, J., Simon, I., Hawthorne, C., Shazeer, N., Dai, A.M., Hoffman, M.D., Dinculescu, M., and Eck, D.: Music Transformer: Generating Music with Long-Term Structure. In: International Conference on Learning Representations (2018)
11. Huang, Y.-S., and Yang, Y.-H.: Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 1180–1188 (2020)
12. Jeong, D., Kwon, T., Kim, Y., Lee, K., and Nam, J.: VirtuosoNet: A Hierarchical RNN-based System for Modeling Expressive Piano Performance. In: Proceedings of the 20th International Society for Music Information Retrieval Conference (2019)
13. Jeong, D., Kwon, T., Kim, Y., and Nam, J.: Graph neural network for music score data and modeling expressive piano performance. In: International Conference on Machine Learning, pp. 3060–3070 (2019)
14. Kong, Q., Li, B., Song, X., Wan, Y., and Wang, Y.: High-resolution piano transcription with pedals by regressing onset and offset times. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29, 3707–3717 (2021)
15. Liu, L., Kong, Q., Morfi, V., Benetos, E., *et al.*: Performance MIDI-to-score conversion by neural beat tracking. In: Proceedings of the 23rd International Society for Music Information Retrieval Conference (2022)
16. London, J.: Hearing in time: Psychological aspects of musical meter. Oxford University Press (2012)
17. Loshchilov, I., and Hutter, F.: SGDR: Stochastic Gradient Descent with Warm Restarts. In: International Conference on Learning Representations (2017)
18. Nakamura, E., Yoshii, K., and Katayose, H.: Performance Error Detection and Post-Processing for Fast and Accurate Symbolic Music Alignment. In: Proceedings of the 18th International Society for Music Information Retrieval Conference (2017)
19. Rafee, S., Fazekas, G., and Wiggins, G.: HIPI: A Hierarchical Performer Identification Model Based on Symbolic Representation of Music. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (2023)
20. Rafee, S., Fazekas, G., and Wiggins, G.: Performer identification from symbolic representation of music using statistical models. In: Proceedings of the International Computer Music Conference 2021, pp. 178–184 (2021)
21. Rhyu, S., Kim, S., and Lee, K.: Sketching the Expression: Flexible Rendering of Expressive Piano Performance with Self-Supervised Learning. In: International Society for Music Information Retrieval Conference, pp. 178–185 (2022)
22. Series, B.: Method for the subjective assessment of intermediate quality level of audio systems. International Telecommunication Union Radiocommunication Assembly (2014)
23. Wiggins, G.A., Miranda, E., Smaill, A., and Harris, M.: A framework for the evaluation of music representation systems. Computer Music Journal 17(3), 31–42 (1993)
24. Zeng, M., Tan, X., Wang, R., Ju, Z., Qin, T., and Liu, T.-Y.: MusicBERT: Symbolic Music Understanding with Large-Scale Pre-Training. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 791–800, Online (2021)
25. Zhang, H., Tang, J., Rafee, S.R., Dixon, S., Fazekas, G., and Wiggins, G.A.: ATEPP: A Dataset of Automatically Transcribed Expressive Piano Performance. In: International Society for Music Information Retrieval Conference, pp. 446–453 (2022)
26. Zhao, Y., Wang, C., Fazekas, G., Benetos, E., and Sandler, M.: Violinist identification based on vibrato features. In: 2021 29th European Signal Processing Conference (EUSIPCO), pp. 381–385 (2021)