Real-Time Co-Creation of Expressive Music Performances Using Speech and Gestures

Ilya Borovik
Skolkovo Institute of Science and Technology
Moscow, Russia
ilya.borovik@skoltech.ru

Vladimir Viro
Peachnote GmbH
Munich, Germany
vladimir@peachnote.de

ABSTRACT

We present a system for interactive co-creation of expressive performances of notated music using speech and gestures. The system provides real-time or near-real-time dialog-based control of performance rendering and interaction in multiple modalities. It is accessible to people regardless of their musical background via smartphones. The system is trained using sheet music and associated performances, in particular using notated performance directions and user-system interaction data to ground performance directions in performances. Users can listen to an autonomously generated performance or actively engage in the performance process. A speech- and gesture-based feedback loop and online learning from past user interactions improve the accuracy of the performance rendering control. There are two important assumptions behind our approach: a) that many people can express nuanced aspects of expressive performance using natural human expressive faculties, such as speech, voice, and gesture, and b) that by doing so and hearing the music follow their direction with low latency, they can enjoy playing the music that would otherwise be inaccessible to them. The ultimate goal of this work is to enable fulfilling and accessible music making experiences for a large number of people who are not currently musically active.

Author Keywords

Expressive music performance; Human-computer interaction; Mobile interface; Deep learning

CCS Concepts

•**Applied computing** → **Sound and music computing;**
•**Human-centered computing** → *Interaction paradigms;*
•**Computing methodologies** → *Neural networks;*

1. INTRODUCTION

Artificial intelligence and machine learning are bringing new perspectives to the process of creating and performing music [10, 20, 15]. In the traditional music interpretation and performance paradigm, for example in classical music, the musician interprets a score and translates the intended expression into the control of the musical instrument, which then produces the sound that conveys affect and emotion to the listener [24, 28]. However, effective control of musical instruments often requires significant expertise, instrument training, and physical ability.

With this work, we aim to alleviate some of these extra-musical requirements by tapping directly into natural human faculties of affective communication, such as speech, voice, and (facial) gestures. We connect them directly to the composed music, which may be seen as a carrier frequency that is being modulated by these naturally expressed affects to transmit them to the audience.

Recent advances in deep learning have provided opportunities for human-in-the-loop systems to enable interactive music creation [15, 8]. In this context, we propose to use speech and gestures to control music performance. Inspired by recent advances in multimodal representation learning [3, 25, 26], we connect user expression data in multiple modalities with music performance features and offer real-time or near-real-time interaction with the music performance. This allows people to intuitively express their creative ideas and play with the music, exploring the musical works and enjoying spontaneous affective expression through music.

Inspired by the practice of music conducting, in which musicians translate the score and the conductor's gestures, facial expressions, and speech direction during rehearsals into music performance [18, 9], we focus on performance rendering for previously notated music. One useful feature of notated music is the presence of performance direction markings in scores, such as *cresc.*, *lento*, note accents, etc., which composers use to communicate certain aspects of intended articulation to the musicians. Using existing musical performances, we ground these labels in musical performance practices and add them to the vocabulary of our system, which can be used to control performances. We use transformers [30] – state-of-the-art deep neural network architectures in sequence modeling – to advance research in expressive music performance rendering [6].

This paper presents the ongoing development of a deep and active learning based system for interactive co-creation of expressive music performances that provides:

1. real-time interactive music performance rendering;
2. human expression (facial expressions and speech) as performance control modalities;
3. accessibility to people without professional musical training and background;
4. frugal design and accessibility through inexpensive smartphone devices with camera and voice recorder;

NIME'23, 31 May–3 June, 2023, Mexico City, Mexico.

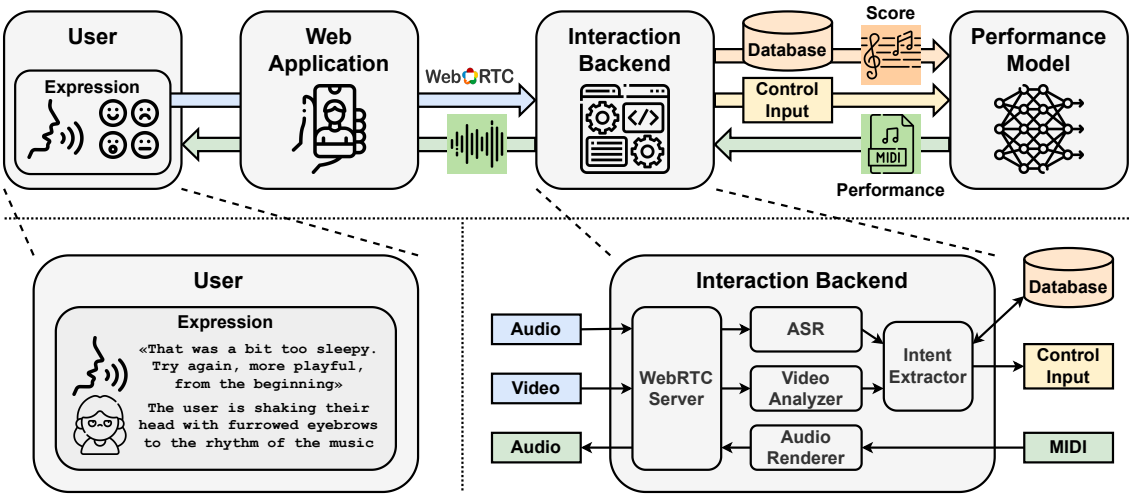

Figure 1: Illustration of an interactive system for real-time co-creation of expressive music performances using speech and facial expressions. The user interacts with the music performance model through a mobile application and interaction backend. The rendered music performance is played back to the user in the mobile application.

Our goal is to provide a new kind of fulfilling, engaging, and accessible music making experience, allowing people to perform great musical works using natural human expression. Our main contributions:

1. We develop a method to interactively control music performance rendering in real-time using speech and facial expressions;
2. We modify transformer models for controllable expressive music performance rendering;
3. We implement a mobile web-interface for interactive music performance co-creation with low hardware requirements.

In the following sections, we present some related work and how our work builds on and differs from it, describe the system architecture, the music performance rendering model, and the application design. We conclude with a discussion of current limitations and future work.

2. RELATED WORK

2.1 Music Generation

Music generation with deep learning [15] is dominated by transformers for learning long-term sequential musical patterns [7, 33, 34] and variational autoencoders for unsupervised style encoding and control [29, 4, 33, 31]. The models offer offline global control of performance style [4, 7] or fine-grained manipulation of performance parameters [33, 31]. Recently, there has been a trend towards description-to-music [31] and text-to-music [1] generation systems, which offer a human intuitive way to musical expression. We adapt the advances in music generation to expressive performance rendering and focus on real-time interactive control.

2.2 Expressive Music Performance

Expressive music performance models render performances for written scores [20, 6]. You can create performances using rules [32, 14] or machine learning models [5, 22, 17, 27]. KTH model [14] uses explicitly learned programmed rules to render and control performance. Basis Mixer [5] maps score features to expressive performance parameters through a set of learned basis functions. VirtuosoNet [17], Maezawa et al. [22], and Rhyu et al. [27] use variational autoencoders for performance style encoding and control, and recurrent neural networks for expressive performance rendering. Our performance model follows a similar design, but uses a transformer architecture [30] to improve long-term music dependency modeling and maps learned style spaces to human expressive control inputs.

2.3 Interactive Music Performance

Interactive music performance systems introduce novel instruments for musical expression [12, 23, 11] and offer control over a generative model through an interface [16, 2, 21, 35]. Wekinator [12] is a computer application and intuitive meta-instrument that learns a mapping between camera-scanned sample inputs, such as gestures or facial expressions, and specific performance actions. CoCoCo [21] offers multi-example sampling with revision and AI steering tools to control the diversity and high-level directions of a generative model. COSMIC [35] provides a novel way to create music through a textual dialog system for coordinating the generation process. We follow the Wekinator approach, but apply it in the domain of expressive performance rendering with a fixed score. We aim for maximum accessibility through a multi-modal mobile web-interface.

3. SYSTEM OVERVIEW

Figure 1 illustrates the architecture of the system and its main components:

1. Music Performance Model;
2. Interaction Backend;
3. Mobile Web Application.

The Music Performance Model enables the controllable creation of expressive performances for written music. Its goal is to mimic human-like musical expressiveness and adequately implement user's expressed wishes by learning from examples of human performance and feedback. By automatically performing the written notes, the model removes the need for a user to play a musical instrument in order to perform a piece of music.

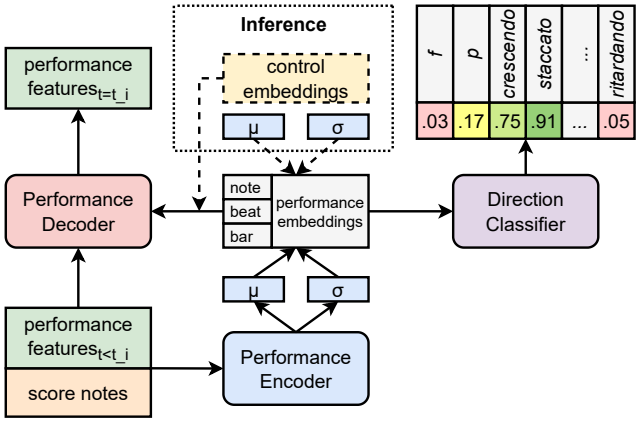

Figure 2: Music Performance Model. The performance encoder compute style performance embeddings on note, beat, and bar levels. The performance renderer decoder outputs performance parameters given note features, past performance, and encoded style representations. The direction classifier associates the performance context with a set of performance direction labels.

The Mobile Application and Interaction Backend connect users to the computational performance model and allow interactive manipulation of the performance. By offering real-time performance rendering, we provide users with online interaction and immediate response. The speech and gestures allow users to express creative ideas using intuitive concepts such as text and emotion.

The following sections describe the technical details behind the performance rendering model, mobile application and interaction backend. Examples of implemented user interactions are listed in Section 5.2.

4. MUSIC PERFORMANCE MODEL

This section describes the computational music performance rendering model shown in Figure 2. It begins with a description of the score and performance features used by the model. Then, we present the expressive performance renderer. Finally, we describe the control modalities and the technical background behind them.

4.1 Score and Performance Data

The performance rendering model requires score and performance data for training. At this point, we restrict ourselves to working only with solo piano music. We preprocess the ASAP dataset of matched piano scores and performances [13]. We compute note-level alignments and filter out performances with less than 80% of matched notes. For simplicity, we correct performances to have a full note-to-note mapping to the score. We remove extra performed notes and interpolate missing notes using the local performance tempo and dynamics.

The score features used to train the models are: note value, bar, position in bar, and performance direction markings (dynamics, tempo, and articulation). The expressive performance parameters predicted by the models are: local performance tempo, note timing, duration, and dynamics.

4.2 Performance Renderer

The Performance Renderer is a deep learning model trained on musical scores and example music performances. It combines transformers [30] for sequential data modeling and variational autoencoders [19] for encoding performance style. The model consists of performance encoder and decoder, shown on the left in Figure 2.

The Performance Encoder computes performance style representations at the note, beat, and bar levels. The Transformer model takes a sequence of score and performance features as input and outputs an embedding for each note. The embeddings are averaged over bars and beats and passed through a linear layer to compute latent bar-, beat-, and note-level performance style embeddings, optimized using a variational evidence lower bound.

The Performance Decoder works with the score features (notes to play), the previous performance context (performance history), and the combined multi-level performance style embeddings computed by the Performance Encoder (style input). The decoder is a decoder-like transformer model with causal attention masking that prevents the model from looking into the future. The outputs are the performance parameters of the next played notes: local onset tempo, note onset deviation, duration, and dynamics. The model is optimized by maximizing the likelihood of the performance parameters. To avoid overfitting to low-level performance embeddings, we randomly drop half of the bar, beat, and note embeddings during model training.

During inference, the randomly sampled and modified performance embeddings can be used to generate and control music performances. Since the embedding space is optimized with the decoder performance generation objective, the latent space encodes features relevant to performance reconstruction. The model supports real-time CPU inference for use in interactive applications.

The model can be fine-tuned on the user-model interaction data. Currently, the active learning framework includes offline periodic fine-tuning of the model on feedback scores. Given a set of performance-feedback score pairs, the performance decoder is optimized with an additional loss function that maximizes the positive feedback per input performance sequence. The model can be fine-tuned within several minutes. In the future, we plan to implement online model fine-tuning.

4.3 Performance Direction Classifier

We associate performance embeddings with musical score directions to provide an intuitive interpretation of the learned control space. We train a Direction Classifier that, given a local context of bar, beat and note embeddings, classifies it into performance direction classes:

- dynamic: degrees of *piano* and *forte*;
- dynamic changes: *crescendo* and *diminuendo*;
- tempo: *adagio*, *largo*, *presto*, etc.;
- tempo changes: *accelerando*, *ritardando*, *a tempo*, etc.;
- articulations: *legato*, *staccato*, *fermata*, etc.

The classifier predicts the likelihood of a direction being performed in a given performance context. Differences between embeddings with high and low likelihoods provide a direction for moving the generation toward a specified performance marking. We can then map these quantified per-direction embedding differences to natural language commands such as *"play more piano here"* or *"switch to largo"* to control performance rendering. This interaction is embedded into the performance control interface.

5. APPLICATION

This section presents a mobile web application[1] for the real-time co-creation of expressive music performances. Our application uses two primary intuitive interaction modalities:

1. **Speech**: the system analyzes the audio stream and recognizes speech-specific phrases. The speech transcription text embeddings are mapped to performance direction classes and corresponding performance control embeddings as described in Section 4.3.

2. **Gestures**: the system processes the video stream and extracts facial expressions. The expression embeddings are mapped to performance direction classes and predefined user-system actions.

These interaction modalities can be combined to create a highly expressive and dynamic musical performance. The following sections go into the implementation details for the backend and frontend of the application.

5.1 Backend

The backend comprises multiple micro-services responsible for different tasks, such as handling the client connection, audio transcription, video analysis, performance rendering, audio rendering, etc. The services are implemented in different languages (Python, C++ and Go) and can be restarted, updated and rolled back independently. They all communicate via a messaging bus. A database stores scores, past performances, user feedback, and performance directions, which we use to optimize the performance model.

The JavaScript client connects to a multi-user WebRTC backend and establishes a bi-directional data channel and audio stream, as well as a video stream from the client. The audio and video streams are analyzed in real-time. Audio is transcribed to text using Whisper [26], which is forwarded to GPT-3 [3] for intent extraction. Intent extraction works with input in multiple languages.

Currently, the system is sequencing and rendering MIDI performances to audio. In the future, we plan to generate audio directly. The MIDI sequencer gets its cues from the gesture and intent recognition services and renders MIDI performances live. The MIDI stream is sent to the audio rendering node. Its audio output is sent back to the WebRTC server process that handles the client connection, and from there the music audio is sent back to the web interface and the user.

The system latency for video-induced performance control is on the order of 0.75 seconds. The behavior seems similar to that of an attentive chamber music partner.

The existing limitations are: small database, limited interaction modalities, rare issues with the quality of rendered performances, support for only one instrument, piano, and no control over acoustic sound properties.

5.2 Mobile Web-interface

The web-interface greets the user and asks them to turn on their camera and microphone on to begin interactive communication with the music performance model. Once the permissions are granted and the WebRTC connection is established, the user sees the camera image in the top half of the screen, while the interaction button appears in the middle of the bottom half. The purpose of the button is to let the user know that the system should pay attention to their input, audio, or video. At the start, the backend selects a random musical composition from the database of musical scores and starts rendering an arbitrary performance for this written music. The user can press the button and ask the system to do any of the following within a single phrase:

1. select a composition
   - *"Let's play Chopin's Mazurka in D major"*

2. pause or stop the performance
   - *"Please stop"*

3. navigate to a different place in the score
   - *"Let's play again from the beginning"*

4. provide feedback on the current performance
   - *"That was still a bit too slow and too much staccato"*

5. ask the system to play in a particular way
   - *"Could you play this like a mother singing a lullaby to her child?"*

6. show the system non-verbally how to play using facial gestures, for example making a blissful expression.

The walkie-talkie button relieves us of the need to continuously evaluate user input and judge whether it is intentional with respect to the performance direction, or accidental (when the user does not intend to direct the performance, but still moves or says something). While the button is pressed, the system continuously evaluates the video input and applies the analysis results to the performance. The audio input is evaluated only after the button is released. However, if a voice activity detector (VAD) detects speech, we immediately reduce the performance volume for the next few seconds in order to not play over the user talking, which tells the user that we are listening and makes it easier to understand the speech.

The information that we are looking for, such as navigation directions, feedback on past performance, and directions for future performance, is extracted from the transcribed speech using GPT-3. This allows us to successfully process free-form speech in multiple languages and offers great flexibility during development, at a cost in reliability and latency that we are currently willing to accept.

The interaction data and feedback are saved in the database to tune the music performance model in subsequent iterations. Specifically, the backend stores the compressed video frame representations, the verbal commands and their embeddings, and the rendered performances. These features are then used in order to fine-tune the performance rendering model according to the desired input control.

6. FUTURE WORK

Our long-term technical goal is to incorporate all natural human modes of expression that are used in musical contexts (conducting, teaching, playing together, etc.). The most obvious is vocalization, pitched or unpitched, which allows one to "show" the system how to play. It can be used to modulate performance tempo, dynamics and articulations. The other mode is full-body gestures. It is unclear whether this mode is practical with the current smartphone-based setup, but if the camera can be positioned farther away from the user, it may enhance possible interaction.

We will explore issues related to personalization of control: how different people describe and show music differently, and how they expect the system to behave. We want to offer a personalized experience for each user, but at the same time benefit from the accumulating grounding of music descriptors that the system collects over time.

[1]Demo: `https://d3dbzxyywswxzm.cloudfront.net`.

We will also work to better understand user intent, whether they are providing control input, disengaged, or engaged but following and reflecting the music rather than trying to lead its performance. We plan to incorporate the user feedback into the training of the performance rendering models.

Another part is the user interface. The goal is to keep it minimal while making it more robust, user-friendly, and inviting. We would like to explore different visualization options that would complement and properly frame the musical functionality of the application.

Designing the system as a web application is a trade-off between the system accessibility and the complexity of the backend. Each active user currently consumes a non-negligible amount of compute resources on the backend for audio rendering and the expressive performance model that needs to be sustained. We will explore approaches to make the system financially viable.

While the current system shows promising results, its real value is yet to be validated. Human evaluation of the system is an important part of the future research.

7. CONCLUSIONS

In this work, we have presented a highly accessible interactive system for the co-creation of expressive music performances using speech and gestures. It allows users to interact with an autonomous, deep learning based piano performance rendering model in real-time through a mobile web application coupled with a backend. Our approach is able to integrate verbal and non-verbal human expressiveness, allowing people to project emotions and affects through music using the expressive language they practice every day. This makes our system accessible to people without musical training or the ability to play musical instruments, and makes complex musical works more widely available for performance and interpretation.

We believe that this work will contribute to the field of interactive music creation and performance, and allow a greater number of people to experience the joy of musical expression. We hope that the system can be used in educational contexts and make the musical tradition and practice more accessible, tangible, and engaging for young people. Since the web application does not require any setup on the user's part, our system is easy to try out. If it produces interesting results right away, it has a chance of being used by many people. Music therapy is another area where we hope to contribute to.

8. ACKNOWLEDGMENTS

We thank Dmitry Yarotsky for his valuable comments during the development of our system and feedback on the preliminary version of the paper.

9. REFERENCES

[1] A. Agostinelli, T. I. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi, M. Sharifi, N. Zeghidour, and C. Frank. MusicLM: Generating Music From Text. *arXiv preprint arXiv:2205.05448*, 2023.

[2] C. Benetatos, J. VanderStel, and Z. Duan. BachDuet: A Deep Learning System for Human-Machine Counterpoint Improvisation. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 635–640, Birmingham, UK, July 2020. Birmingham City University.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020.

[4] G. Brunner, A. Konrad, Y. Wang, and R. Wattenhofer. MIDI-VAE: Modeling Dynamics and Instrumenta- tion of Music with Applications to Style Transfer. *arXiv preprint arXiv:1809.07600*, 2018.

[5] C. E. Cancino-Chacón. *Computational Modeling of Expressive Music Performance with Linear and Non-linear Basis Function Models*. PhD thesis, Johannes Kepler University Linz, Austria, December 2018.

[6] C. E. Cancino-Chacón, M. Grachten, W. Goebl, and G. Widmer. Computational Models of Expressive Music Performance: A Comprehensive and Critical Review. *Frontiers in Digital Humanities*, 5:25, 2018.

[7] K. Choi, C. Hawthorne, I. Simon, M. Dinculescu, and J. Engel. Encoding Musical Style with Transformer Autoencoders. *arXiv preprint arXiv:1912.05537*, 2019.

[8] S. Dadman, B. A. Bremdal, B. Bang, and R. Dalmo. Toward Interactive Music Generation: A Position Paper. *IEEE Access*, 10:125679–125695, 2022.

[9] R. Dannenberg, D. Siewiorek, and N. Zahler. Exploring Meaning And Intention In Music Conducting. In *Proceedings of the International Computer Music Conference, ICMC 2010*, pages 327–330, 01 2010.

[10] R. L. de Mantaras and J. L. Arcos. AI and Music: From Composition to Expressive Performance. *AI Magazine*, 23(3):43, Sep. 2002.

[11] Donahue, Chris and Simon, Ian and Dieleman, Sander. Piano genie. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 160–164, 2019.

[12] R. Fiebrink, D. Trueman, and P. R. Cook. A Meta-Instrument for Interactive, On-the-Fly Machine Learning. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2009.

[13] F. Foscarin, A. Mcleod, P. Rigaux, F. Jacquemard, and M. Sakai. ASAP: a Dataset of Aligned Scores and Performances for Piano Transcription. In *Proceedings of the 21st International Society for Music Information Retrieval Conference*, 2020.

[14] A. Friberg, R. Bresin, and J. Sundberg. Overview of the KTH rule system for musical performance. *Advances in cognitive psychology*, 2(2):145, 2006.

[15] C. Hernandez-Olivan, J. Hernandez-Olivan, and J. R. Beltran. A Survey on Artificial Intelligence for Music Generation: Agents, Domains and Perspectives. *arXiv preprint arXiv:2210.13944*, 2022.

[16] C.-Z. A. Huang, C. Hawthorne, A. Roberts, M. Dinculescu, J. Wexler, L. Hong, and J. Howcroft. The Bach Doodle: Approachable music composition with machine learning at scale. *arXiv preprint arXiv:1907.06637*, 2019.

[17] D. Jeong, T. Kwon, Y. Kim, K. Lee, and J. Nam. VirtuosoNet: A Hierarchical RNN-based System for

Modeling Expressive Piano Performance. In *Proceedings of the 20th International Society for Music Information Retrieval Conference*, pages 908–915, 2019.

[18] S. N. Kelly. Using Conducting Gestures to Teach Music Concepts A Review of Research. *Update: Applications of Research in Music Education*, 18(1):3–6, 1999.

[19] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[20] A. Kirke and E. R. Miranda. *Guide to Computing for Expressive Music Performance*. Springer, 2013.

[21] R. Louie, A. Coenen, C. Z. Huang, M. Terry, and C. J. Cai. Novice-AI Music Co-Creation via AI-Steering Tools for Deep Generative Models. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery.

[22] A. Maezawa, K. Yamamoto, and T. Fujishima. Rendering Music Performance With Interpretation Variations Using Conditional Variational RNN. In *Proceedings of the 20th International Society for Music Information Retrieval Conference*, 2019.

[23] T. R. Næss and C. P. Martin. A Physical Intelligent Instrument using Recurrent Neural Networks. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 79–82, Porto Alegre, Brazil, June 2019. UFRGS.

[24] C. Palmer. Music performance. *Annual review of psychology*, 48(1):115–138, 1997.

[25] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[26] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust Speech Recognition via Large-Scale Weak Supervision. *arXiv preprint arXiv:2212.04356*, 2022.

[27] S. Rhyu, S. Kim, and K. Lee. Sketching the Expression: Flexible Rendering of Expressive Piano Performance with Self-Supervised Learning. *arXiv preprint arXiv:2208.14867*, 2022.

[28] J. Rink. *Musical Performance: A Guide to Understanding*. Cambridge University Press, 2002.

[29] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, and D. Eck. A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music. In *Proceedings of the International Conference on Machine Learning*, pages 4364–4373. PMLR, 2018.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[31] D. von Rütte, L. Biggio, Y. Kilcher, and T. Hoffman. FIGARO: Generating Symbolic Music with Fine-Grained Artistic Control. *arXiv preprint arXiv:2201.10936*, 2022.

[32] G. Widmer and W. Goebl. Computational Models of Expressive Music Performance: The State of the Art. *Journal of New Music Research*, 33(3):203–216, 2004.

[33] S.-L. Wu and Y.-H. Yang. MuseMorphose: Full-Song and Fine-Grained Piano Music Style Transfer with One Transformer VAE. *arXiv preprint arXiv:2105.04090*, 2021.

[34] B. Yu, P. Lu, R. Wang, W. Hu, X. Tan, W. Ye, S. Zhang, T. Qin, and T.-Y. Liu. Museformer: Transformer with Fine-and Coarse-Grained Attention for Music Generation. *arXiv preprint arXiv:2210.10349*, 2022.

[35] Y. Zhang, G. Xia, M. Levy, and S. Dixon. COSMIC: A Conversational Interface for Human-AI Music Co-Creation. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Shanghai, China, June 2021.