I Am Sitting in a (Latent) Room

Nicholas Shaheed
Stanford University
660 Lomita Drive
Stanford, California 94305
nshaheed@ccrma.stanford.edu

Ge Wang
Stanford University
660 Lomita Drive
Stanford, California 94305
ge@ccrma.stanford.edu

ABSTRACT

In this paper we describe *I Am Sitting in a (Latent) Room*, a real-time structured group improvisation system inspired by Alvin Lucier's "I Am Sitting in a Room," and the general process of degrading sound by repeatedly passing it through an acoustic medium. But there is a twist. Unlike "I Am Sitting in a Room," which unfolds as a gradual process with no further interaction once the process has begun, I Am Sitting in a (Latent) Room gives the improvisers the ability to intervene and interact with the process of degradation in real time. An audio clip is repeatedly encoded and decoded through two parallel instances of a bespoke variational autoencoder (VAE) model. On top of this process, the performers manipulates the model's latent embeddings in real-time, exploring the latent space (or "room") of the model over the course of the performance. Two performances with the composer and live-coding duo RGGTRN are presented. This work explores human-in-the-loop AI systems through group improvisation, interactive AI performance, and creating datasets as a part of the compositional process.

Author Keywords

Interactive machine learning, feedback, latent space

CCS Concepts

•**Applied computing** → *Sound and music computing;*
•**Computing methodologies** → *Machine learning;*
•**Human-centered computing** → *Interactive systems and tools;*

1. INTRODUCTION

The intertwining of looping, feedback, and musical process have had a profound impact on western art music from the 20^{th} century to the present. Steve Reich describes "pieces of music that are, literally, processes," where the processes happen slowly and gradually so that "listening to it resembles watching a minute hand on a watch–you can perceive it moving after you stay with it a little while." [20] Allowing one to sit with an iterative process and observe the subtle changes over time.

Alvin Lucier's "I Am Sitting in a Room" is a landmark work of process music, taking a recording of the composer's narration of a text, playing the recording back inside of a room, and then rerecording the audio with the acoustics of that space. This re-recording process produces a slow-moving feedback loop that warps and degrades the recorded text in small ways, compounding on each other with each iteration.

While Reich's manifesto (and Lucier's work) present this gradual process as a fully autonomous system that, once it begins, continues without human intervention, this paper introduces I Am Sitting in a (Latent) Room, a group improvisation system that takes this autonomous, gradual process and sticks an interactive system in the middle of it all. Existing in the context of, and at odds with, the purity of the process unfolding, while still channeling the core perceptive consideration of music that changes gradually so that one fixates on the subtle changes.

2. BACKGROUND

I Am Sitting in a (Latent) Room is a real-time structured group improvisation system that explores feedback, improvisation systems, and generative machine learning for audio synthesis.

2.1 Deep Learning for Audio Synthesis

Advancements in audio synthesis using deep-learning models have been numerous[11][9][27]. More recently these advancements have come from generative models[1][7], with architectures such as diffusion models[24], generative adversarial networks (GANs)[12], and variational autoencoders (VAEs)[14]. Particularly, advances in both computing speed and architectural improvements of models have resulted in models capable of generating audio faster than real-time[3][4]. As these techniques make their way into real-time system, they have seen adoption in NIMEs[25][5][19].

2.2 Latent Space in NIMEs

Many generative audio models encode high-dimensional data into a lower-dimension latent embedding. Points in this latent space can then be decoded by another model into the output format. Because it is possible to manipulate these latent values after an input audio has been encoded, these latent embeddings provide a rich intermediary between audio and features that have been explored in the NIME literature[23][22][13] . One salient aspect of latent embeddings is that they are learned automatically by the neural network, and their learned features do not necessarily correspond with prevailing cultural taxonomy of sounds. Efforts

Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Copyright remains with the author(s).

NIME'24, 4–6 September, Utrecht, The Netherlands.

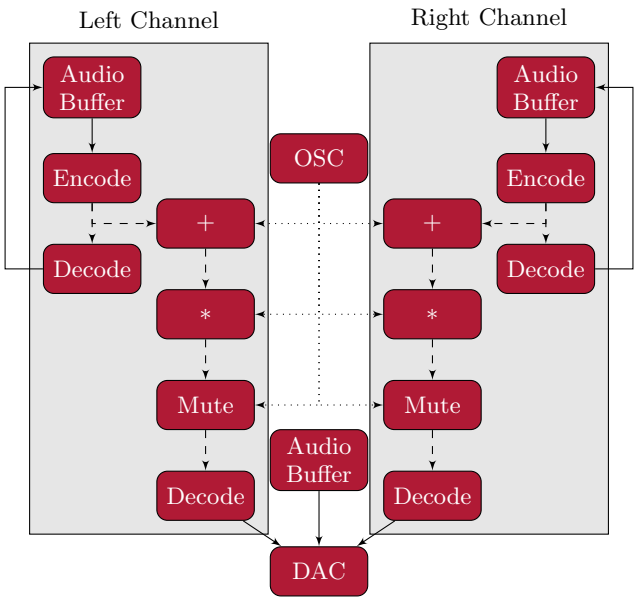

Figure 1: The signal flow of the I Am Sitting in a (Latent) Room system. Solid lines indicate an audio signal, dashed lines indicate latent embeddings, and dotted lines indicate OSC message.

have been made to improve controllability of these models[18][31] , but this paper seeks to explore this generally non-intuitive representation of sound as a rich means of control and interaction.

2.3 Interactive AI

A great deal of recent work around AI art and media involves the "Big Red Button" approach to its creation: the AI system receives some high-level human input and presents a finished work. However, interactive AI can be framed as systems integrating "human[s]-in-the-loop"[28], where the AI generates some intermediary result for one or more humans to interact with and curate, and then feed back into the AI system[2].

Examples of Interactive AI include Wekinator-enabled machine learning systems that enable iterative creation and real-time control of models, enabling "human-computer control and computer-human feedback"[10]. This approach of interactive, human-centric approaches to machine learning has been explored in NIME literature [16][26][8].

3. DESIGN AND IMPLEMENTATION

The system is built around a single 25-second loop (see Section 4), and the process of continuously encoding and decoding that loop through the autoencoder to degrade the sound. Figure 1 shows the flow of the audio signals, being encoded into latent embeddings, being modified via an OSC API, and being decoded back into audio. It is composed of three components: two independent degrading loops running simultaneously, and a buffer looping the original audio that seeds the degrading loop systems.

Each degrading looper has two parts; the feedback loop and the performer interventions. In the feedback loop, the original audio is initially stored in a buffer. In real-time, the buffer is encoded and decoded, and the decoded audio is written back into the buffer. This is done continuously over the course of the performance. Since the loop is a part of the model's training set, the reconstructed audio is fairly

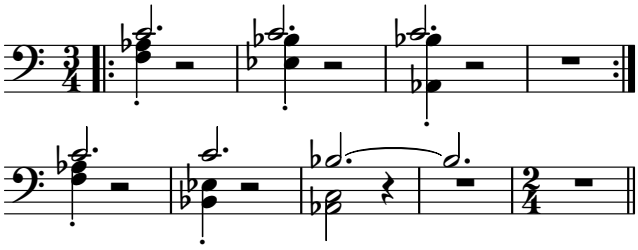

Figure 2: The score of the original loop, that is then run through the degrading looper.

close to the original. However, the reconstruction process is imperfect, and the resulting output ends up slightly different than the input. As this feedback loop continues, the sound slowly drifts from the original, slowly losing detail, warping, and eventually smoothing out until it reaches a steady state of continuously alternating between an A-flat and B-flat, a shadow of the original musical sequence (see Figure 2).

The encoded latent values are forked to a secondary signal chain, from here the performer interventions manipulate the latent values before they are decoded back into audio and played through the DAC.

3.1 The Model

The model used in I Am Sitting in a (Latent) Room is a Realtime Audio Variational autoEncoder (RAVE) model[3]. A VAE consists of an encoder and decoder model, trained on 63 minutes of audio with an RTX 3090 over a period of two days. The RAVE model encodes an input audio buffer of 2048 samples into a compressed, lower-dimensional (128-dimensional) latent space. The decoder takes a value in latent space and decodes it back into a 2048 sample output buffer. RAVE further reduces the dimensionality of the latent space via Singular Value Decomposition (SVD). For this model, audio is encoded into a 16-dimensional latent vector.

Most VAEs (including RAVE) use unsupervised learning - they are trained on unlabelled data. The training process consists of using the encoder-decoder structure to try and reconstruct the training data. The model attempts to minimize the difference between the input data and its output.

A key property of the latent space of VAEs is that a well-formed latent space is locally smooth: points in latent space that are close to each other (i.e. have a short distance) generally are decoded into outputs that are similar. In the case of audio, this means that close points in latent space should sound similar to one another, while further apart points will sound different.

3.2 Performer Interventions

There are three types of interactions: addition, multiplication, and muting. Because the latent embeddings are vector of floats, they can be treated as arbitrary numbers that can be manipulated with standard arithmetical operations. The interface for the system allows for these operations to be performed, and is transmitted from the player's client to the audio rendering server via OSC [30].

For addition, the performers individually address any of the latent vector's 16 dimensions and add a number to it. This results in a shift in the quality of the sound, changing the pitch/timbre/etc. of the audio. Different dimensions affect different aspects of the sound. Because the latent space was learned automatically as part of the training pro-

cess, these dimensions are not guaranteed to correlate with typical means of categorizing sounds (i.e. there is no dedicated pitch dimension), but because of the dimensionality-reduction performed on the latent space the lower-indexed dimensions encode the highest amount of variance and altering those dimensions will have a more dramatic effect than altering the higher indexed dimensions[3].

With multiplication, the entire latent vector is multiplied by a scalar value. This serves as a sort of exaggeration/diminution slider. If the scalar is <1, the sound both gets quieter and smooths out - there are fewer differences in the moment-to-moment sound. If the scalar is >1, the features of the audio get exaggerated, sometimes reaching a point in latent space the model does not have a clear embedding for, causing the audio to glitch or cut out.

Muting is effectively multiplication with a scalar of 0. Rather than setting the gain of the audio to 0, the vector becomes all 0s, outputting the sound the model has trained at that point in latent space.

Once the loop's latent values have been manipulated by the performers, the resulting latent values are then decoded and outputted to the DAC. The two degrading loopers are panned, and the looping buffer of the original recording in the center of the mix. The levels of all three components can be adjusted during performance. Since RAVE's encoder and decoder are not fully deterministic, in addition to the performers manipulating the two degrading loopers in different ways, the feedback loop process will degrade in slightly different ways.

3.3 Software

I Am Sitting in a (Latent) Room consists of a core audio rendering server, and three clients sending commands to the server via OSC. The server was written in ChucK[29], with inference of the RAVE models running via a UGen made with the ChuGin API[21]. This UGen is part of the ChAI (ChucK for AI) framework[15]. Each client corresponded to one player. Two of the clients were implemented in SuperCollider[17] and one in ChucK.

4. PHILOSOPHICAL UNDERPINNINGS

4.1 Artful Datasets

Machine learning models are intractably linked to the dataset it is trained on. Its qualities, results, and biases are all intimately tied to the data and how it interacts with the model's architecture during training. Because of this, in creating this system, we sought to make the building of the model's dataset an essential aspect of the creative process: the initial conceptions of the work informed the goals and choices made with what to include in the dataset, and in turn the resultant model informs the realization of the system in practice and performance. This channels Perry Cook's fifth principle of designing computer music controllers: "make a piece, not an instrument"[6].

The dataset consists of 63 minutes of audio divided into three categories: variations of the 25 second main loop. Field recordings of a train descending from Mount Koya on a stormy day in Wakayama Prefecture, Japan, and field recordings of Japanese cicadas.

The main loop was composed by the author, and made in Ableton Live. Approximately 25 minutes of variations of this loop were created using Ableton Live's note chance tools: notes were set to play 100%, 90%, 70% 50%, 30%, and 10% of the time. Multiple runs of the loop were used for each of the chance levels, augmenting the dataset with multiple different randomized variations of the loop. The intent of this is the create a cohesive "degredation space" - enough different note combinations of the loops ranging from the full sounds to a sparse, pointillistic texture. This is done to allow the model to learn a smooth representation of different densities. This was done because both because of the concept of the piece: trying to cause a smooth degradation via feedback loops, and knowing that latent embeddings work with spatial similarity - the closer a latent vector is to another latent vector, the more similar the two outputted audio should be. Thus, providing a smoother dataset with more points of similarity should result in a smoother outputted audio.

This portion of the dataset is constructed with the main gesture of the work in mind (the feedback loop). The next two categories of data served as points to further explore in the latent space to allow for a richer space of sounds to explore while manipulating the latent values in performance. The first is a set of field recordings of a train descending from Mount Koya on a stormy day in Wakayama Prefecture, Japan. These recordings feature a prominently pitched drone from the train rubbing against the sloped tracks. These were then pitch-shifted to augment the sounds with different pitches of drones.

The third portion of the dataset is a 10 minute recording of Japanese cicadas. With the main loop serving to fulfill the primary gesture of the derogatory feedback loop, the two additional categories of data, the train and cicada recordings, provide a more diverse range of sounds for the performers to explore. However, balance between amount of looping variations and auxiliary sounds in the dataset are needed - one attempt at training the model with significantly more auxiliary sounds from more diverse sound sources, but the same amount of looping sounds, yielded a main loop with noticeably less fidelity.

4.2 Aesthetic Description of the Work

The main loop of the work (Figure 2) is structured to be listened to repeatedly: slow motion, gentle voice leading, and with a two beat extension to the twelve bar phrase to lead into the next repetition. Timbrally, the loop consists of layered pads and cackling noises from a wavetable synthesizer. The result is the loop is both prominently pitched, but with enough noise to bridge between the timbral qualities of the loop and the noisier field recordings of trains and cicadas.

Because of the smoothness of the latent space in the RAVE model, both continuous motion and great timbral jumps are possible and navigating this gradient provides a key gestural space for the improvisation. One performer used a 16-knob midi controller mapped to the addition function of each individual dimension of the latent space, providing smooth motions. The two performers using SuperCollider utilized a mix of LFOs along both addition and multiplication, as well as randomized step sequencer - every step in the sequencer sent a random value to the first two dimensions of both models (i.e. the two highest variance dimensions). The magnitude of these random values is increased and decreased over the course of a performance. The effect of all of this is to move in and out of similarity to the reconstructed loop.

All of this provides a chance for play in a chaotic system: three people controlling 16-dimensions, that interact smoothly, but unintuitively and nonlinearly on top of a constantly shifting base. Unexpected events happen, with no one is ever totally in control, leaving plenty of space for happy accidents. But, because the model is relatively cohesive, and is centered around a singular idea a sense of

cohesion is maintained.

Another source of happy accidents is glitches in the system itself. Quirks in the code caused a number of compelling moments over the two performances, particularly at the end of the performance. In one performance the sound completely stopped from both models, resulting in a sudden and striking conclusion. In another performance the loop completely cut out, only to return mid-phrase. Both of these unexpected results from the system resulted in delightful subversions of the form established by the rest of the piece.

4.3 Humans in the Loop

4.3.1 Model Building

In contrast to many mainstream AI systems that are "Big Red Button (BRB) system[s]," [28] I Am Sitting in a (Latent) Room adopts an approach to AI as a medium of performance, instrument construction, and through multiple levels of human-in-the-loop approaches to interactive design. As discussed in 4.1, the construction of the datasets and the process of training the model form one level of the human-in-the-loop systems. Multiple variations of the dataset were created to test different versions of the model used in the system. Specifically, more field recordings were collected from fellow performers to provide different avenues to deviate from the base degradation process.

As well, the conception of the work and the sculpting of the created sounds and the curation of the field recordings come from a tacit knowledge of the RAVE architecture developed through the process of creating models from a variety of different types of sounds and working with them over the course of several projects and pieces. This back-and-forth of model building and hyperparameter tuning combined with critical listening of the outputed sounds (as opposed to relying on benchmarks or validation loss). While this iterative process is standard part of working with ML, the small amount of the needed data and the ability to train on a single consumer-level GPU over the course of 12 hours - 3 days when training a RAVE model mean that model building can happen at the scale of an individual working on a single project (this model was trained as part of a two-week residency). This is in contrast to LLMs and diffusion models which can require hundreds of thousands or millions of dollars of compute time on vast internet-scale amounts of data.

4.3.2 ...in the (Feedback) Loop

While the process part of the system is driven by the AI itself, the performers choose when, and how, to intervene. While the process is set, it ultimately provides a foundation, and a creative constraint, from which the piece and its form emerge from.

More poetically, humans are literally in the AI's loop: a VAE is trained by trying to reconstruct the audio of a dataset – i.e. minimize the loss of the output of the encoding-decoding procedure. This continuous attempt to loop back to the original sound is directly intercepted by the performers, through their manipulation of latent space.

4.3.3 Simplicity of Interaction

The system's modes of interaction with the latent space are simple and direct: addition, multiplication, and muting (which is just multiplying by 0). This channels Cook's sixth principle of designing computer music controllers: "instant music, subtlety later." [6] Because the encoding of the degrading audio already provides a firm bedrock of richer sounds, the performer's role can take on many different levels of involvement: from complex randomization and algorithmic sequences behaviors to slowly sculpting the sound with the knobs of a midi controller. With 18 parameters for each model (addition for each of the 16 latent dimensions, multiplication, and muting, yielding 34 parameters in total) and the relatively short rehearsal schedule of four rehearsals over the course of two weeks meant that this flexibility was a necessary part of creating a successful performance.

This low-level API additionally channels Cook's third principle: "copying an instrument is dumb, leveraging expert techniques is smart." Rather than giving the performers a prebuilt set of more complex interactions with the models, the lower-level design of the API mimics the types of parameter manipulations that are the bread and butter of the live coder's toolbox. Additionally, this more open-ended approach offered more possibilities, gave performers creative agency, and leveraged their existing skillset in Supercollider and artistic language as live coders to make the experience a more collaborative one.

4.4 Performance

The premiere performance of the I Am Sitting in a (Latent) Room system was an audiovisual group improvisation titled "degr d t n." [1] Three performers, one of the authors and two members of Mexican live coding collective RGGTRN gave both a premiere performance as part of a residency with the duo and a recording session of the work. The performance consisted of the trio manipulating both I Am Sitting in a (Latent) Room and a TouchDesigner patch (designed by Emilio Ocelotl) performing live manipulations of various videos of flowing water/nature for the visuals.

The performance consisted of three separate computers, with two players interacting with the system using Supercollider, communicating via OSC over a local network connection. The third player used a MIDI controller, using knobs to control the different parameters. The system was split amongst the three performers to minimize parameter overlap between the three players.

5. ETHICAL CONSIDERATIONS

There are many new and unanswered ethical and aesthetic questions regarding the use of AI, including the use of generative models in artistic fields. There remains more questions than answers regarding authorship, displacement of labor, aesthetic and cultural implications, and the role of advanced computational machinery in human creative endeavors.

One virtue this paper prioritizes is working with AI at an individual human scale: building a holistic system from its software to its model, to its performance as an artistic endeavor. The model used in I Am Sitting in a (Latent) Room was trained on a single, admittedly quite expensive, consumer-grade GPU. It's dataset was composed and recorded by a single person, and ultimately resulted in three artists coming together to perform music. In contrast to many tools built with generative AI that seek to supplant or remove humans from creative efforts, I Am Sitting in a (Latent) Room offers a system that is channeling and facilitating human creativity, connection, and art making at every stage.

[1] A recording of the performance can be found at `https://vimeo.com/943876017`

6. CONCLUSION

There is an irony in the design of "I Am Sitting in a (Latent) Room" and it has to do with its insistence for human interaction. Lucier's "I Am Sitting in a Room", once started, is "fully automatic", it fits the definition of Reich's "music as a gradual process" [20]. In this sense, Lucier's original resembles a "Big Red Button"; while the sound loops and degrades, human interaction is not in the loop. This setup is an essential aspect of the piece as it allows the room acoustics to reveal itself, its resonances eventually even dominating the overall sound. Lucier's piece, of course, employed no AI. On the other hand, "I Am Sitting in a (Latent) Room" swaps the "room" for a "latent space (room)". This piece is made with AI at a time where much AI research is racing ahead for more Big Red Buttons, with perhaps no real regard for how much automation is what we'd want in the first place. AI development largely seems to embody the ethos of "move fast; break things", far outpacing the awareness to slow down and critically think about things. Yet, I Am Sitting in a (Latent) Room insists on human interaction, even when its role is not quite needed (no one asked for this!). So, what we have is a situation where Lucier's original is a Big Red Button but involves no AI whatsoever, and I Am Sitting in a (Latent) Room is full of AI but explores moving away from Big Red Buttons, trendy as they may be.
Not sure what all this quite means, but this is where we are. The experiment continues.

7. ETHICAL STANDARDS

This paper complies with the NIME ethical standards. No human or animal participants are involved.

8. REFERENCES

[1] A. Agostinelli, T. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi, et al. Musiclm: Generating music from text. 2023.

[2] S. Amershi, M. Cakmak, W. B. Knox, and T. Kulesza. Power to the people: The role of humans in interactive machine learning. *AI Magazine*, 35(4):105–120, Dec. 2014.

[3] A. Caillon and P. Esling. Rave: A variational autoencoder for fast and high-quality neural audio synthesis. *arXiv preprint arXiv:2111.05011*, 2021.

[4] A. Caillon and P. Esling. Streamable neural audio synthesis with non-causal convolutions. *arXiv preprint arXiv:2204.07064*, 2022.

[5] C. Carr and Z. Zukowski. Generating albums with samplernn to imitate metal, rock, and punk bands. *arXiv preprint arXiv:1811.06633*, 2018.

[6] P. Cook. 2001: Principles for designing computer music controllers. *A NIME Reader: Fifteen years of new interfaces for musical expression*, pages 1–13, 2017.

[7] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez. Simple and controllable music generation. *arXiv preprint arXiv:2306.05284*, 2023.

[8] M. O. DeSmith, A. Piepenbrink, and A. Kapur. Squishboi: A multidimensional controller for complex musical interactions using machine learning. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 353–356, 2020.

[9] J. Engel, C. Resnick, A. Roberts, S. Dieleman, M. Norouzi, D. Eck, and K. Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *International Conference on Machine Learning*, pages 1068–1077. PMLR, 2017.

[10] R. A. Fiebrink. *Real-time human interaction with supervised learning algorithms for music composition and performance*. Princeton University, 2011.

[11] F. Ganis, E. F. Knudesn, S. V. Lyster, R. Otterbein, D. Südholt, and C. Erkut. Real-time timbre transfer and sound synthesis using ddsp. *arXiv preprint arXiv:2103.07220*, 2021.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[13] R. Ishino and N. Tokui. Miami: A Mixed Reality Interface for AI-based Music Improvisation. *International Conference on New Interfaces for Musical Expression*, jun 22 2022. https://nime.pubpub.org/pub/9af67f6s.

[14] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *Stat*, 1050:1, 2014.

[15] Y. Li and G. Wang. Chai: Interactive ai tools in chuck. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2024.

[16] L. McCallum and M. S. Grierson. Supporting interactive machine learning approaches to building musical instruments in the browser. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 271–272. Birmingham City University Birmingham, UK, 2020.

[17] J. McCartney. Rethinking the computer music language: Super collider. *Computer Music Journal*, 26(4):61–68, 2002.

[18] A. Pati and A. Lerch. Is disentanglement enough? on latent representations for controllable music generation. *arXiv preprint arXiv:2108.01450*, 2021.

[19] T. Pelinski, V. Shepardson, S. Symons, F. S. Caspe, A. L. B. Temprano, J. Armitage, C. Kiefer, R. Fiebrink, T. Magnusson, and A. McPherson. Embedded ai for nime: Challenges and opportunities. In *International Conference on New Interfaces for Musical Expression*. PubPub, 2022.

[20] S. Reich. Music as a gradual process. *Writings on Music, 1965-2000*, pages 34–36, 1968.

[21] S. Salazar and G. Wang. Chugens, chubgraphs, chugins: 3 tiers for extending chuck. In *ICMC*, 2012.

[22] H. Scurto and L. Postel. Soundwalking deep latent spaces. In *Proceedings of the 23rd International Conference on New Interfaces for Musical Expression (NIME'23)*, 2023.

[23] V. Shepardson and T. Magnusson. The living looper: Rethinking the musical loop as a machine action-perception loop. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2023.

[24] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015.

[25] K. Tahiroğlu, M. Kastemaa, O. Koli, et al. Al-terity: Non-rigid musical instrument with artificial intelligence applied to real-time audio synthesis. In *Proceedings of the international conference on new interfaces for musical expression*, pages 337–342, 2020.

[26] A. Tsiros and A. Palladini. Towards a human-centric design framework for ai assisted music production. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 399–404, 2020.

[27] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. In *Arxiv*, 2016.

[28] G. Wang. Humans in the loop: The design of interactive ai systems. *J. Artificial Intelligence Res*, 64:243–252, 2019.

[29] G. Wang, P. R. Cook, and S. Salazar. Chuck: A strongly timed computer music language. *Computer Music Journal*, 39(4):10–29, 2015.

[30] M. Wright, A. Freed, et al. Open soundcontrol: A new protocol for communicating with sound synthesizers. In *ICMC*, 1997.

[31] S.-L. Wu, C. Donahue, S. Watanabe, and N. J. Bryan. Music controlnet: Multiple time-varying controls for music generation. *arXiv preprint arXiv:2311.07069*, 2023.