Designing & Evaluating Interactive Generative Audio Models for Sound Design

Purnima Kamath

A thesis submitted for the degree of
Doctor of Philosophy

Department of Communications & New Media
National University of Singapore
2024

Thesis Advisors:
Dr. Kokil Jaidka, National University of Singapore
Dr. Suranga Nanayakkara, National University of Singapore

Examiners:
Dr. Alex Mitchell, National University of Singapore
Dr. Subhayan Mukerjee, National University of Singapore

Declaration of Authorship

I hereby declare that this thesis is my original work, and I have written it in its entirety. I have duly acknowledged and attributed all the sources of information used in this thesis.

This thesis has also not been submitted for any degree in any university previously.

Purnima Kamath

July, 2024

Dedication

Dr. Rajalaxmi Kamath and Akshaya Kamath,
for inspiring me on this Ph.D. journey.

Acknowledgements

First and foremost, I would like to thank all my advisors for their support throughout my Ph.D. journey. I want to thank Suranga Nanayakkara for taking me on as a Ph.D. student, introducing me to the fascinating world of human-computer interaction, and giving me a place to dock and a team to belong to at the Augmented Human Lab. I am thankful for his encouragement and constant push towards thinking about interactivity and human-centeredness while developing the algorithms in this thesis. I gained much experience in writing and presenting from him, and I am forever grateful for it.

I want to thank Kokil Jaidka for having faith in me and taking me on as a Ph.D. student midway through my candidature, pushing me to always do more, and helping me persevere through the last few crucial years of my Ph.D. I am grateful for her invaluable advice and guidance in navigating my career during and beyond this Ph.D. and for giving me the flexibility to pursue my own research interests.

Additionally, I want to thank Lonce Wyse for being my supervisor during the initial years of my Ph.D. and introducing me to the world of generative modeling, audio textures, controllability, and morphing.

I want to thank Chitralekha Gupta for collaborating with me on numerous projects, always being available to soundboard ideas, and sharing my energy and excitement during those discussions. I thank my National University of Singapore co-authors, Zhuoyao Li, Yize Wei, Priambudi Lintang Bagakara, and Tasnim Nishat Islam, for persevering through all those projects with me. I want to thank Fabio Morreale from the University of Auckland for his collaboration and guidance while conducting the qualitative research in this thesis. I would also like to thank all the sound design experts involved during the qualitative user studies in my research for their time and valuable insights on AI-based Creative Support Tools during our discussion.

I want to thank my lab mates at the Augmented Human Lab—Sankha Cooray, Malsha de Zoysa, Dinithi Dissanayake, Moritz Messerschmidt, Hannah Qiao, Hussel Suriyaarachchi, Qin Wu, Phoebe Chua, Mia Nguyen, Shreyas Sridhar, Prasanth Sasikumar, Dixon Prem Daniel, Felicia Tan, Michele Wu, Suveen Ellawela, Hyung Woon Lee—for fostering a friendly and supportive community. I will forever appreciate and cherish our time together during our reading circles, trips, excursions, and tea breaks during my candidature. I want to thank my CNM-graduate conference gCON 2021 co-organizers-Yuanyuan Wu, Francis Luis Torres, Zishan Lai, Qiaofei Wu, and Samseer Mambra-for their collaboration and support in bringing our department's flagship conference together in 2021. I also want to thank my other CNM mates, Nisar Keshwani, Paul Jerusalem, Curie Roe, and Stanley Wijaya, for their friendship, support, and camaraderie while navigating complex communication topics during the early stages of our research careers.

Finally, I am grateful for the support of my family, especially my husband, Akshaya Kamath. He has been my constant champion, mentor, and advisor throughout this Ph.D. I appreciate his support in putting our life on hold for the last four years, for tolerating my many moods through this journey, enduring the ups and downs of paper acceptances and rejections with me, and for his late-night coffee and snack supplies. I am also grateful to Rajalaxmi Kamath for nudging me on to this Ph.D. journey and for her guidance and wisdom in helping me through those many moments of self-doubt. My Ph.D. studies have occurred at a stage in my life when I should be working towards supporting and taking care of my parents and in-laws. I am forever grateful to them for supporting my drive and determination to pursue my ambitions at this stage of my life and theirs.

Contents

Abstract

Sound design involves creatively using sounds to build cinematic experiences for films and games. It includes creating and manipulating environmental sounds, maintaining large sound effects databases, and dealing with challenges in modifying recorded sounds. This thesis aims to design and evaluate AI-based creative support tools to assist sound designers in controlling and editing the semantic properties of generated sounds, providing novel avenues for creative sound exploration and discovering new sounds for use in their creative projects.

Designing and evaluating generative audio AI models for sound design poses many interaction design challenges. One challenge stems from the current lack of semantically labeled environmental sound datasets. Another challenge is the lack of support for creative tasks such as morphing two or more sounds. Thus, in the first part of this thesis, I develop and evaluate methods for granular semantic guidance using models trained on unlabeled datasets and explore novel algorithms for the creative task of sound morphing using pre-trained generative models.

In the second part of this thesis, I explore methods to evaluate generative models and algorithms for sound design. The evaluation metrics literature lacks methods to perceptually evaluate sounds generated by semantically guided or controllable generative audio AI models. This thesis outlines novel methods to evaluate such AI models using perceptual listening tests. Furthermore, I investigate the opportunities and challenges of applying such AI-based creative support tools for the professional practice of sound design.

In this thesis, I incorporate human-centered design principles to design digital tools for creativity in sound design. In summary, the contributions from this work are as follows:

- The design and evaluation of AI-based creative support tools that encourage novel sound exploration, perform semantic edits, and support creative tasks for sound design, such as morphing.
- The design and evaluation of novel methods to perceptually evaluate outputs from AI-based creative support tools for sound design.

- Empirical findings from qualitative user studies for creative engagement conducted with expert sound designers.
- Human-centered design insights and recommendations to guide future research on AI-based tools for creativity support in sound design.

Through this work, I demonstrate the potential to design steerable creative support tools for sound design using generative audio AI. I design affordances using tools that enable sound designers to achieve their creative goals by using semantically relevant attributes or properties of the environmental sounds they want to generate. Additionally, I develop methods to perceptually evaluate the steerable models and their creative output using subjective listening tests on crowdsourced platforms. I also study the challenges and opportunities of applying such models in a practice-oriented sound design environment. Finally, I highlight this work's limitations and discuss potential future directions for designing and applying generative audio models for sound design. Through this work, I offer novel human-centered ways to design and evaluate future AI-based creative support tools for sound design.

List of Tables

List of Figures

Symbols and Acronyms

Symbols

$\mathbb{R}$	Set of Real numbers.
x	Scalars, such that $x \in \mathbb{R}$. Lower-case.
$\mathbf{x}$	Vectors, such that $\mathbf{x} \in \mathbb{R}^n$. Lowercase, bold.
$\mathbf{X}$	Matrices, such that $\mathbf{X} \in \mathbb{R}^{n \times n}$. Uppercase, bold.
$\mathbf{X}^T$	Matrix, transposed.
$\{\mathbf{w}_0, ..., \mathbf{w}_n\}$	Set of n $\mathbf{w}$ vectors.
$\mathcal{N}(0, 1)$	Standard unit gaussian distribution.
$\mathcal{Z}, \mathcal{W}$	Euclidean spaces (Used to denote GAN latent spaces in this thesis).
$F(.)$	Function of a scalar, vector, or a matrix.
$\mathbb{E}_{\mathbf{z} \sim \mathcal{N}(0,1)} F(\mathbf{z})$	Expectation of $F(\mathbf{z})$, given $\mathbf{z}$ has distribution $\mathcal{N}(0, 1)$.
$\|\mathbf{x}\|_2$	L2 norm of $\mathbf{x}$.
$\sum_n$	Sum over n.
(a, b)	Open interval between a, b. Excluding a, b.
$[a, b]$	Closed interval between a, b. Including a, b.
$\mathbf{a} \cdot \mathbf{b}$	Dot product between vectors $\mathbf{a}$ and $\mathbf{b}$.
$\mathbf{a} \odot \mathbf{b}$	Hadamard product between vectors. Element-wise multiplication.
$\mathbf{ab^T}$	Vector multiplication, given $\mathbf{a}$, $\mathbf{b} \in \mathbb{R}^n$.

Acronyms

2AFC	Two-Alternative Forced Choice
nD	$n-$ dimensional, e.g.,1-dimensional/2-dimensional
AMT	Amazon Mechanical Turk
AWS	Amazon Web Services
CDN	Content Delivery Network
CLAP	Contrastive Language-Audio Pretraining model
CST	Creative Support Tools
DAC	Digital to Analog Converter
DAW	Digital Audio Workstation
DDPM	Denoising Diffusion Probabilistic Models
DDSP	Differentiable Digital Signal Processing
DFT	Discrete Fourier Transform
FD	Fréchet Distance
FID	Fréchet Inception Distance
FAD	Fréchet Audio Distance
GAN	Generative Adversarial Network
GPU	Graphic Processing Unit. Device for number crunching
Hz	Hertz. Measurement of frequency
IF	Instantaneous Frequency (as in IF representations)
IS	Inception Score
LDM	Latent Diffusion Model
LPIPS	Learned Perceptual Image Patch Similarity metric
MICI	Mixed-Initiative Creative Interfaces
MIDI	Musical Instrument Digital Interface
MSE	Mean Squared Loss
MUSHRA	Multiple Stimuli with Hidden Reference and Anchor
PGAN	Progressive GAN
PGHI	Phase Gradient Heap Integration
ResNet	Residual Network
RNN	Recurrent Neural Network
SeFa	Semantic Factorization
SFX	Sound Effects
SMF	Sound Model Factory
SPA	Single Page Applications
STFT	Short-Time Fourier Transform
SOM	Self-Organizing Maps

TA	Thematic Analysis
TDP	Thermal Design Power. Measurement for thermal efficiency of a device
TF	Time-Frequency (as in TF representations)
TTA	Text-to-Audio
UMAP	Uniform Manifold Approximation & Projection
VAE	Variational Autoencoder
VGG	Visual Geometry Group (deep convolutional networks)
XAI	Explainable AI

Chapter 1

Introduction

Imagine you are walking in a forest. Your footsteps crunch as you walk across the forest floor. You can hear the cicadas trilling. There is a slight drizzle, and you can hear the sound of the raindrops. Our sense of perception of the world surrounding us relies on our ability to direct our attention to such sounds. These sounds made by us interacting with our surrounding environment are the sounds of immediacy and our physical presence [1]. The use of such sound in films and games is therefore crucial in creating an "atmosphere," mood, or feeling that enhances the cinematic experience for us as its consumers [2–6]. It complements the visual information presented and communicates additional subliminal non-verbal cues about the film's environment [7].

Environmental sound effects such as wind, rain, or footsteps provide subtle sonic cues about the environment and play a vital role in reinforcing the perception of reality and immersing the viewer in the film's narrative [4, 8]. Most of the environmental sounds that we hear in movies, such as the sound of the actors' movements or footsteps, etc., are usually not recorded on location but added in post-production [9, 10], i.e., in a studio using sound editing tools. Augmenting films with sounds in post-production falls under the purview of craftspeople such as sound designers [2, 3, 7].

Sound designers create new sounds or use existing sound effect recordings to augment video recordings. This involves synthetically generating or using pre-existing recorded sounds using technical tools such as Digital Audio Workstations (DAWs). For instance, it is a common sound design practice to synthesize the sound of a sword swooshing by processing or filtering white noise using signal processing algorithms [6]. Or editing an impact sound by layering animal growls underneath the sounds of an explosion to make the effect more hyper-real or powerful [11]. Further, sound designers make use of Foley

sound techniques when necessary. These techniques involve using physical materials to generate sounds synchronized to the video recordings.

Designing digital tools for sound design that enable craftspeople to use environmental sounds creatively poses a few interaction design challenges. Sound design tools should allow some form of interaction or control of a sound and facilitate a suitable sound space for novel sound exploration instead of asking designers to search through a large sound effect library [6]. Currently, from a technological perspective, editing or modifying a pre-existing recorded sound based on its perceptually relevant properties is difficult when using traditional sound editing tools [12]. In the last few years, generative AI models based on deep neural networks for music have successfully demonstrated their capabilities in generating novel creative musical artifacts [13, 14]. Such algorithms are steered or controlled using semantics such as the type of instrument to be used in the composition or the pitch of the sound [15, 16]. These innovations pave the way for developing steerable generative models for environmental sound effects, enabling novel sound exploration and the development of Creative Support Tools (CSTs) [17] for sound design.

This thesis investigates approaches to designing and implementing steerable generative AI models and CSTs for sound design. We also investigate ways to perceptually evaluate such steerable models and build our understanding of the challenges and opportunities of applying such models in a practice-oriented sound design environment.

1.1 Motivation

Our everyday sonic environment is usually composed of music, speech and a myriad of environmental sounds [4]. Often, sound designers work with environmental sounds that lack the rhythmic and harmonic structures normally found in musical compositions. Let's consider an example where a sound designer is tasked with augmenting a video with the sound of footsteps. The sound of the footsteps is governed by the semantic and material properties of the environment where the footsteps occur. I use the term *semantic properties* to describe the attributes of audio that affect human perception of sound [18]. I use this term to describe the properties of non-musical sounds that cannot be described by acoustic attributes such as pitch or loudness. For instance, semantic properties of the sound of the footsteps are affected by the material of the floor, whether it is a hard metal or a wooden surface or whether it is dry grass or snow. Further,

properties such as the type of shoes worn by the walker, whether the shoes are high heels or boots, etc., can also impact the sound. While designing sounds for a film, a sound designer works towards creating a believable and persuasive sound effect that matches the material environment and the timing and pace of the actor's movement across the screen.

Previously, generative AI models for text and images have demonstrated their ability to generate novel, diverse, and high-quality artifacts. Such models are becoming increasingly integral to creative practices in the arts and have moved from exclusively being a research endeavor to finding practical applications [19, 20]. Such models can also be semantically steered or controlled and have been shown to assist in human-AI co-creation [21–28] successfully. Thus, in this thesis, we aim to design and evaluate generative audio AI algorithms as CSTs to assist sound designers in their creative practice.

1.1.1 Technical Challenges

Although existing generative AI algorithms from the domains of text, images, or music can provide productive avenues for researching generative models and CSTs for audio, their adoption for sound design has a few challenges:

Lack of strongly labeled environmental sounds datasets for training generative models: Semantically controllable, *"guided"* or *"steerable"* deep neural networks require training on large, strongly labeled datasets. Steerability or guidance here alludes to the ability of the algorithm to controllably generate sounds with semantically relevant attributes and perform continuous fine-grained semantic attribute edits to the generated sound. There is currently a lack of large semantically well-labeled environmental sound datasets. This is because, while large datasets of environmental sounds can be readily recorded in the wild, semantically labeling them is expensive, time-consuming, and prone to errors due to human annotator subjectivity [29, 30]. Therefore, there is a need to design steerable generative algorithms for sound design that provide semantic attribute guidance without the supervision of labeled data.

Need for support for sound design-specific creative tasks such as morphing: Sound designers often strive to create new and distinct sound effects for their creative work. Designers usually record new sounds or use techniques such as *"sound morphing"* [31] to generate new source material. Sound morphing refers to the process of gradually transforming one sound into another to generate novel sounds and hybrid timbres. Such techniques are useful for creating unreal but plausible sound effects, such as those

made by fantastical alien-like creatures in movies [31]. Although current state-of-the-art generative models can generate high-quality plausible sounds, they are not designed to support other creative tasks, such as morphing or the pursuit of creating semantically *"in-between"* or hybrid sounds, which are crucial to supporting the creative practice of sound design. Therefore, there is a need to design generative algorithms for sound design that can provide novel exploration methods to generate semantically hybrid sounds.

Lack of perceptual methods to evaluate the descriptive qualities of sound: Generative algorithms for sound design must be evaluated on their ability to generate sounds based on descriptive qualities such as smoothness or goodness of the morphed sound or realism or plausibility of the generated sound [13, 32–36]. Further, sound design tools should be evaluated for their steerability or ability to control or semantically edit the generated sound's user-defined descriptive semantic properties such as *"brightness"* or *"tinniness"* (to indicate the presence of high pitched components). Typically, generative models are evaluated using objective quality metrics such as lack of distortion or noise in the generated artifacts [37–44]. Such objective metrics are faster to evaluate but fail to find meaningful differences between descriptive perceptual measures. Further, they fail to evaluate the ability of the model to semantically edit sounds continuously and in a fine-grained way. Therefore, there is a need to devise novel ways to evaluate sound design algorithms perceptually for the descriptive qualities of their generated sounds and their ability to steer generation using subjective listening tests.

Need to understand the opportunities for generative models in sound design practice: Generative models are well studied for their potential to support co-creation in the human-AI interaction literature for music [15, 16]. And yet, despite the growing adoption of such models as co-creation tools for music production [45], very few empirical studies exist to assess their potential to offer new possibilities to the practice of sound design. Further, most human-AI interaction studies for audio focus on the applicability of steerable interfaces to empower novice users in their creative goals [15, 16, 46–48]. Expert sound design practitioners spend years developing their creative design process and building inventories of sounds to apply in their next design project [4]. As such, their needs, expectations, and ways of working with AI-based tools differ from those of novices. Thus, it is necessary to explore how generative audio models can assist expert sound designers in their creative practice using specifically designed studies.

The above challenges pave the way for this thesis's computational aims. In the next section, I expand on these aims and ground them in a human-centered approach for designing and evaluating AI-based creative support tools for sound design.

1.1.2 Human-Centeredness in Design

> *"all art algorithms, including methods based on machine learning, are tools for artists; they are not themselves artists."*
>
> —Aaron Hertzmann, *In Arts, 2018* [22]

Ben Shneiderman's human-centered AI (HCAI) interaction design philosophy [49] argues for developing AI with a human-centered orientation. That is, while innovations in AI technologies bring high automation, there is also a need for a high level of human control to build trustworthy systems. Further, while most AI projects often aim to replace humans or fully automate tasks, he proposes to add value by building "supertools" to amplify or boost human abilities. This design philosophy is central to the aims of this thesis. Instead of creating entire compositions and automating sound design pipelines, this thesis aims to design and implement steerable AI-based Creative Support Tools (CSTs) that assist sound designers in creating individual sound units to employ in their creative compositions. The sound designers will eventually be the creators and owners of the resulting creative work product.

Digital tools are increasingly becoming integral to all aspects of our lives [50], especially in creative art practices [51]. As creative practitioners in the domain of new media arts increasingly demand more effective digital computer systems, tool designers must incorporate more human-centered design principles when developing such digital tools. I list a few below:

Encouraging exploration: Tools for creativity should empower users to explore the system's potential, starting with a basic set of capabilities and allowing them to go deeper when needed [17]. Current AI-based tools offer exploration using a pre-defined set of controls, typically designed by the system's developers. There is thus a need to design CSTs to offer the ability to explore the AI-generated design space based on semantics defined by the system's end-user.

Steering by way of interactive controls: A central tenet of the HCAI design is steering intelligent supertools using control panels that enable fine-grained control [17, 49, 52]. These control panels resemble the steering controls for other interactive systems, such as video games. They can be slider-based fader controls or knobs, such as those found on modern music synthesizers, to guide or steer the output generated by the AI-based tool. Currently, AI-based tools that support text-based interactions usually do not afford the ability to granularly steer generation for creative tasks that need fader-like controls. This research aims to explore avenues to address this gap.

Designing for non-experts: Another key design principle for human-AI interaction is accommodating users of all experience levels [17, 52]. With the rapid advancements in AI technologies, the research community is increasingly emphasizing using crowdsourced platforms to evaluate the outputs of such algorithms. Often, participants on such platforms are not audio experts and may not understand technical jargon such as "morphs" or the quality of "sound progression". Thus, this research aims to design perceptual evaluation procedures that minimize the complexity during evaluation for non-experts.

Designing for creative engagement in practice: As AI-based tools become more integral to creative pursuits, there is a need to evaluate such technologies beyond just their relevance, efficiency, or error-free performance [17, 53]. There is a need to understand how the properties emerging from such data-driven systems, such as unpredictability in AI-generated responses or ambiguity in the emergent properties of the AI-generated representational spaces, can add value or benefit or be leveraged by creative practitioners for their work.

In summary, focusing on the technical challenges outlined in this thesis from a human-centered approach, this research aims to design and evaluate generative algorithms for creative support tools for sound design. Such tools should:

- Enable exploration using user-defined semantic attributes to steer or guide a generative model trained on large, unlabeled environmental sound datasets.
- Facilitate steerable ways to support creative tasks such as morphing sounds.
- Be subjectively evaluated through listening tests by listeners of varied experiences for descriptive qualities such as realism or sound progression.
- Be studied for their creative engagement in the creative professional practice of sound design.

These requirements form the foundational aims of this thesis.

1.2 Thesis Aims

My overall research aims are to design interactive generative algorithms and CSTs to empower sound designers with additional tools in their creative work. I envision CSTs with interfaces that allow multi-dimensional semantic sound space exploration, enabling designers to create and discover new sound effects. I aim to design this interactivity with

"low thresholds, high ceilings, and wide walls" [49], i.e., tools that are easy for novices to use yet provide ambitious functionality for expert use.

Following the requirements outlined in the previous section, I now formulate the following research questions (RQs) for this thesis:

- **RQ1** How can we perform exploration using generative audio models trained on unlabeled data to generate environmental sounds using user-defined semantic attributes?

- **RQ2** How can we build steerable generative audio models that support creative sound design tasks such as audio morphing?

- **RQ3** How can we perceptually evaluate audio generated using generative audio models for their descriptive semantic qualities using non-experts on crowdsourced platforms?

- **RQ4** How can steerable generative audio models assist professional sound designers in their creative practice?

1.3 Thesis Structure

This thesis has seven chapters organized as shown in Figure 1.1.

Chapter 1 introduces the aims and contributions of this thesis.

Chapter 2 outlines the theory of sound design, the human-centered AI framework, foundational audio concepts, and the technical background of various generative algorithms used in this thesis.

Part I: Designing interactive generative audio models. In the first part, I discuss the design and implementation of the steerable deep learning algorithms for sound design.

Chapter 3: User-Defined Semantic Attribute Guidance from Unlabeled Training Data addresses **RQ1**. This chapter addresses the human-centered aim of encouraging exploration and steering of the AI-based tool using user-defined controls. I introduce a novel algorithm that perceptually guides environmental sound generation using user-defined semantic controls. This framework operates on a generative model trained on

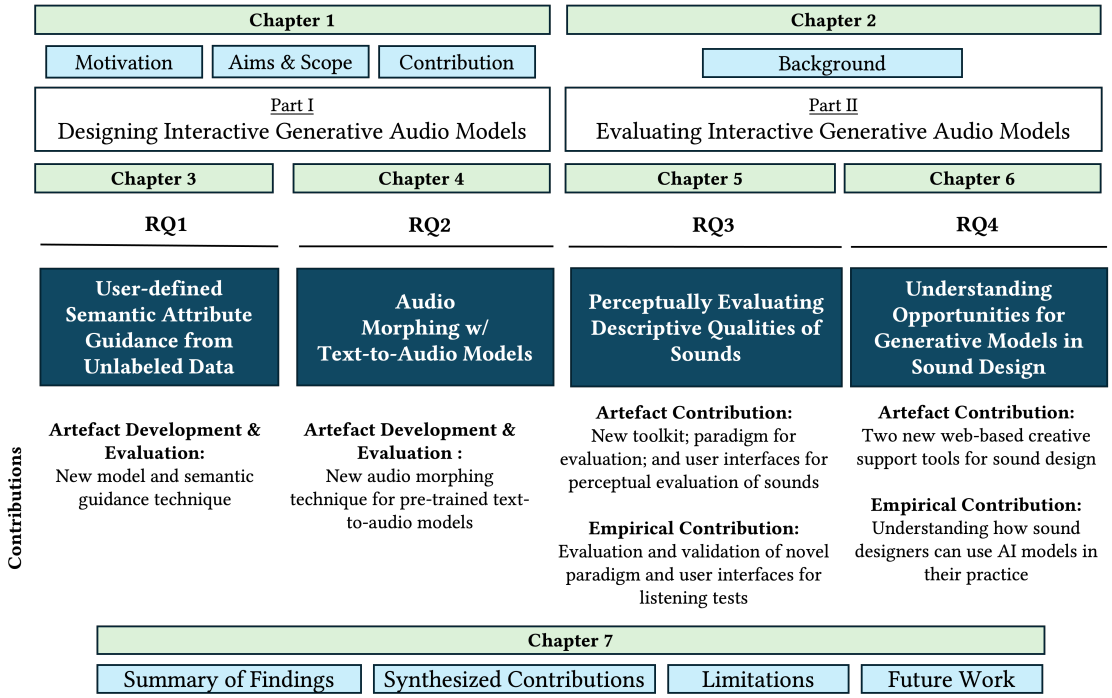


FIGURE 1.1: Overview of thesis structure

unlabeled sounds. In this method, users can create synthetic sounds to communicate their creative goal and *"query"* or *"search"* the generative model to explore and discover novel sound samples. Further, using synthetic sounds, users can define semantic guidance controls to perform semantic edits to the generated sounds. The method is evaluated using objective metrics and perceptual listening tests to demonstrate its effectiveness in providing semantic guidance while training on unlabeled sounds.

Chapter 4: Audio Morphing with Text-to-Audio Models addresses **RQ2**. This chapter addresses the aim of steering AI-based tools using granular, human-understandable fader-like controls to support sound-design-specific creative tasks such as morphing. I introduce a novel algorithm to granularly morph the semantics of two sounds generated by pre-trained text-to-audio (TTA) models. Using this method, sound designers can smoothly morph sounds generated by disparate text prompts and increase or decrease the emphasis on semantic word descriptors while morphing. This technique allows sound designers to explore the semantic sound space generated by TTA models in novel ways. This method can operate on any pre-trained TTA model without requiring extra training procedures or fine-tuning. This method is evaluated using objective metrics and perceptual listening tests to demonstrate its effectiveness in generating hybrid sounds *"in-between"* text prompts.

Part II: Evaluating interactive generative audio models In the second part of

this thesis, I discuss the evaluation aspects of the steerable generative models for sound design.

Chapter 5: Perceptually Evaluating Descriptive Qualities of Sounds addresses **RQ3**. This chapter addresses the research gap of the lack of perceptual methods to evaluate the descriptive qualities of AI-generated sounds. Further, it addresses the human-centered aim of designing such methods for non-experts in audio. I introduce novel visual constructs to perceptually evaluate the temporal descriptive qualities of sounds generated using deep learning models. I demonstrate the effectiveness of such visual constructs by designing listening test interfaces to evaluate sounds in rank ordering and pairwise comparison types of tasks. Using musical instrument sounds and noisy environmental sounds, I conduct experiments to investigate how the quality of responses varies with audio and task complexities. I validate the effectiveness of using such constructs by conducting a study on a crowdsourced platform and verifying their effectiveness in improving the overall quality of responses in a listening test.

Chapter 6: Understanding opportunities for generative models in sound design addresses **RQ4**. This chapter aims to understand how AI-based tools can benefit creative engagement in sound design practice. This chapter describes a study conducted with professional sound designers to understand the challenges and opportunities of using generative models for their creative work. We designed two interactive generative AI models as CSTs and invited professional sound design practitioners to apply the CSTs in their creative practice. Through this study, we develop a novel understanding of how such models can support creative exploration for sound design and provide recommendations for future researchers designing CSTs using generative models.

Chapter 7 summarizes the findings in this work, discusses its limitations, and outlines future directions for designing CSTs for sound design using generative models.

1.4 Contributions

The research contributions from this thesis to the field of human-AI interaction, based on research contribution categories outlined in Wobbrock et al. [54] are—

1.4.1 Empirical Contributions

- This thesis empirically demonstrates that generative models trained on unlabeled sounds can support creative exploration in a user-defined way. By leveraging the

emergent properties of the latent space of the model, we can controllably generate sounds based on user-defined semantic attributes.

- This research shows that existing pre-trained TTA models can be used for novel tasks such as morphing and be steered to generate perceptually plausible morphs between two or more sounds interactively in a fine-grained way.
- This work demonstrates that visual metaphors designed to articulate audio quality constructs effectively improve the quality of responses in perceptual listening tests conducted using non-experts on crowdsourced platforms.
- This work develops a novel understanding of generative models supporting creative exploration for expert practitioners in sound design. Further, design recommendations for future AI-based tool designers developing CSTs at the intersection of human-AI interaction and sound design are offered.

1.4.2 Artefact Contributions

- A novel algorithm "Example-based Framework" (or EBF) for perceptually guided sound effect generation for environmental sounds. This method can perform perceptually relevant semantic edits on generated sounds in real-time.
- A novel algorithm "MorphFader" that morphs two or more sounds generated using different text prompts. The method can emphasize word descriptors while morphing sounds semantically. I developed interfaces over this algorithm to demonstrate its effectiveness in real-time.
- A novel paradigm to evaluate temporal descriptive qualities of sounds using visual metaphors. Reusable interfaces to conduct perceptual listening tests using visual metaphors are designed and developed as contributions to the research community.
- A frontend framework "Crowd-Eval-Audio" to conduct perceptual listening tests on crowdsourced platforms. This framework needs minimal infrastructure (no database installation required, etc.). It can be extended to conduct listening test experiments using any experimental design (such as repeated trials or Latin squares) on crowdsourced platforms.
- Two web-based interfaces as CSTs for sound design. Each interface wraps around two steering techniques developed for generative algorithms. Using these interfaces, sound designers can steer the generative models using either synthetic sounds or perform sound edits directly in the learned latent space of the generative models.

1.5 Publications

The contents of this thesis have appeared in the following top-tier peer-reviewed publications (or are currently under review at a conference or journal venue as indicated).

- **Kamath, P.**, Gupta, C., Nanayakkara, S. (2024). MorphFader: Enabling Fine-grained Semantic Control for Text-to-Audio Morphing through Fader-like Interactions. *(Under Review).*
- **Kamath, P.**, Gupta, C., Wyse, L., & Nanayakkara, S. (2024). Example-Based Framework for Perceptually Guided Audio Texture Generation. In IEEE/ACM Transactions on Audio, Speech, and Language Processing (Vol. 32, pp. 2555–2565). Institute of Electrical and Electronics Engineers (IEEE). `https://doi.org/10.1109/taslp.2024.3393741`
- **Kamath, P.**, Morreale, F., Bagaskara, P. L., Wei, Y., & Nanayakkara, S. (2024). Sound Designer-Generative AI Interactions: Towards Designing Creative Support Tools for Professional Sound Designers. In Proceedings of the CHI Conference on Human Factors in Computing Systems. CHI '24: CHI Conference on Human Factors in Computing Systems. ACM. `https://doi.org/10.1145/3613904.3642040`
- **Kamath, P.**, Li, Z., Gupta, C., Jaidka, K., Nanayakkara, S., & Wyse, L. (2023). Evaluating Descriptive Quality of AI-Generated Audio Using Image-Schemas. In Proceedings of the 28th International Conference on Intelligent User Interfaces. IUI '23: 28th International Conference on Intelligent User Interfaces. ACM. `https://doi.org/10.1145/3581641.3584083`

Although I am the primary first author of the abovementioned publications discussed in this thesis, these contributions would not have been possible without the collaboration with my co-authors. Therefore, in the remainder of my thesis, I use the first person plural "we" in the subsequent chapters to reflect their contributions.

The following is a list of publications and technical reports I have co-authored that comprise important elements of this thesis, to which I have made substantial contributions and which have paved the way for the more independent work I have published listed above. I briefly discuss these publications in Chapter 2 technical background.

- **Kamath, P.**, Islam, T., Gupta, C., Wyse, L., & Nanayakkara, S. (2023). DCASE Task-7: StyleGAN2-based Foley Sound Synthesis. DCASE Foley Sound Synthesis

Challenge 2023. Online. Technical Report. https://dcase.community/documents/challenge2023/technical_reports/DCASE2023_Kamath_6_t7.pdf
Awarded 3rd place.

- Gupta, C.*, **Kamath, P.***, Wei, Y., Li, Z., Nanayakkara, S., & Wyse, L.* (2023). Towards Controllable Audio Texture Morphing. In ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. https://doi.org/10.1109/icassp49357.2023.10096328.
(* denotes equal contribution).

- Gupta, C. Wei, Y., Gong, Z., **Kamath, P.**, Li, Z., & Wyse, L. (2022). Parameter Sensitivity of Deep-Feature based Evaluation Metrics for Audio Textures. In 23rd International Society for Music Information Retrieval Conference (ISMIR). ISMIR 2022. https://archives.ismir.net/ismir2022/paper/000055.pdf.

- Wyse, L., **Kamath, P.**, & Gupta, C. (2022). Sound Model Factory: An Integrated System Architecture for Generative Audio Modelling. In Artificial Intelligence in Music, Sound, Art and Design (pp. 308–322). Springer International Publishing. https://doi.org/10.1007/978-3-031-03789-4_20.

- Gupta, C., **Kamath, P.**, Wyse, L. (2021). Signal Representations for Synthesizing Audio Textures with Generative Adversarial Networks. In Simone Spagnol Davide Andrea Mauro and Andrea Valle, editors, Proceedings of the 18th Sound and Music Computing Conference (pp 159 - 166). Sound and Music Computing Network, Axeasas/SMC Network, 2021. https://doi.org/10.5281/zenodo.5113511

1.6 Environmental Impact & Data Attribution

Environmental impact from CO_2 Emissions Related to Experiments: In this thesis, all models were trained using my personal RTX 2080 Ti (TDP of 250W) GPU-enabled Ubuntu desktop computer that I self-assembled in 2019. I also used an RTX 3090 (TDP of 350W) GPU-enabled Ubuntu desktop computer located in the lab at NUS. Both machines have a carbon efficiency of 0.432 kg CO_2 eq/kWh. Over the last 4 years of my Ph.D. candidature, an approximate total of 608 hours of model training/computation was performed on RTX 2080 Ti hardware, and 528 hours of training/computation was performed on RTX 3090 hardware. Total emissions are estimated to be 145.49 kg CO_2 eq. Which is equivalent to 72.9 kgs of coal burned. Estimations were conducted using

the "MachineLearning Impact calculator"[1] presented in [55]. One of my aims in this research has been to develop algorithms for steering or inducing control over pre-trained models. As in Chapter 4, instead of training new models, I aim to sustainably reuse large foundational models trained by other research organizations and develop steering algorithms based on the emergent properties of the AI model to leverage them as CSTs for sound design.

Data and Model Attribution: In this research, I have used datasets that are freely available under the Creative Commons Licensing agreement for research purposes. I designed and implemented two generative algorithms for sound design in this research. The first algorithm (in Chapter 3) is trained on an audiovisual dataset recorded for research purposes by researchers at the Massachusetts Institute of Technology (MIT), United States of America (USA), released in 2016. This dataset is available under the Creative Commons attributed Non-Commercial Licence[2]. I trained this algorithm on another dataset of water-filling sounds recorded by researchers at the 'Arts & Creativity Lab' at the National University of Singapore (NUS), led by Dr. Lonce Wyse. The second algorithm (in Chapter 4) uses a pre-trained text-to-audio (TTA) model developed by researchers at the University of Surrey, United Kingdom (UK), released in 2023. This model was downloaded from the research group's HuggingFace[3] repository and is made available under the Creative Commons Non-Commercial Share-Alike Licence[4]. This model was trained on large audio datasets that conform to the UK copyright exception for data for academic research. More details on the datasets can be found in the respective chapters in this thesis.

[1] `https://mlco2.github.io/impact#compute`
[2] https://creativecommons.org/licenses/by/4.0/legalcode
[3] https://huggingface.co/
[4] https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode

Chapter 2

Background & Related Work

Chapter Synopsis

This chapter discusses the background and technical foundations for the work presented in this thesis. First, we provide a background on the sound design practice and the human-centered design approach of this thesis. Next, we provide system-level technical foundations for the work presented in this thesis. We provide some background on the types of audio and their representations used in this research. Finally, we provide high-level architectural details on the deep neural networks used in implementing the artifacts resulting from this thesis.

Research on deep neural networks, specifically generative models, has grown significantly in recent years. This chapter's technical details provide a brief overview of the generative models used in this thesis without aiming to comprehensively survey each architecture discussed.

2.1 Sound Design

Sound design is the creative use of sounds to produce engaging cinematic experiences for various consumer media [2, 31]. It is a multi-faceted practice that is both highly technical and artistic in nature and involves creating 'new' sounds. Susini et al., [4] define 'new' sounds as those that cannot be found in existing sound databases or recorded sounds that cannot be used in a given context without being manipulated or modified. Sound design is the deliberate use of such sounds to create immersive experiences in music

composition or other media. Typically, sound designers focus on working out the sonic details required to enrich or complement the visual information presented in films and games [3]. They also focus on communicating additional non-verbal information through interactions in games or product design [4].

Previously, in the "Art of Thought" [56], Graham Wallas proposed a generalized creativity model that consisted of 4 phases: preparation, incubation, illumination, and verification. Similarly, for the specific purpose of sound design, Susini et al. [4] proposed a model which involved three discrete successive stages: *Analysis* assisted by *Exploration*, *Creation*, and *Validation* [4, 57] with the last two being set in an iterative loop until the sound converges towards an optimized solution [58] or the creative goal of the sound designer. The *Analysis* stage is a research-focused phase, where designers are involved in understanding the perceptual requirements of the project using their own knowledge and background in psychoacoustics and sound cognition. It also involves the purposeful exploration of a large inventory of existing sounds and field recording (recording outside of the studio) of new sounds. In the *Creation* stage, sound designers manipulate the sounds or synthetically create new sounds in line with the specifications from the *Analysis* stage. This stage may layer together various sound samples to create montages as a final artifact. The final stage consists of *Validating* the sound specimens created either informally based on the designer's intuition or more formally using listening tests (especially while designing sounds for products [57]).

Throughout the sound design process, designers need to employ different modes of working - as a researcher during *Analysis* and exploration phase, as a programmer or employing their tools-based expertise during the *Creation* phase, and as a qualitative researcher or tester during the *Validation* phase. Through these phases, they also employ different listening techniques such as *causal*, *semantic*, or *reduced* listening [3]. Such listening techniques help designers to associate sounds to sources (*causal*), associate information or meaning (*semantic*) to them, or focus on fine-grained timbre-specific details of the sound (*reduced*) when listening. These modes of working help them develop "*Sonic Vocabularies*" [58] and "*Sound Palettes*" for various current and future projects.

Given this background on the sound design workflow, the primary aim of this thesis is to design AI-based CSTs that can integrate well into a sound designer's creative process. I aim to study areas within the sound design process where such AI-based CSTs can be integrated and the challenges of adopting them in others.

2.2 Human-Centered AI for Creativity Support

AI systems are trained to reveal patterns within the training data. Their effectiveness, however, may be limited by the strategies used to collect the training data, the system's ability to respond to extreme cases, etc. The Human-Centered AI (HCAI) framework [49], by Ben Shneiderman, is a design philosophy that considers the AI system's limitations and foregrounds the design of such systems in empathy for its human users. It bases the design practice on specific guidelines to ensure human understanding and intuition about the AI system for building trustworthy and reliable applications. While most guidelines cater towards designing critical decision-making systems, such as those that assist in healthcare and other recommender systems, in this thesis, I extend the four principles outlined below to apply them to designing AI-based creative support tools.

The design community has long relied on Shneiderman's 8 golden rules for designing interactivity on user interfaces [52]. These rules include guidelines on preventing errors, keeping users in control (instead of the tool being in control), and reducing short-term memory load to build comprehensible and controllable systems. While such rules still apply to designing AI systems, the HCAI framework builds on them to include important guidelines such as *"steerability by way of interactive controls"*. Furthermore, research on creativity support outlines guidelines to design systems that *"support exploration"* and enable use by both *"novices and experts"*. Finally, researchers studying creativity support offered by new media technologies [53] ground evaluation of such tools for their *"creative engagement in practice"* beyond a typical work-oriented view of relevance or usefulness.

AI technologies can empower new media artists with novel forms of expression by providing powerful Creative Support Tools (CSTs) to support their creative work. The research in HCAI and digital tools for AI-based creativity support is novel and growing. This thesis builds upon and contributes to the HCAI literature and design guidelines, focusing on designing and evaluating controllable AI models for the specific purpose of sound design.

2.2.1 Support For Exploration

Digital tools offering creativity support are largely based on the hypotheses about how creativity happens in the human mind [59]. In her seminal work to understand human creativity and outline its definition, Margaret Boden first defines a conceptual space in people's minds that can be explored and transformed to give rise to novel creative concepts [59]. These conceptual spaces are created based on prior structures of thoughts

and concepts learned by individuals throughout their lifespans. Creativity by exploration of this conceptual space enables us to develop novel ideas and explore diverse creative possibilities. Representational spaces generated by trained AI systems are analogous to these conceptual spaces that can be explored to discover novel artifacts[1].

In creativity research, performing exploratory searches to find relevant ideas from previous work and combining them in novel ways is well known [17, 59]. For instance, creative visual artists have been known to explore such conceptual spaces for ideation in the domains of fashion [60] and while creating new images [24, 61]. Researchers regularly use metaphors and visual sketches for ideation for music as an alternative to browsing and searching for existing sounds [62]. For audio AI-based tools, recently, Scurto et al. [63] developed tools based on reinforcement learning algorithms and studied user exploration behaviors for the generated high-dimensional representational spaces.

In [64], HCAI researchers outline the principle of *"capturing intent, rather than input"* to build interactive supertools. They propose designing AI-based systems that capture explicit user intent and implicitly provide additional steering or guidance to the AI based on the user's interactions with the tool. John Maeda, a pioneer of mathematical and computational arts, argues for capturing this explicit and implicit creative intent by emphasizing the term "software sketches". An initially poorly defined piece of code or software can be iterated to help move the process of exploration and idea generation forward [65]. Such a "sketching" or "intent capturing" process requires fluid engagement with the design material at hand for rapid exploration and feedback.

In line with these ideas of rapid exploration by sketching the user's creative intent, this thesis designs and evaluates generative models that allow real-time exploration of the AI's representational space to generate and edit novel sounds. We provide novel affordances to explore the representational space based on their user's definition and sonification of semantics in the sound they want to create.

2.2.2 Steering By Way of Interactive Controls

Steerable CSTs for sound design allow designers to controllably generate sounds with semantically relevant attributes and perform continuous fine-grained semantic attribute

[1]Although we use the conceptual space analogy to understand AI's representational space, it should be noted that Boden uses computer programs, and in the latest revision of the book, she uses AI models, to create an analogy of conceptual spaces in the human mind. In this section, I reverse this analogy to develop an intuition about exploring the AI's representational space for this thesis.

edits to the generated sound. The current landscape of generative algorithms includes CSTs [17, 50] that are either fully autonomous or support co-creation as Mixed-Initiative Creative Interfaces (MICIs) [66, 67]. Fully autonomous AI-based CSTs usually accept simple user inputs to generate artifacts with less granular control. For instance, consider the example of Jukebox [68], which generates full music compositions based on inputs such as artist, genre, and lyrics but does not provide control for improvisations. AI-based co-creation CSTs, on the other hand, feedback and improvise on user creation in the domain of visual arts [25, 26], in writing [27, 28], in UX design and engineering [69–72], and new musical interface design [73]. Researchers have also worked at the intersection of explainable AI (XAI) and arts to explore novel ways to steer AI-based CSTs for creative endeavors [74, 75]. In the field of audio, machine learning models have long been used for creating music [76], from established tools such as Wekinator [77] to the more recent AI music performance art [78]. While steerability for CSTs generating music would mean improvising on a particular piece of music, for the pursuit of sound design, steerability would refer to granularly editing the pitch of a dog bark or changing the surface material for footsteps.

Generative models for sound generation previously relied on specialized labels designed by the AI model's developers [13, 79, 80]. Although such models enable building steerable interfaces for CSTs [15, 16, 48, 81, 82], they focus solely on music generation and less on the needs for sound design. Recently, diffusion-based text-to-audio (TTA) models have democratized how we generate sounds using AI models. Sound designers of all experience levels can use natural language to leverage AI models in their creative work. While such TTA models train on large datasets and can generate diverse sounds, discrete text prompt interfaces do not provide avenues to granularly edit or morph the generated sounds. For instance, text-based controls provide little ability to continuously and granularly control the ability to morph or edit the surface material for footsteps from metallic thuds to a snowy crunch.

In this thesis, we address this gap and develop ways to granularly steer AI algorithms. We provide avenues to semantically edit and morph the sound generated by the generative algorithms with an aim to allow sound designers to explore the AI model's conceptual representational space better and in a fine-grained way.

2.2.3 Design For Both Novices and Experts

A prominent guideline for designing creative supertools is the principle of designing interfaces for users of all experience levels [17, 52]. Typically, perceptual listening tests

for evaluating audio generated by AI algorithms are conducted using expert listeners in a controlled lab-based listening environment. In-person listening tests require a considerable amount of the researcher's time and effort and are expensive to set up. Thus, there is an increasing push within the audio deep learning community to move towards crowdsourced platforms such as Amazon Mechanical Turk (AMT) to conduct these tests. AI-generated audio is typically evaluated on quality concepts for sound progression, such as the quality of a morph or how two sounds are interpolated with each other. While such concepts can be easily understood by audio experts, non-experts on crowdsourced platforms may find such technical jargon difficult to understand.

An ideal audio quality description in a listening test should explicate the complexity observed in the sound space in a human-understandable way. Outlining such complex qualities verbosely using language makes for lengthy task instructions, which reduces participant interest in such tasks [83] and affects the overall quality of responses. Furthermore, the design of a typical listening test interface involves listening to two or more sounds in comparison to each other or with respect to a reference. As the number of sounds increases (for instance, a MUSHRA test [84] sometimes involves listening and comparing up to 12 sounds with each other), the demands on the listener's audio memory also increase, thus increasing the task's complexity. In contrast, for example, image annotation or evaluation tasks often require only a simple 'glance-and-click' action [29].

Recent human computation research on crowdsourcing shows that as task complexity increases, the quality of responses decreases, and participants more frequently abandon such tasks or submit poor quality responses [83, 85]. This thesis investigates ways to design intuitive interfaces for conducting perceptual listening tests that minimize audio task complexity. It expands on existing human-computation and crowdsourcing research [83, 85–87] and develops and evaluates methods for audio listening tests that assist novice listeners in understanding descriptive audio qualities under evaluation and in turn, help researchers collate better and more meaningful responses from such listening tests.

2.2.4 Designing For Creative Engagement In Practice

Most empirical studies in human-computer interaction (HCI) and HCAI usually focus on the ability of the interfaces to assist users in performing a set of tasks in an efficient and error-free way. For instance, most AI-based co-creation studies in music generation using expert practitioners focus on their ability to efficiently articulate their creative intent using the system [16, 48, 88]. Other studies focus on how visual artists [24] and engineering designers [72] use AI-based algorithms for creative decision-making. These

studies observe the role of the intelligent tool in relation to the designer and measure the effectiveness and performance of the designer in completing the predefined task to assess the applicability of AI-based CST. Instead, Jonas Löwgren suggests that creative work practices could benefit from evaluating CSTs based on *"use-qualities"* of the system. That is, to evaluate CSTs based on qualities that usually do not arise from a work-oriented view. Some of such use-qualities include pliability and ambiguity. [17, 53, 89].

Löwgren defines pliability as a property of a responsive design material that can be used to create new artifacts in a highly involved, rapid, iterative, and exploratory creative process. A user "makes a move", the system generates an outcome, and the user perceives the outcome and proceeds to further make changes in a tight loop. He further defines ambiguity as another positive use-quality to be considered when designing and evaluating interactivity. Generally, ambiguity in interactions is considered to be detrimental in HCI as it stands in opposition to efficiency and transparency. Previously, researchers argued that unpredictability and non-determinism could be detrimental to the user experience of an AI system [90]. However, Gaver et al. showed that for digital art tools, ambiguity can be effectively used to develop close personal engagement and enable reflection while creating art [91]. While ambiguity may make easy interpretation of the system impossible, it provokes discussions and forces users to participate in meaning-making [89].

In their recent work, Caramiaux et al. [23] showed that artists usually embrace emergent and unpredictable behavior in generative AI systems rather than consider it a limitation while creating visual art. In this thesis, we take inspiration from prior work to explore this aspect of pliability and meaning-making with ambiguity in design with AI-based CSTs and explore the role of unpredictability and non-determinism in generative audio AI output for the creative work practice of sound design. Furthermore, we explore the notion of ambiguity in conjunction with the sound design workflows and processes outlined previously in section 2.1.

2.3 Technical Background

2.3.1 Audio data classes and representations

2.3.1.1 Audio Data Classes

While generatively modeling sounds using deep neural networks, researchers generally distinguish between the signal representations of speech, music, and environmental sounds.

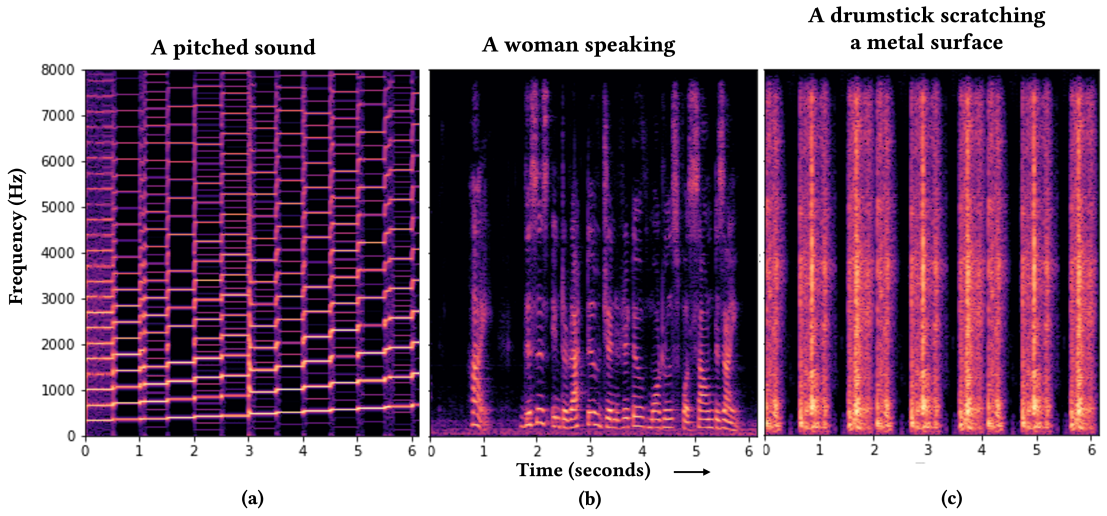


FIGURE 2.1: 2D Spectrogram representations for different sound types.

This is because speech and music usually contain harmonic frequencies, the knowledge of which can be useful while modeling them. On the other hand, environmental sounds can be noisy and contain multiple inharmonic components, making modeling them difficult compared to their harmonic counterparts. Thus, the current research landscape treats the three classes as separate generative modeling tasks.

Figure 2.1 visually demonstrates the structural differences between representations of the different types of sounds. Figure 2.1(a) shows a ***"pitched sound"*** made by a brass instrument rising in pitch in steps, and (b) shows a speech signal of a woman saying "Interactive Generative Audio Models for Sound Design." Such sounds usually have structured representations and are governed by the presence of a fundamental frequency (or f0) and its multiples (or its harmonics) at each time step. On the other hand, environmental sounds have unstructured representations. For instance, the example shown in Figure 2.1(c) is an impact sound made by a drumstick hitting a metal surface. Such sounds have sharp attack transients (onset or the beginning of the sound event) with multiple inharmonic and noisy frequency components, making them harder to model than pitched sounds.

In this thesis, we focus on modeling environmental sounds. We also focus on a subclass of environmental sounds called ***"audio textures"***. Audio textures are sounds generated by the super-position of multiple similar acoustic events [36]. Textures can also be considered as sounds whose informational content asymptotes after a certain amount of time [92]. Audio textures can be a series of discrete sound events, such as the sound of footsteps or the sound of a wooden drumstick repeatedly hitting a hard metal surface. They can also be noisy and continuously varying, such as the sound of water filling a

container. Typically, parametric synthesizers [93–96] or physics-based models [97–99] can be used to generate synthetic textures. Although synthesizers provide fine-grained control during generation, the generated sounds usually sound synthetic and may lack the timbres associated with real-world sounds.

2.3.1.2 Audio Data Representations

Audio can be represented in many ways. When recorded using a microphone, the air pressure changes in the sound wave are converted to an analog electrical signal [100]. This signal is *"sampled"* or digitized to an array of floating point numbers representing the amplitude of the sound at a given point in time. This one-dimensional (1D) array of floating point representations is called a raw audio waveform. The quality of the digitized sound is governed by properties such as sampling rate. This rate is the number of samples digitized from the analog signal per second, measured in Hertz (or Hz). Typically, most deep learning algorithms use sampling rates of 16kHz (i.e., the analog sound is converted into $16,000$ samples per second of the sound wave), 22.05kHz, or 44.1kHz. Higher sampling rates can faithfully preserve and digitize the high-frequency components from the analog signal in a less lossy fashion. Larger sampling rates result in larger 1D arrays and require higher computational processing and larger GPU-enabled computers while training.

Many higher-level representations can be derived based on this low-level raw audio. Time-frequency (TF) representations are often derived from raw audio when modeling sounds using deep neural networks. TF representations are two-dimensional (2D) matrices representing frequency on the y-axis and time on the x-axis. They can be used to visually demonstrate the *spectromorphology* [101] of the sound or how its frequencies change over time. One commonly used TF representation is the 2D spectrogram generated by the Short-Time Fourier Transform (STFT) [93]. STFT of a sound sample is calculated by computing a Discrete Fourier Transform (DFT) [93] across a small moving time window. A 2D STFT spectrogram can be considered a stack of individual DFTs performed for small time windows across the entire length of the sound sample. Note that each TF bin of the STFT is a complex number, where the real part is the magnitude of the spectrogram, and the imaginary part is the signal's phase within that window. Such complex valued spectrograms can be edited and "inverted" back into the 1D raw audio domain, which can then be converted to analog using digital-to-analog converted (DAC) to be played over headphones or speakers. Often, we use the magnitude of the spectrogram without the complex-valued phase for modeling using deep neural networks.

The choice of audio representation depends on the modeling task [102] and further governs the deep learning network architecture. This choice also depends on the type of sound being modeled - whether the sound is pitched or inharmonic with many noisy components. Typically, when generatively modeling raw audio, autoregressive architectures such as Recurrent Neural Networks (RNNs) [103, 104] are used. Network architectures such as Generative Adversarial Networks (GANs) [105] are usually trained on 2D log-magnitude spectrograms, which are computed by taking the logarithm of the absolute value of the magnitude spectrogram. Human hearing is logarithmic with regards to amplitude and frequency [106]. Thus, we typically use log-magnitude spectrograms or use mel spectrograms by transforming the frequency-axis of the spectrogram to mel-scale (or quasi-logarithmic scale) [100, 107]. Usually, only magnitude or log-magnitude spectrograms are used to train GANs. Some architectures, such as in [108], train on both real and imaginary parts (phase) of the STFT. Other architectures use log-magnitude spectrograms in conjunction with Instantaneous Frequency (IF or unwrapped phase) representations [13] during training and generating musical sounds with great success.

A spectrogram's magnitude and phase are both needed to invert a sound to the raw audio domain during inference. Generative models are usually trained to estimate the magnitude of the spectrogram. The phase (or the imaginary component of the STFT) is usually estimated from the generated magnitude spectrogram. Currently, there are three popular methods to estimate phase from magnitude. First, iterative optimization-based algorithms such as Griffim-Lim can be used. And second, using a pre-trained HiFi-GAN [109] vocoder which converts spectrogram to raw audio. And third, using IF representations and reconstruction [13] techniques. In our previous work [110], we used a fourth technique, namely, Phase Gradient Heap Integration (PGHI) [111], to reconstruct the phase for environmental sounds. PGHI uses the mathematical relationship between the magnitude of Gaussian windowed STFT and the phase derivatives in time and frequency of the Fourier transform to reconstruct the phase using the magnitude spectrogram. We showed that by using PGHI, we can reconstruct clear and sharp transients (attack and decay of the sound event) better than IF representation and reconstruction methods. Further, PGHI is a closed-form estimation method, and unlike the iterative optimization-based Griffin-Lim algorithm, it is faster to compute phase in real-time. This real-time nature is immensely useful for real-time applications of sound design.

With this background, in this thesis, we develop algorithms using environmental audio data sampled at 16kHz and leave extrapolating our algorithms to higher sampling rates for future work. Further, we utilize Gaussian windowed log-magnitude spectrograms when using GANs to train environmental sounds. Due to its ability to reconstruct sharp

transients in the resulting sounds, we use PGHI to estimate the phase of the sound during generation.

2.3.2 Generative Algorithms for Audio Generation

Currently, a multitude of generative model architectures exists for generatively modeling environmental sounds. Each architecture solves a certain set of problems and has its limitations. For instance, autoregressive architectures, or models that predict future values based on past values, such as Recurrent Neural Networks [103, 104, 112], WaveNet [33], or Transformers [113, 114], trained on raw audio, can generate sounds of indefinite duration. However, the time taken for their responses is usually large, which makes their adoption in practice difficult. On the other hand, non-autoregressive models, such as those based on Generative Adversarial Networks (GANs) [105], are responsive but generate samples of a pre-defined duration, usually a few seconds long. Such models are expressive in their ability to generate novel sounds or morphs [115] compared to autoregressive architectures [116]. Recently, researchers have successfully explored another class of fixed-duration sound models using Denoising Diffusion Probabilistic Model architectures (DDPM, or simply diffusion-based models) [117, 118] that generate better quality and more diverse sounds than GANs.

Autoregressive models trained on raw audio samples need long inference time and have large training memory requirements. Generating a single raw audio sample requires considering the samples that existed before it. For example, to generate one second of sound sampled at 16kHz, the last sample needs to maintain the context of the previous 15,999 samples. Thus, as the audio length increases, the context length increases, quadratically increasing the computational complexity of training such models using architectures such as Transformers [119]. Furthermore, using auto-regressive models, the time taken to generate a few seconds of sound might be in the order of minutes. Low latency and shorter system response times are important for designing and building user-driven CSTs for sound design. Therefore, this thesis uses GANs and diffusion-based TTA models to model environmental sounds and design CSTs.

In the next few sections, we outline the technical architectural details of GAN and diffusion-based TTA models used in this research.

2.3.2.1 Generative Adversarial Networks

Sounds occurring in the real world are considered to belong to a highly dimensional data distribution that is also sparse; that is, not all points in this high-dimensional space result in believable or perceptually plausible sounds. For instance, consider a 2D spectrogram. The probability that a randomly generated spectrogram (randomly selecting frequencies for each time window) will generate a plausible or believable sounding sample is small. Thus, GANs aim to achieve three main goals: (1) Reduce the dimensionality of real-world datasets to create a learned distribution, called *"latent spaces"*, which can be randomly sampled to generate plausible sounds. Although the learned representation has fewer dimensions than the real-world data distribution, it is still high-dimensional for human visualization or cognition. (2) Generate a *"disentangled"* latent space, where semantically different sounds are clustered separately. (3) Allow for smooth interpolation between sounds within the learned latent space, enabling creative exploration and discovery of novel *"in-between"* sounds.

GANs can be trained to generate sounds controllably. Broadly, two approaches exist for training controllable GANs. One approach involves training models using labeled datasets in a supervised way [120, pg. 137]. For such models, generation is controlled using pre-defined labels. This approach is popular with most GANs trained on musical instruments. For such models, sounds are generated by controlling for pitch or instrument type [13]. Another approach is where the models are trained on unlabelled datasets, and controllability is inferred using unsupervised methods [120, pg. 142]. This approach is especially useful for environmental sounds as they can be recorded easily "in-the-wild", but annotating them with labels reliably is difficult. GANs can be trained on large, unlabeled environmental sound datasets to generate an expressive latent space, which can be used to search, generate, and manipulate new sounds. Recently, researchers have been working at the intersection of explainable AI (XAI) and arts to explore novel ways to explore such latent spaces for creative endeavors [74, 75]. In this thesis, we aim to develop algorithms to facilitate the exploration of the latent space of a GAN trained on unlabelled environmental sounds in a human-understandable way.

Figure 2.2 shows a GAN architecture schematic. GANs generate the expressive latent spaces $\mathcal{Z}$ by simultaneously training two competing networks: a generator network called G and a discriminator network D (also known as the critic). The generator function G maps a randomly sampled noise vector from a prior known distribution to the training data space. The discriminator D receives either a sample generated by G or a true data sample and must distinguish between the two [105]. The noise is sampled from a "prior" (the known or assumed noise distribution) $\mathcal{Z}$ and passed through the generator

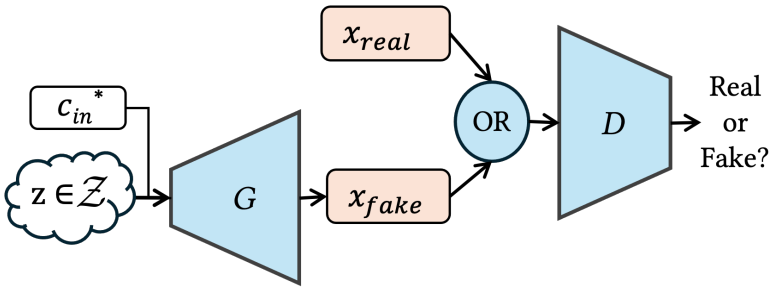


FIGURE 2.2: Schematic of a Generative Adversarial Network

network to generate the fake sample. At the same time, a real sample x_{real} from the training data is sampled. The discriminator D is incentivized to discriminate the real from the fake sample. Similarly, the generator G is incentivized to fool D to generate samples that closely resemble data from the training distribution. In deep learning parlance, the generator G is said to be pitted against the discriminator D in a *"minmax"* game. In this setup, the latent space is structured by G based on feedback from D alone without direct access to the training data [104]. Training G this way results in the generation of high-quality samples resembling real-world data. However, it also leads to entanglement in the latent space [104, 121] $\mathcal{Z}$ and a structure where the individual dimensions do not correspond to semantic features of the underlying data. One way to induce additional structure to this latent space is by conditioning the training on relevant attributes, i.e., train both G and D with extra information regarding the input (e.g., pitch or instrument type when training on musical instrument sounds). This additional conditioning information is represented as c_{in}^* in Figure 2.2.

Formally, say $\mathcal{X}$ is the training data distribution, and z is the prior defined on the noise variable, then $G(z;\theta_g)$ formalizes a trainable generator function that maps noise z to the data space $\mathcal{X}$ with θ_g as trainable parameters for the generator. We also define the discriminator as $D(x;\theta_d)$, where θ_d are the trainable parameters of network D. This network D outputs a single scalar value representing whether the input sample came from the training distribution $\mathcal{X}$ or the generated fake data. The D is trained to maximize the probability of assigning the correct label to samples from real data (as real) and from G (as fake). Further, G is trained to fool the generator by creating samples that closely resemble real data. Please see appendix A.1 for the commonly used loss formulations for GANs.

For audio, researchers have successfully used a type of GAN architecture called Progressive GAN (or PGAN) [122], to generate high-quality musical sounds. To train PGANs, we start with low-resolution spectrograms and then progressively increase the resolution

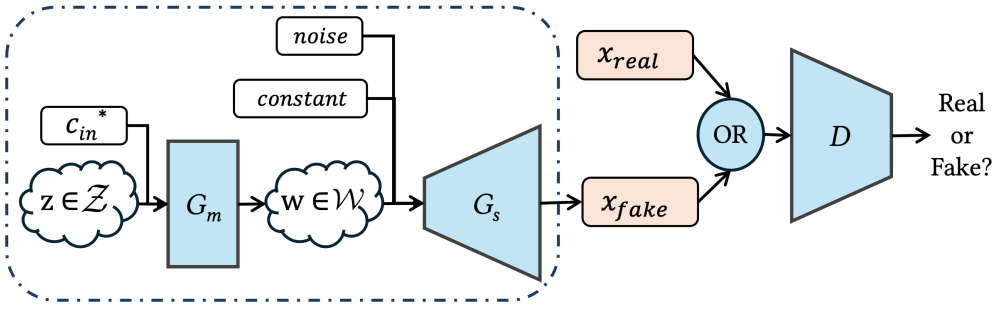


FIGURE 2.3: Schematic of a StyleGAN

of the generated output during training by synchronously adding new layers to the generator and the discriminator network. By modifying the network architecture this way, we can generate high-resolution spectrograms that can reconstruct transients (event onsets and decays) in the sound events [110] faithfully. This training strategy has been useful for conditionally training GANs for music [13, 123] and environmental sounds [110, 116, 124].

In summary, by training PGANs, we can generate a learned representation, also termed as a latent space $\mathcal{Z}$ of the model, which can be randomly and continuously sampled to generate diverse, high-quality audio samples that resemble naturally occurring sounds for qualities such as plausibility or realism. Further, by conditioning the training using labeled data, we can semantically control the generation of sounds. However, there are challenges in the latent space organization of PGAN, namely semantic entanglement, which makes it difficult to induce semantic control when trained on unlabeled sounds.

2.3.2.2 StyleGAN Architectures

Previously, we discussed a GAN architecture where the training procedure generates a latent space $\mathcal{Z}$. This network architecture and training procedure leads to unavoidable entanglement and less control over individual semantics during the generation when trained on unlabelled sounds. To circumvent this problem, Karras et al. [125, 126] propose an architecture called StyleGAN, where they re-designed the generator G architecture as shown in Figure 2.3. The generator of a StyleGAN accepts the noise vector z along with any optional conditioning c_{in} to generate an intermediate latent space $\mathcal{W}$. This architectural change leads to the automatic, unsupervised disentanglement of high-level semantic attributes within the intermediate latent space. It also enables smooth, intuitive, and perceptually linear semantic interpolation operations while generating sounds [125].

Figure 2.3 and 2.4 shows a high-level and detailed schematic of a StyleGAN respectively. The generator is split into a mapping network G_m and a synthesis network G_s. The

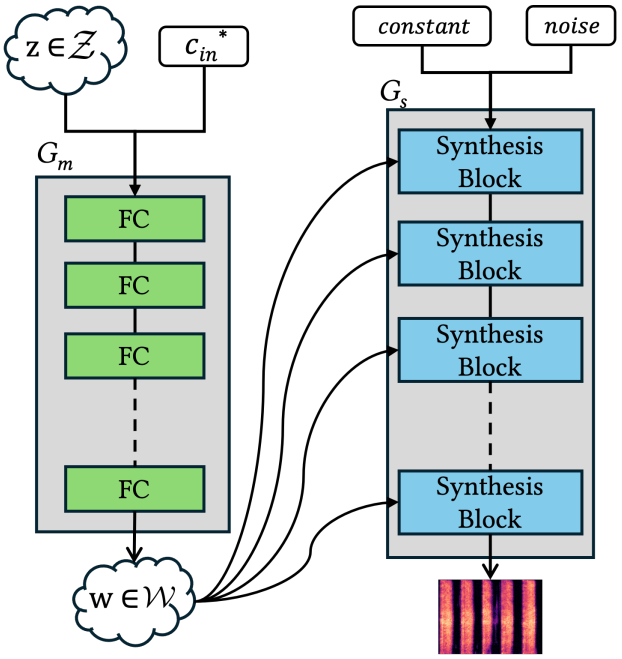


FIGURE 2.4: Schematic of the generator network of a StyleGAN

noise latent z is passed through G_m to generate $\mathcal{W}$. The resulting w vector, $w \in \mathcal{W}$, is the intermediate latent or style space. The G_s also accepts a noise vector responsible for localized, stochastic variations in the generated samples. The mapping network G_m is a sequence of fully connected layers, denoted by *"FC"*. The *"style vector"* from the intermediate latent space generated this way is fed into each layer of the synthesis network G_s. Each layer of this synthesis G_s is termed a *"synthesis block,"* which scales the input to this network with the style vector, followed by convolutions and normalization layers. The synthesis network generates a spectrogram output that can be inverted to raw audio. In this work, although we empirically select the number of FCs and synthesis blocks as hyperparameters during training, we do not directly modify the architectural components of this network. More details regarding both G_s and G_m can be found in [125] and [126].

Although StyleGANs were originally developed for computer vision tasks, we use the architecture to generate audio by training on 2D spectrogram representations. Recently, in the DCASE Foley Sound Synthesis Challenge [127, 128] models trained unconditionally using StyleGAN architectures ranked in the top-3 submissions. The challenge involved generating novel, high-fidelity, and diverse sounds for seven sound classes, such as dog barks, footsteps, motor vehicle sounds, etc. For our submission[2] we trained StyleGANs conditionally and unconditionally, i.e., without using labels. We trained one StyleGAN

[2]Please see submission titled "Kamath_NUS_task7_trackB_2" under track B:`https://dcase.community/challenge2023/task-foley-sound-synthesis-results` which ranked third in the challenge.

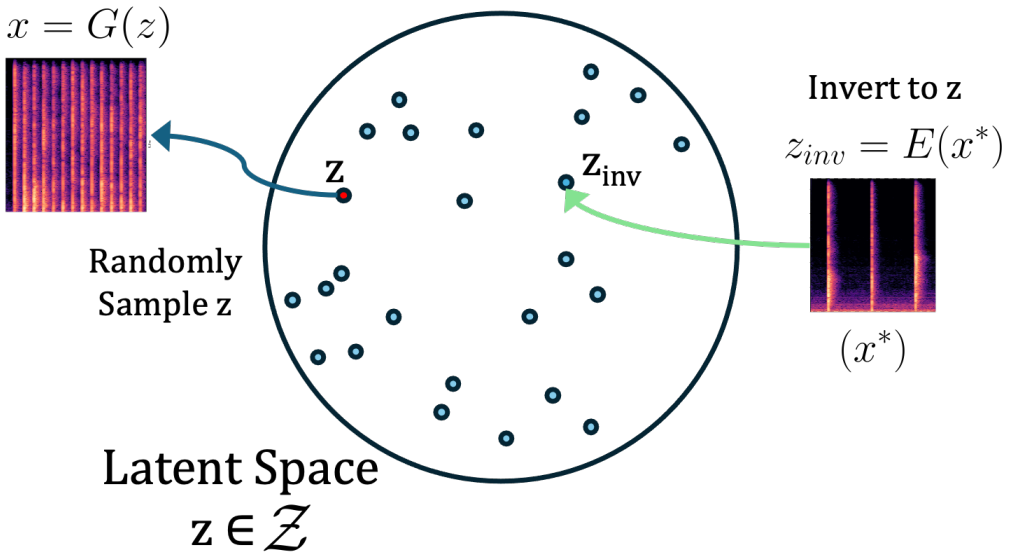


FIGURE 2.5: Schematic of GAN Inversion

network using one-hot class identifying vectors for the conditional system. We trained seven StyleGAN networks for the unconditional system, one model for each class. Although both systems performed better than the baselines provided during the challenge, our unconditional system submission was ranked third (top three selected of 27 submissions) in the challenge. Furthermore, in our experiments, sounds generated by StyleGAN were of better quality than PGAN. This demonstrates the efficacy of using StyleGANs in our work to generate sounds using unlabelled datasets. Please see the technical and other evaluation details for the StyleGAN architecture in our technical report in appendix A.2.

2.3.2.3 GAN Inversion

GAN Inversion, or GAN Encoding, is the process of inverting or encoding an image or audio sample into the latent space of a pre-trained GAN (either PGAN or StyleGAN). This inversion process results in a latent vector z (or, in the case of StyleGAN, the latent vector w), which can be used to reconstruct it using the generator network [129] faithfully. This technique is especially useful for performing edits to the image or sound in the latent space of the GAN. Figure 2.5 shows a schematic for GAN inversion. Say we have a GAN trained on a large dataset of clapping or tapping sounds. Randomly sampling a z vector can generate a fake but plausible-sounding sample using the generator G as $x = G(z)$. Using GAN inversion, we can find the approximate or "nearest" latent vector, which can faithfully reconstruct a real sound x^*, using an Encoder network or function E, such that $z_{inv} = E(x^*)$. Using the inverted latent vector z_{inv}, we can easily perform latent

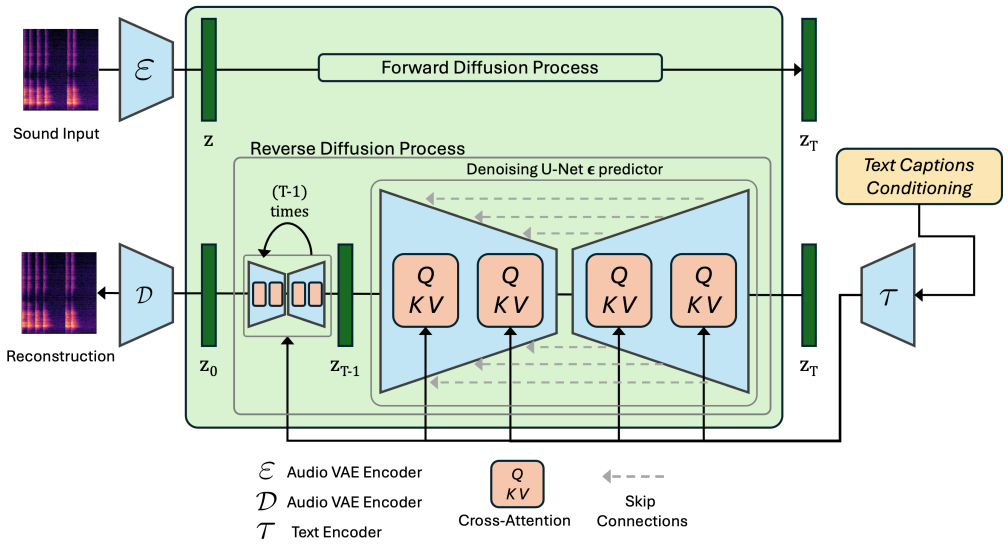


FIGURE 2.6: Schematic of a Latent Diffusion Model conditioned on text captions. Originally from [137], it was recreated with changes for this thesis.

space exploration around the inverted vector and perform semantic edits to the sound in the latent space of the GAN. In this thesis, we primarily use StyleGAN for generation. We thus design an Encoder that inverts real-world samples into the $\mathcal{W}$ space of the StyleGAN, i.e., estimate w vector instead of z as shown in the Figure.

2.3.2.4 Latent Diffusion-based Text-to-Audio Models

Diffusion-based [130] text-to-audio models [117, 131–134] have recently become popular as an alternative to GANs for text-based conditional audio generation. In this paradigm, the model is trained on large audio datasets in conjunction with weakly labeled text captions. Diffusion models are trained in two stages: (1) In the first stage, or the forward diffusion process, the training samples are noised by progressively adding a small amount of Gaussian noise in a series of pre-determined steps known as a noise schedule. (2) In the reverse diffusion process, the noisy sample is denoised by estimating the noise per step of the diffusion process. Typically, a denoising U-Net [135] is used to predict the amount of Gaussian noise to remove from the sample. Diffusion models for audio generally work on noising and denoising spectrogram representations. Such models usually require large computational resources for both training and inference. To circumvent this, researchers recently have used diffusion models to noise and denoise the latent vectors of a pre-trained Variational AutoEncoder (VAE) [136] instead of spectrograms. Such Latent Diffusion Models (or LDMs) are more efficient regarding resource consumption and inference times than spectrogram-based diffusion models.

A schematic of an LDM is shown in Figure 2.6. During training, the sound input is first encoded into the latent space of a pre-trained VAE using ε, resulting in the latent vector z. This latent vector is progressively noised in T steps using a forward diffusion process to generate z_T. Note that the distribution of the final z_T vector resembles that of Gaussian noise. In the reverse diffusion process, the LDM tries to estimate the noise ϵ added during the forward process in each step. This ϵ estimate is progressively denoised from the z_T vector in T steps to obtain the final z_0 vector. The ϵ noise estimate is predicted using a U-Net [135] based network.

The text conditioning used to control generation is injected into the diffusion process using a series of cross-attention layers [113] in the U-Net. At a high level, cross-attention can be considered a function that computes the semantic similarity between a text caption and the to be generated sound. LDMs are trained such that sounds and their associated text captions have high semantic similarity. The U-Net noise predictor is a deep neural network with a series of cross-attention layers. The text prompt is iteratively injected into these layers for every step of the diffusion process.

The final estimated z_0, after T diffusion steps, is decoded using the VAE's decoder network to reconstruct the spectrogram. A randomly initialized z_T and a text-prompt conditioning are input to the LDM during inference. That is, only the reverse diffusion process is executed. The cross-attention layers in the U-Net ensure the predicted noise is adjusted based on the distance in the VAE's latent space and the difference in the text-based semantic conditioning space.

GAN-based architectures are prone to mode collapse in a multi-class setting. Diffusion-based architectures, on the other hand, can generate better-quality and more diverse sounds than GANs [117], especially when using text-based conditioning. However, it is challenging to control semantics described in the text prompt granularly. Further, using LDMs to perform creative tasks such as morphing between two prompted sounds is unexplored. We aim to address these challenges in this thesis. We use existing, foundational pre-trained TTA LDMs such as AudioLDM [132] to build CSTs for sound design.

2.4 Summary

This chapter provided theoretical and technical foundations for this thesis. Figure 2.7 visually summarizes the theoretical and technical concepts discussed in this chapter, along with the chapters and research questions they are associated with.

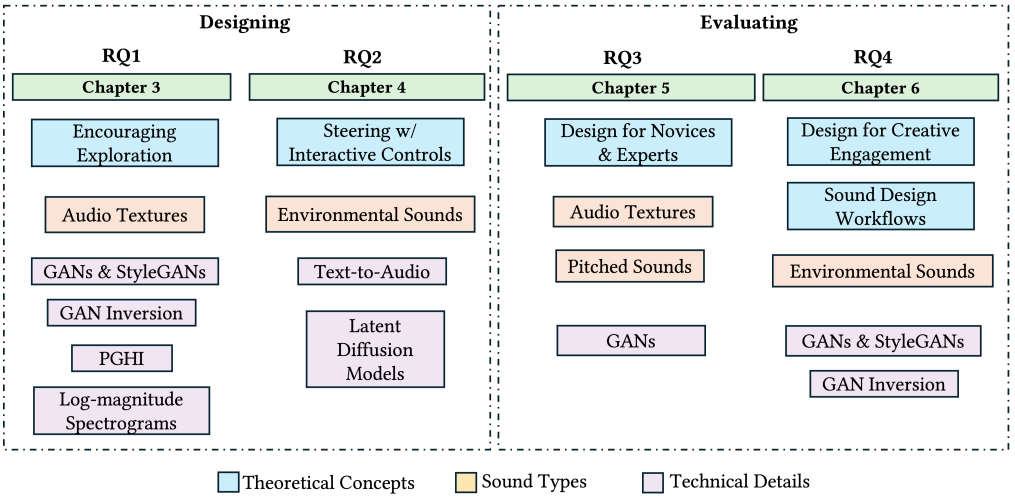


FIGURE 2.7: A conceptual diagram outlining the theoretical and technical concepts applied and discussed in each chapter and its corresponding research question.

First, this chapter provided a theoretical background on sound design workflows, including the stages of creating sounds for various media. It also outlined sound designers' use of sound palettes and different listening techniques. Next, HCAI approaches that this thesis builds upon and contributes towards were outlined. Human-centered approaches grounded in supporting exploration for creativity, steering by interactive controls, designing listening tests for non-expert use, and studying the use-qualities of generative AI models for sound design were discussed.

This chapter additionally provided technical background on the deep learning representations and architectures used in this thesis. Insight into the decisions based on our prior work in Chapter 1, Section 1.5 were provided. Different audio representations used in deep learning were outlined. A background was provided on using Gaussian windowed log-magnitude spectrograms for environmental sounds in conjunction with PGHI for phase reconstruction. Next, we summarized that StyleGAN architectures were best suited for inducing semantic guidance when trained on unlabeled data, as the disentangled intermediate $\mathcal{W}$ space can be utilized to provide semantic guidance. Finally, as recent Latent Diffusion Models (LDMs), such as AudioLDM, have demonstrated remarkable capabilities in generating sounds based on text prompts, we summarized that extending their functionality to steer sound generation in a fine-grained way to be a productive avenue of research for sound design CSTs.

Chapter 3

User-Defined Semantic Attribute Guidance from Unlabeled Training Data

Chapter Synopsis

In this chapter[1], we address **RQ1**. In Chapter 1, we motivated the need for modeling semantically unlabeled sounds. In this chapter, we specifically target modeling the subclass of environmental sounds, namely audio textures. We propose a method to induce semantic control over a StyleGAN unconditionally trained on unlabeled texture datasets. We develop an example-based framework to determine guidance vectors for audio texture generation based on user-defined semantic attributes. Our approach leverages the semantically disentangled latent space of the StyleGAN. Using a few synthetic examples to indicate the presence or absence of a semantic attribute, we infer semantic guidance vectors in the latent space of the StyleGAN to control that attribute during generation. Our results show that our framework can find user-defined and perceptually relevant guidance vectors for controllable generation for audio textures. Furthermore, we demonstrate an application of our framework to other tasks, such as selective semantic attribute transfer.

[1]With minor modifications from:
Kamath, P., Gupta, C., Wyse, L., & Nanayakkara, S. (2024). Example-Based Framework for Perceptually Guided Audio Texture Generation. In IEEE/ACM Transactions on Audio, Speech, and Language Processing (Vol. 32, pp. 2555–2565). Institute of Electrical and Electronics Engineers (IEEE). doi: 10.1109/taslp.2024.3393741

3.1 Introduction

Audio textures are a subclass of environmental sounds, such as water filling a container or a wooden drumstick repeatedly hitting a metal surface. Guided or controllable generation of such sounds using deep neural networks is usually achieved by conditioning generative models using semantically labeled data. For instance, impact sound textures can be semantically guided using object or material properties of the impact surface, and a continuously varying water-filling texture can be guided by attributes such as the fill level of the container. While large datasets for audio textures can be readily recorded, labeling these sounds using semantic attributes such as material hardness or fill level is difficult. Therefore, to control generation, we develop a method to infer the vectors for semantic attribute guidance without the supervision of large labeled datasets.

Generative adversarial networks (GANs) [138] such as StyleGANs [125, 126] generate semantically disentangled latent spaces by learning the most statistically significant factors of variation within a dataset. Such disentangled latent spaces can be analyzed to find guidance vectors for controllable generation. We thus analyze the disentangled latent space of a StyleGAN to find guidance vectors based on user-defined semantic attributes to control generation.

This thesis proposes a method that uses audio examples to guide latent space access and navigation. Similar to music information retrieval (MIR) techniques such as query-by-example [139–142] and query-by-humming [143–146] we generate synthetic sound examples representative of the semantic attribute we want to control during generation. We encode these examples into the latent space of a StyleGAN unconditionally trained on real-world audio textures. Then we use these latent embeddings to define guidance vectors in the latent space along which desired semantic attributes can be systematically varied during texture generation. As shown on our webpage[2], we use these guidance vectors to guide texture generation for various user-defined semantic attributes such as “Brightness”, “Rate” or “Impact Type” for impact sounds and “Fill-Level” for the continuously varying texture of water filling.

We validate the effectiveness of our method for user-defined semantic guidance of texture generation through a comprehensive attribute rescoring analysis. We also conduct perceptual listening tests to evaluate the effectiveness of our method in changing specific attributes for various randomly generated sounds. In summary, our contributions are:

[2]https://purnimakamath.com/thesis-related/chapter_3/

- An Example-Based Framework (EBF) to find user-defined attribute guidance vectors to control audio texture generation semantically.
- A synthetic audio query approach for latent space exploration of a generative model.
- An application of our framework for semantic attribute transfer between textures.

3.2 Related Work

3.2.1 Supervised Controllability in Audio

Generative models for music, such as [13, 79], enable controllability by training on datasets with labels. This supervision helps organize the model's latent space according to the timbre-specific features in the datasets. Musical instrument datasets are usually labeled during dataset creation [79], and such labels are used to conditionally train generative models using attributes for pitch, loudness, or instrument timbres [13, 147]. Further, some architectures [123] additionally condition generation by extracting attributes such as sharpness or warmth automatically from the sound using feature extractors such as Audio Commons [148] and Essentia [149]. Similarly, DDSP [80] based architectures, such as DDSP-SFX [150], extract attributes such as loudness and pitch from the sounds to condition generation. While such supervised training methods are highly effective for modeling musical instrument sounds, their effectiveness is limited in textures due to the lack of large-scale semantically labeled audio texture datasets. Further, the attributes used to control generation for inharmonic audio textures differ from those of musical sounds [98, 99]. For instance, when synthesizing impact sounds, we are more likely to be interested in controlling the object or material properties (such as impact surface hardness, etc.) than attributes such as pitch or loudness typically associated with musical sounds. Currently, there is a lack of audio texture datasets with such object or material property labels that can be used for supervised training.

To circumvent this lack of attribute labels for audio textures, MorphGAN [115] uses features extracted from the penultimate layer of a classifier for supervision to generate smooth texture morphs. Similarly, DarkGAN [124] is trained on soft labels distilled from an audio tagging classifier [151] trained on tags from the AudioSet ontology [152]. The Sound Model Factory [116] trains a GAN, which is used to create novel timbres followed by an RNN trained on sounds produced from the GAN and conditioned on points along smoothly parameterized trajectories through the GAN latent space. These supervised

training methods rely on an additional class or parametric information while training generative algorithms. Since GANs, particularly StyleGANs can disentangle the latent space based on semantic attributes in the training data [126], our research explores finding user-defined semantic directions in the latent space of a StyleGAN to guide generation without the need for any explicit conditioning or labeled data during training.

3.2.2 Unsupervised Controllability in Audio

In computer vision, algorithms such as [153, 154] leverage StyleGAN's ability to disentangle the latent space to find directional vectors for editing semantics on images. Similarly, in audio, GANSpaceSynth [155] applies the GANSpace algorithm to control a pre-trained GANSynth trained on musical instruments in an unsupervised manner. More recently, in computer vision, Semantic Factorization (SeFa) [154] performed better than other unsupervised algorithms to find vectors for controllable generation in the latent space of a pre-trained GAN. This method decomposes the layer weights that create the disentangled representation to find the vectors for maximum variation. Such vectors are then used to edit semantics on unconditionally generated images. However, the directional vectors generated using SeFa must be semantically labeled manually after observing edits across multiple samples.

For speech and music [156, 157] infer controllability based on supervision from a few labels. For images, FLAME [158] uses supervision from a few positive-negative image pairs by semantic editing and inverting real images in a StyleGAN's latent space. Direction vectors for semantic attribute editing are found by optimizing for cosine similarity between the pairs' difference vectors. In our work, we modify the FLAME method for audio textures and propose using a few fully synthetically generated examples to assist in deriving vectors in the latent space of a StyleGAN for attribute controllability. A cluster of similar synthesized audio examples is inverted [159] to define clusters in StyleGAN's latent space. A prototype [160] latent vector is derived from each cluster and is an abstract average of the semantic cluster they represent. Since such prototypes are designed to differ in a specific attribute, the difference vector between them in the latent space can be used for guiding audio texture synthesis and for semantic attribute transfer.

3.2.3 Synthetic Texture Generation

While real-world sounds could also be inverted to find latent representations in a trained StyleGAN, they are much more difficult to control than parametric acoustic sound synthesizers [93–96] or physics-based models [97]. For our inharmonic textures, we use a physically informed synthesis technique in William Gaver's seminal work on auditory perception [98, 99]. His approach is based on the idea that humans hear and describe sound events in terms of their sources and source attributes better than the acoustic properties of the sounds themselves. The sound events in Gaver sounds are modeled on the physics of the objects interacting to produce the sound, such as the hardness of the material under impact or the force of impact. Gaver [98] refers to analysis-by-synthesis as updating synthesis parameters to match a target sound, which we use to discover StyleGAN latent vectors with synthetic audio queries.

Although algorithmically synthesized sounds can sound unnatural, we employ them only for querying and searching the latent space of a StyleGAN. Multi-event synthetic textures can be quickly and easily generated using an analysis-by-synthesis approach with attributes adequate for this exploration task.

3.3 Proposed Framework

As shown in Figure 3.1, we partition our goal to find semantic attribute vectors for controllable texture generation and propose a framework comprised of the following modules:

- A Generator module (G_s) of a StyleGAN trained on real-world audio for high-fidelity texture synthesis,
- A GAN Encoder (E), also known as a GAN inversion network, to encode an audio example into the latent space of a pre-trained StyleGAN,
- A parametric Gaver synthesizer for sounds used to locate desired points in the latent space of the StyleGAN,
- An algorithm to derive semantic attribute clusters and prototype vectors for guiding semantic synthesis trajectories in the latent space of the StyleGAN.

Chapter 3 introduced the StyleGAN architecture central to this framework. Figure 3.1c illustrates our framework (during inference). G_s is a StyleGAN generator and E is the

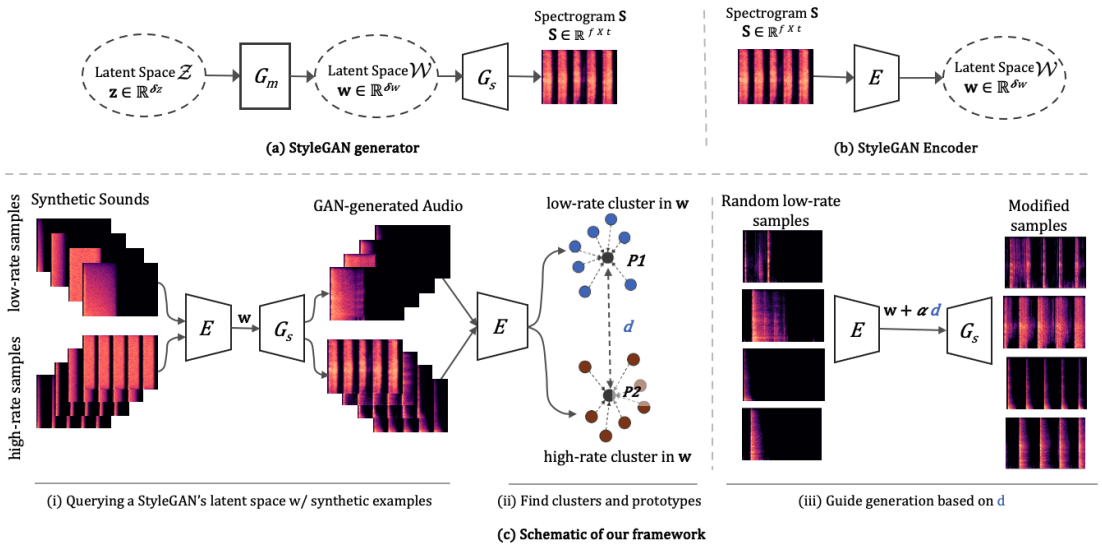


FIGURE 3.1: Schematic outlining the modules within our framework. (a) A StyleGAN's generator. Mapping network G_m maps latent space $\mathcal{Z}$ to intermediate latent space $\mathcal{W}$ ($\mathbb{R}^{\delta_z} \rightarrow \mathbb{R}^{\delta_w}$). Synthesis network G_s maps an intermediate latent vector $\mathbf{w}$ to spectrograms $\mathbf{S}$ ($\mathbb{R}^{\delta_w} \rightarrow \mathbb{R}^{f \times t}$). (b) Schematic of an Encoder E which inverts spectrograms to the intermediate latent space $\mathcal{W}$ ($\mathbb{R}^{f \times t} \rightarrow \mathbb{R}^{\delta_w}$). (c) Schematic of our framework during inference.

GAN Encoder. (i) We generate synthetic Gaver sounds for a semantic attribute we want to control. In the diagram above, we demonstrate this using "Rate", or the number of impact sounds in a sample, as the semantic attribute. We encode these synthetic sound examples into the latent space of a StyleGAN to find their $\mathbf{w}$ embeddings. (ii) Next, we derive the semantic attribute clusters and generate prototypes using the algorithm elaborated in section 3.3.4. The direction vector to guide generation for that semantic concept is indicated by "**d**". (iii) Shows how we can use direction vector "**d**" to guide generation on any randomly generated audio sample to increase or decrease "Rate".

3.3.1 GAN for Audio Textures

While our framework can be applied to derive attribute guidance vectors within the latent space of any pre-trained generative model, such as Variational Autoencoders [34], Progressive GANs [123], or StyleGANs for audio, in this paper, we demonstrate this using StyleGAN2 [126] trained on audio textures. Figure 3.1 (a) shows a schematic of a StyleGAN2's generator. We have excluded the discriminator section of StyleGAN2 in the schematic for brevity. Overall, a StyleGAN2's generator can be modeled as a function $G(.)$ that maps a latent space $\mathcal{Z}$, where $\mathbf{z} \in \mathbb{R}^{\delta_z}$, to the higher dimensional spectrogram space $\mathbf{S} \in \mathbb{R}^{f \times t}$, such that $\mathbf{S} = G(\mathbf{z})$. Here δ_z is the dimensionality of the $\mathcal{Z}$ space, and

f, t are the number of frequency channels and time frames of the generated spectrogram, respectively. StyleGANs further learn an intermediate representation $\mathcal{W}$, where $\mathbf{w} \in \mathbb{R}^{\delta_w}$, between that of $\mathcal{Z}$ and $\mathcal{S}$ via a mapping network $G_m(.)$. This intermediate latent space further disentangles factors of variation as compared to the latent $\mathcal{Z}$ space [126]. Further, a synthesis network $G_s(.)$ maps the $\mathbf{w}$ vector to a spectrogram $\mathbf{S}$. A StyleGAN's intermediate $\mathcal{W}$ latent space is considered to be more disentangled, in terms of the various factors of variation in the training data, than its $\mathcal{Z}$ space [125]. We thus operate our framework and method in the intermediate latent space $\mathcal{W}$ to find semantically meaningful directions for controllability during generation.

3.3.2 GAN Encoder

Figure 3.1 (b) shows a schematic of our Encoder. While GANs learn to map latent space embeddings to real-world sounds, GAN inversion techniques learn inverse mapping, i.e., from the real-world sounds to the latent space embeddings. We adapt the encoder model from [159] to estimate a $\mathbf{w}$ vector from an audio spectrogram randomly sampled from a pre-trained StyleGAN2. This model is based on the ResNet [161] architecture. Residual Network (or ResNet) architectures use stacks of residual blocks (a set of convolutional layers with skip connections) to learn residual functions with reference to the layer inputs. Such architectures have been previously successfully used for large-scale audio classification tasks [162].

The input to the Encoder, as shown in Figure 3.1 (b), is a spectrogram of the audio sample to be inverted. Previously, [163, 164] have shown that masking techniques for spectrograms are effective while learning generalized vector representations for audio. We extend this idea of arbitrarily masking the spectrogram to learn a $\mathbf{w}$ vector representation from the Encoder. This approach is especially useful during inference to generalize the encoder to synthetic Gaver sounds. It assists in projecting the synthetic sounds into a reasonable part of the latent space even though the encoder (or the GAN) is not directly trained on these sounds. Note that while training the Encoder, the weights of the StyleGAN2 generator are frozen. We only optimize the Encoder weights during training.

For noisy textures, such as the sounds of water filling a container, we further employ amplitude thresholding of the spectrogram during training. This thresholding ensures that the encoder ignores the low-level noise and focuses on the spectrogram's most prominent events and frequencies while estimating the $\mathbf{w}$ vector. To train the Encoder, we modify the loss function from [159] to estimate only in the $\mathcal{W}$ space instead of $\mathcal{Z}$ as:

$$\mathcal{L} = \mathbb{E}_{\mathbf{z}\sim\mathcal{N}(0,1),\mathbf{w}=G_m(\mathbf{z}),\mathbf{S}=G_s(\mathbf{w})}[\|\mathbf{S} - G_s(E(\mathbf{S}))\|_2^2 + \|\mathbf{w} - E(\mathbf{S})\|_2^2] \tag{3.1}$$

In Equation 3.1, $G_m(.)$ is the mapping network, $G_s(.)$ is the synthesis network of the StyleGAN2, and $E(.)$ is the Encoder that inverts a spectrogram $\mathbf{S}$ into the $\mathcal{W}$ space. While training the encoder, we randomly sample a $\mathbf{z}$ from the $\mathcal{Z}$ space to generate the target spectrogram $\mathbf{S}$ using $G(\mathbf{z})$. We estimate the $\mathbf{w}$ for this spectrogram using the encoder $E(\mathbf{S})$. For the first loss term, we pass the inverted $\mathbf{w}$ through the synthesis network of the generator $G_s(\mathbf{w})$ and find the mean squared error (MSE) loss between the original and reconstructed samples. The second term is the MSE loss between the actual and the estimated $\mathbf{w}$ vector.

The loss function of the original Encoder algorithm [159] additionally used a perceptual similarity loss term called LPIPS [165] that calculates the distance between image patches to preserve the perceptual similarity of the estimated images. In our experiments, we evaluate the need for such perceptual loss terms for our task compared to our loss formulation in Equation 3.1.

3.3.3 Synthesizing Examples with User-Defined Semantics

To generate audio examples for querying the GAN latent space, we use two Gaver synthesis methods - (1) based on physical parameters of the interacting objects and (2) based on object resonance as a series of bandpass filters. The first method is useful in generating sharp impact sounds or dripping sounds, and the second is for producing a larger variety of impacts and scraping sounds. More formally, a synthetic impact sound can be described as -

$$F(t) = \sum_n \phi_n e^{\zeta_n t} cos\omega_n t \tag{3.2}$$

where $F(t)$ describes the generated sound, ϕ_n is the amplitude of the n^{th} partial, ζ_n is a damping constant, and ω is the frequency of the partial, $\sum$ signifies a sum over the total number of partials. From an ecological perspective, each component in the equation controls a physical aspect of the objects interacting to generate the impact sound. For instance, ζ in the equation controls the material hardness, ϕ controls the force of impact, and ω and n control the object's size.

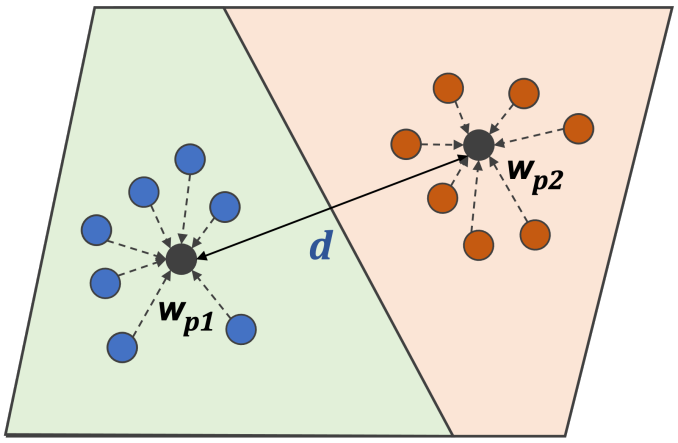


FIGURE 3.2: Schematic for generating semantic attribute clusters, prototypes $\mathbf{w_{p1}}$ and $\mathbf{w_{p2}}$, and the direction vector $\mathbf{d}$.

Method 2 creates impact and scraping sounds by passing Gaussian noise $\mathcal{N}(0, I)$ through band-pass and fade filters. The amplitude of the impact sound governs the force of impact, while the frequency bands, together with damping provided by linear or exponential fade filters, govern the material of impact of the sound.

3.3.4 Generating Semantic Clusters, Prototypes, and Guidance Vectors

Having generated synthetic Gaver sounds, we invert them into the latent space of the StyleGAN to generate the $\mathbf{w}$ embeddings. We then cluster the sounds in the $\mathcal{W}$ space to generate prototypes as shown in Figure 3.2.

Assume, for example, that we want to derive directional vectors to control the attribute of "Brightness" of an impact sound. We define brightness as an attribute that indicates the presence or absence of high-frequency components in a sound. We generate a cluster of Gaver sounds where the semantic attribute is present (represented by blue dots in the figure) and another cluster of Gaver sounds where the semantic attribute is absent (or "dull" impact sounds represented by orange dots). We find the prototypes $\mathbf{w_{p1}}$ and $\mathbf{w_{p2}}$ representative of each semantic attribute cluster using Algorithm 1.

To generate our prototypes, we adopt a technique from computer vision for generating Eigenfaces. First, we shift or center the inverted $\mathbf{w}$ embeddings of the synthetic samples by subtracting the center of mass of the $\mathcal{W}$ space, namely $\mathbf{w\_avg}$, from them. $\mathbf{w\_avg}$ is the the mean $\mathbf{w}$ vector encoded from our training set. These mean-subtracted $\mathbf{w}$ embeddings record how each synthetic sample differs or varies w.r.t the mean sample in

Algorithm 1: Get Prototype

Input:

$\mathbf{W_n}$ is a matrix of $\{\mathbf{w_0}, ..., \mathbf{w_n}\}$ encoded synthetic samples as column vectors, such that $\mathbf{W_n} \in \mathbb{R}^{\delta_w \times n}$;
w_avg is a column vector for the center of mass of $\mathcal{W}$ space;

Output:

w_ptype the prototype representation;

Function *GetPrototype (*$\mathbf{W_n}$*,* **w_avg***)*:

$\mathbf{W} \in \mathbb{R}^{\delta_w \times n} \leftarrow \mathbf{W_n} - \mathbf{w\_avg}$; ▶Subtract **w_avg** from each column of $\mathbf{W}$
$\mathbf{U}, \mathbf{S}, \mathbf{V} \leftarrow \mathbf{SVD}(\mathbf{W})$;
$\mathbf{s} \leftarrow diag(\mathbf{S})$; ▶Extract singular values from diagonal matrix $\mathbf{S}$ as vector $\mathbf{s}$;
$\mathbf{w\_ptype} \leftarrow \mathbf{w\_avg} + \mathbf{u_s}\mathbf{u_s}^T\overline{\mathbf{w}}$; ▶$\overline{\mathbf{w}}$ is mean $\mathbf{w}$ sample vector from $\mathbf{W_n}$
▶$\mathbf{u_s} \leftarrow \mathbf{U}[:, argmax(\mathbf{s})]$

return w_ptype

the $\mathcal{W}$ space. Next, we stack all the mean-subtracted $\mathbf{w}$ embeddings for the synthetic samples in a semantic cluster together as matrix columns. We perform singular value decomposition on this matrix and select the component associated with the maximum singular value to construct the prototype. The intuition behind doing this is that after decomposition, the component with the highest singular value has the most common prominent feature amongst all the analyzed samples, i.e., the semantic attribute being modeled. Furthermore, by modeling the mean-subtracted $\mathbf{w}$ embeddings, we ensure that we model the variations in the $\mathbf{w}$ vectors better instead of focusing on the shared common features encoded by **w_avg**. Constructing a prototype this way is more robust to outliers or artificial synthesis artifacts.

The difference between the $\mathbf{w}$ embeddings of the two prototypes $\mathbf{w_{p1}}$ and $\mathbf{w_{p2}}$, denoted as direction vector ($\mathbf{d}$), can be used to continuously and sequentially edit the semantic attribute as follows -

$$\mathbf{w_{edited}} = \mathbf{w} + \alpha * \mathbf{d} \qquad \text{(where } 0 < \alpha < 1\text{)} \tag{3.3}$$

where $\mathbf{w}$ is a randomly chosen $\mathcal{W}$ vector, '+' and '*' indicate element-wise operations, and α is a continuous scalar parameter that signifies step size. Larger values of α correspond to a greater degree of semantic attribute edit on the sample. Further, using $-\mathbf{d}$ reverses the direction of the edit. The edited sounds can be reconstructed by passing the $\mathbf{w_{edited}}$ through the StyleGAN2 synthesis network $G_s(.)$.

3.4 Experiments

3.4.1 Datasets

We use two audio texture datasets in our experiments: (1) The Greatest Hits dataset [166] to demonstrate the effectiveness of our approach on impact sounds and (2) a Water filling a container dataset [167] for continuously varying audio textures. Through these two datasets, we demonstrate our method's effectiveness in covering a range of event-based and continuously varying textures.

3.4.1.1 The Greatest Hits Dataset

This dataset contains audio and video recordings of a wooden drumstick probing indoor and outdoor environments by hitting, scraping, and poking different objects of different material densities. We use this dataset to explore the rich timbres arising from the interactions between the wooden drumstick and various hard and soft surfaces such as tree trunks, dirt, leaves, metal cans, ceramic mugs, carpets, soft cushions, etc. The dataset contains approximately 10 hours of denoised audio split into 977 audio files, each approximately 35 seconds. Each file contains impact sounds interacting with different types of objects. We split the audio files into consecutive 2-second sounds sampled at 16kHz to train our StyleGAN2 unconditionally. We develop semantic attribute clusters, prototypes, and attribute guidance vectors for the attributes ***Brightness*** (whether the sound contains mostly high-frequency components or is dark or dull containing mostly low-frequency components), ***Rate*** (whether the number of impact sounds in a sample is high or low), and ***Impact Type*** (whether the sounds are sharp impacts or scraping/scratchy sounds made by dragging the stick across the surface).

3.4.1.2 Water filling a container

This dataset [167] contains 50 audio recordings of water filling a container at an approximately constant rate for an average duration of ∼30 seconds. We develop semantic attribute clusters, prototypes, and attribute guidance vectors for the continuously varying attribute of ***Fill-Level*** of the container. We sample the recorded audio files using a sliding window of 100ms to generate approximately 10,000 2-second audio files sampled at 16kHz to train our StyleGAN2 unconditionally. We choose a small sliding window size of 100ms to achieve better interpolatability [18] for ***Fill-Level*** in the $\mathcal{W}$ space of the GAN.

3.4.2 Implementation Details

StyleGAN2: We set $\mathcal{Z}$ and $\mathcal{W}$ space dimensions δ_z and δ_w both to 128 and use 4 mapping layers in the Generator for all our experiments. Further, we use the log-magnitude spectrogram representations generated using a Gabor transform [168](n_frames= 256, stft_channels= 512, hop_size= 128), a Short-Time Fourier Transform (STFT) with a Gaussian window, to train the StyleGAN2 and the Phase Gradient Heap Integration (PGHI) [111] for high-fidelity spectrogram inversion of textures to audio [110]. For training the generator and discriminator of the StyleGAN2, we use an Adam optimizer with a learning rate of 0.0025, β_1 as 0.0, and β_2 as 0.99.

Encoder Training: We use a ResNet-34 (a stack of 34 residual blocks) [161] backbone as the architecture for our GAN Encoder network. We use an amplitude thresholding of -17dB for Water and -25dB for the Greatest Hits. We mask the frequency components with a magnitude below -17 or -25dB for the respective datasets. We use an Adam optimizer to train the Encoder with a learning rate of 0.00001, β_1 as 0.5, and β_2 as 0.99.

Gaver Sound Synthesis: In all our experiments, we use 10 synthetic Gaver examples (5 per semantic attribute cluster) to generate the guidance vectors for controllable generation. We outline a cluster-based analysis for real and synthetic sounds using UMAP [169] visualizations on our supplementary webpage.

3.4.3 Evaluation metrics

For audio quality, we utilize the *Fréchet Audio Distance* [39](FAD) metric. FAD is the distance between the distribution of real and synthesized audio data embeddings extracted from a pre-trained VGGish model. We utilize this metric to evaluate the quality of sounds generated by inverting the synthetic Gaver sounds and real-world sounds from the latent space of the GAN.

To evaluate the effectiveness of our method in changing a semantic attribute of a texture, we perform *rescoring analysis*. By *rescoring*, we mean the change in accuracy scores reported by an attribute classifier before and after the change in the semantic attribute on a sound. For this, we train an attribute presence or absence classifier based on a Dense Convolutional Network (or DenseNet) architecture [170]. Previously, [171] showed that an ImageNet pre-trained model fine-tuned for audio datasets could achieve state-of-the-art results in environmental sound classification tasks. We adapt the classifier from [171] using a DenseNet architecture with ImageNet pre-training and fine-tune it

for our attribute classification task. Please see our supplementary webpage for classifier architecture and training details.

We begin our evaluation by manipulating an attribute on randomly generated sounds. We then record how the attribute classifier score changes for those sounds before and after the manipulation. Further, we evaluate if the attribute change occurs without modifying other sound attributes. For instance, when editing ***Brightness***, we first analyze if the intended attribute of brightness changes. We then analyze if other attributes such as ***Rate*** changes with it. As our datasets are unlabelled, to train the *rescoring analysis* classifier, we manually curate and label a small subset of sounds. To do this, we selected approximately 250 samples of 2-second sounds for each semantic attribute under consideration. This manual curation involved visually analyzing the video and listening to the associated sounds to detect the semantic attribute. For more details on the dataset curation, please see our webpage. Note that this curated dataset is only used for quantitative analysis and not to train our GAN or Encoder models.

3.4.4 Baseline Selection

We evaluate our method's effectiveness in finding user-defined attribute guidance vectors in the latent space of the GAN by comparing it with an unsupervised method for latent semantic discovery called closed-form Semantic Factorization (SeFa) [154].

Typically, to evaluate novel methods for controllability, the ideal method would be to compare the method with a conditionally trained model. If we had a large dataset of labeled audio textures, we could have trained a GAN conditionally using those labels for comparison. However, it is difficult to annotate sounds for continuously varying or fine-grained semantic labels. For example, annotating the sound of water filling to determine the fill level of the container, such as whether it's 30% or 40% full based on the sound alone, is challenging. While labeling such sounds is an actively researched topic, due to the current lack of granularly labeled datasets, we needed a baseline to compare our work, which did not rely on training with supervision. Based on these considerations, we chose SeFa as our baseline.

SeFa decomposes the pre-trained weights of a GAN to find statistically significant vectors for guided generation. Although SeFa is relatively under-studied in the audio domain, we use it as a baseline for comparison because, like our method, SeFa works on unconditionally trained GANs. Given the novelty of our task in deriving guidance vectors in a post-hoc fashion, to the best of our knowledge, the SeFa method is the state-of-the-art method in this regard. We thus use it for comparison.

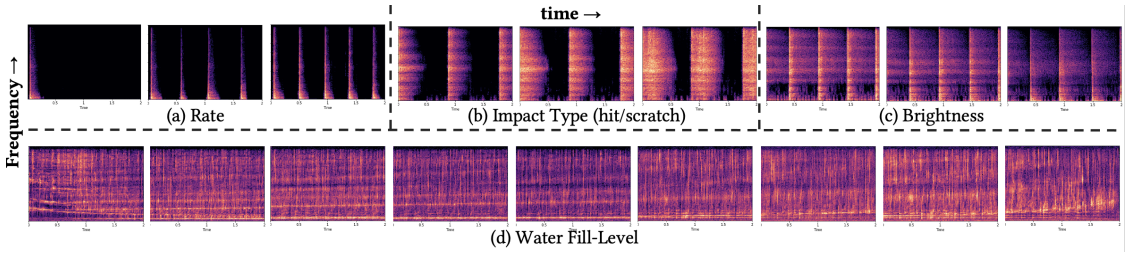


FIGURE 3.3: (Top Row) Spectrogram examples of guided generation using our method based on change in the attributes of (a) Rate (increases L to R), (b) Impact Type (becomes scratchy L to R), and (c) Brightness (decreases L to R). Note that for each example, the other attributes do not change as one attribute changes. (Bottom Row) Examples of guided generation for water filling a container based on Fill-Level. Note how the Fill-Level and its frequency components gradually increase from L to R. All sounds can be listened to on our webpage `https://purnimakamath.com/thesis-related/chapter_3/`.

3.4.5 Experimental Details

We first conduct ablation studies to understand the effects of individual components of the loss functions outlined in section 3.3.2. We report this analysis using both *rescoring analysis* and *FAD* scores. Next, we study the impact of the change of an attribute on other attributes under consideration. We then compare our method (EBF) with the SeFa method as a baseline. Finally, we qualitatively study the effectiveness of our method by conducting listening tests. Figure 3.3 shows some spectrogram examples of guided generation using our method. The standard error of means in all tables in this section was reported by bootstrapping the samples over 100 iterations.

3.4.5.1 Ablation Studies

We conduct three types of ablation studies in our paper - (1) to study the effect of different components of the loss function on the Encoder, (2) to study the effect of the number of synthetic samples needed to create a semantic cluster, and (3) to study the effect of the magnitude of scalar α in equation 3.3.

Ablating Encoder Loss Components: We first study the effect of using LPIPS and MSE loss terms with and without thresholding while training the Encoder. Table 3.1 shows the *rescoring* accuracy scores for attribute changes for each type of Encoder. (↑) indicates that higher values are better. We report the accuracy for the main attribute and the average change reported in other attributes for each attribute change. Ideally, we would like the main attribute accuracy to be high and the change in other attributes to be low. For the Greatest Hits dataset, we find that the system with MSE and MSE+Thresholding

outperforms the one with MSE+LPIPS loss for all attributes. For Water, using an Encoder with MSE and thresholding works best.

Table 3.2 shows the FAD scores for GAN-generated sounds (column called GAN) and Encoder reconstructions for each type of Encoder for both GAN-generated and synthetic Gaver sounds. The FAD Scores were computed based on 10,000 randomly generated samples compared to the entire training set. We find that the encoder trained using MSE only or MSE+thesholding outperforms MSE+LPIPS regarding the quality of the generated audio (FAD scores). Thus, based on this table and Table 3.1, we choose the Encoder with MSE+Thresholding (qualified with a † in Tables 3.1 and 3.2) as the best-performing Encoder for both datasets for the remainder of the paper.

Ablating the Number of Gaver Samples: Next, we study the effect of the number of Gaver samples used to find the guidance vectors for different attributes. We derive guidance using different N, starting with N=1 to N=5. We observe that as N increases, the effectiveness of the directional vector edits also increases. Also, such edits preserve other unedited attributes better with higher N. The samples with different N's can be hlyellowlistened to on our supplementary webpage.

Ablating the effect of the scalar value α: In equation 3.3, the scalar value α governs the magnitude of the edit performed on the sample $\mathbf{w}$ using the semantic attribute direction vector $\mathbf{d}$. In all our experiments, the value of α is in between $[0, 1]$. Also, all examples on our supplementary webpage edit $\mathbf{w}$ in linear steps until $\alpha = 1$. In this section, we qualitatively study the effect of using a value $\alpha > 1$, i.e., extrapolating the semantic edits beyond the magnitude of the difference vector $\mathbf{d}$ (or beyond the selected prototype in the latent manifold). The samples from different α's can be found on our webpage. We observe that, for all attributes, for $\alpha >= 3$, the edited $\mathbf{w}$ vectors escape the latent $\mathcal{W}$ manifold and generate noisy or unintelligible samples.

3.4.5.2 Baseline Comparison

Table 3.3 reports the *rescoring analysis* for each attribute using our method compared to SeFa. We report the score for change in the main intended attribute being edited and the average change in other attributes. (↑) indicates higher values are better. Our method reports better accuracies for change in the main attribute for both datasets than SeFa.

We further report pairwise attribute edit comparisons to study the effect of change in one attribute individually on every other attribute. Tables 3.4 and 3.5 show this for the

TABLE 3.1: Ablation Studies

	Greatest Hits						Water
	Brightness		Rate		Impact Type		Fill-Level
	Acc.(↑)	Avg. Change Others (↓)	Acc.(↑)	Avg. Change Others (↓)	Acc.(↑)	Avg. Change Others (↓)	Acc.(↑)
EBF: MSE+LPIPS [165]	0.53 ± 0.08	0.24 ± 0.04	0.64 ± 0.08	0.35 ± 0.07	0.69 ± 0.07	0.20 ± 0.04	0.60 ± 0.1
EBF: MSE	0.71 ± 0.06	0.14 ± 0.03	0.88 ± 0.06	0.30 ± 0.07	**0.81** ± 0.04	0.24 ± 0.4	0.27 ± 0.1
EBF: MSE+Thresholding†	**0.82** ± 0.05	0.21 ± 0.06	**0.89** ± 0.06	0.35 ± 0.1	0.80 ± 0.06	0.27 ± 0.06	**0.97** ± 0.04

TABLE 3.2: FAD Scores for GAN generated sounds and Encoder reconstructions

	Greatest Hits			Water		
	GAN	GAN Recon.(↓)	Gaver Recon.(↓)	GAN	GAN Recon.(↓)	Gaver Recon.(↓)
EBF: MSE+LPIPS [165]	0.6	1.12	4.40	1.17	1.92	9.45
EBF: MSE		**0.72**	4.61		1.59	11.77
EBF: MSE+Thresholding†		2.83	**4.16**		**1.42**	**7.92**

TABLE 3.3: Comparison with Baseline

	Greatest Hits						Water
	Brightness		Rate		Impact Type		ton Fill-Level
	Acc.(↑)	Avg. Chng Others (↓)	Acc.(↑)	Avg. Chng Others (↓)	Acc.(↑)	Avg. Chng Others (↓)	Acc.(↑)
SeFa [154]	0.49 ± 0.11	0.19 ± 0.12	0.45 ± 0.12	0.29 ± 0.14	0.42 ± 0.15	0.31 ± 0.09	0.92 ± 0.09
EBF: MSE+Thresholding †	**0.82** ± 0.05	0.21 ± 0.06	**0.89** ± 0.06	0.35 ± 0.10	**0.80** ± 0.06	0.27 ± 0.06	**0.97** ± 0.04

Greatest Hits dataset and Table 3.6 for the Water dataset. For SeFa, since we do not know which vector (of the δ_w=128 dimensions) edits a specific attribute, we report scores for edits performed by the top 10 vectors with the highest singular values (top 10 for Greatest Hits and top 3 for Water Filling) for comparison in the table. (↑) indicates higher values are better and scores highlighted with "*" indicates no significant differences ($p > 0.05$). Each row indicates a semantic attribute manipulation using a specific guidance vector, and each column evaluates how the scores change for that attribute. The darkened cells in the table indicate dimensions with the highest score for a semantic attribute (in that column).

TABLE 3.4: Pairwise rescoring for Greatest Hits (EBF)

	Brightness(↑)	Rate(↑)	Impact Type(↑)
Brightness	**0.82** ± 0.06	0.06 ± 0.04	0.40 ± 0.07
Rate	0.40 ± 0.09	**0.89** ± 0.06	0.38 ± 0.09
Impact Type	0.35 ± 0.07	0.19 ± 0.03	**0.80** ± 0.05

TABLE 3.5: Pairwise rescoring for Greatest Hits (SeFa)

	Brightness(↑)	Rate(↑)	Impact Type(↑)
Dimension 0	0.10 ± 0.07	0.07 ± 0.06	0.18 ± 0.13
Dimension 1	0.30 ± 0.11	**0.45** ± 0.12	0.28 ± 0.16
Dimension 2	0.31 ± 0.11	0.09 ± 0.06	**0.42** ± 0.15*
Dimension 3	**0.49** ± 0.12	0.09 ± 0.06	0.30 ± 0.16
Dimension 4	0.12 ± 0.08	0.14 ± 0.08	0.17 ± 0.12
Dimension 5	0.31 ± 0.11	0.10 ± 0.06	0.30 ± 0.14
Dimension 6	0.32 ± 0.11	0.14 ± 0.08	0.19 ± 0.13
Dimension 7	0.14 ± 0.08	0.10 ± 0.06	**0.38** ± 0.16*
Dimension 8	0.18 ± 0.09	0.09 ± 0.07	0.28 ± 0.16
Dimension 9	0.34 ± 0.10	0.11 ± 0.07	0.20 ± 0.13

For both datasets, our method reports a significant change in the main attribute being manipulated. Further, we analyze if each dimension or direction vector from both methods manipulates only a single attribute. For this, we perform a two-way t-test for the scores between any two SeFa dimensions. We particularly notice that for SeFa, the semantic attribute of ***Impact Type*** is affected by at least two dimension vectors, namely Dimension 2 and Dimension 7 in Table 3.5. This implies that methods such as SeFa may not always guarantee one-to-one correspondence between statistically found vectors for guidance and the semantic attributes of interest. Furthermore, the first dimension associated with the largest singular value extracted using SeFa does not correlate with both datasets' main perceptually varying attributes. This implies that such automated methods do not always guarantee to find vectors that control perceptually relevant attributes in the latent space of a generative model for audio.

TABLE 3.6: Pairwise rescoring for Water (EBF and SeFa)

EBF†	Fill-Level(↑)
Fill-Level	**0.97** ± 0.04

SeFa	Fill-Level(↑)
Dim 0	0.14 ± 0.11
Dim 1	**0.92** ± 0.09
Dim 2	0.27 ± 0.16

TABLE 3.7: Listening Test Results

	Water	Greatest Hits		
	Fill Level(↑)	Brightness(↑)	Rate(↑)	Impact Type(↑)
SeFa	0.47 ± 0.03	0.75 ± 0.03*	0.58 ± 0.04	0.51 ± 0.04
EBF†	**0.55 ± 0.03**	0.75 ± 0.04*	**0.68 ± 0.04**	**0.67 ± 0.05**

3.4.5.3 Listening tests

We recruited 20 Amazon's Mechanical Turk (AMT) participants to evaluate the modified sounds using both methods. Only participants with more than 95% approval rate on their previous tasks on AMT across at least 1000 completed tasks were allowed to attempt our listening test. Before attempting our listening test, participants underwent a hearing screening designed for crowdsourced platforms based on [172]. The participants were requested to sit in a quiet place and use a pair of headphones for the test duration. During the hearing screening, the participants were presented with two audio samples. Each sample contained tones generated at random frequencies between $55Hz$ and $10kHz$. They were asked to count the number of tones in each audio sample. Participants who completed the screening by correctly estimating the number of tones were allowed to attempt our listening test. The audio samples in the hearing screening ensured that the participants were of normal hearing, were using a pair of headphones, and were in a quiet environment when attempting the listening test.

We created the audio samples for our main listening test by randomly sampling from the StyleGAN and then editing each sample using both methods' direction vectors. For the Greatest Hits dataset, we randomly sampled 20 sounds from the StyleGAN2's latent space and modified them using vectors derived using our method for ***Brightness***, ***Impact Type*** and ***Rate***. For SeFa, we used the vectors with the highest rescoring accuracy from Table 3.5 to manipulate the samples. We developed a listening test interface to evaluate our attribute edits. The participants were presented with the unmodified original reference sound and the manipulated samples. They were asked to evaluate if the two samples differed in the 3 attributes. We randomly sampled the latent space for the Water

dataset 10 times and modified the samples using vectors for ***Fill-Level***. As the ***Fill-Level*** for Water varies continuously, we wanted to evaluate if manipulating the sound samples sequentially and linearly using both methods preserves the interim ***Fill-Level***s (such as when the bucket is empty, quarter or half full, etc.). To do this, we use the rank-ordering interfaces outlined in [173] to measure the perceptual linearity of linearly manipulating the sample using the guidance vector for ***Fill-Level***. The interfaces for the listening tests can be viewed on our supplementary webpage.

We use accuracy scores to evaluate our listening tests, with 'accuracy' formulated as the fraction of the listening test trials where participants correctly selected the attribute being manipulated for a sample in comparison to a reference. Table 3.7 shows the scores from our listening tests for both datasets and their respective attributes. (↑) indicates that higher values are better and scores highlighted with "*" indicates no significant differences ($p > 0.05$). For Water, participants could perceptually rank-order the water-filling sounds in increasing order of ***Fill-Level*** significantly better when using our method. For the Greatest Hits dataset, participants found our method to perform significantly better while manipulating the sounds for ***Rate*** and ***Impact Type***. However, for the attribute of ***Brightness***, participants found both methods to perform equally well. By qualitatively listening and comparing the brightness samples generated by the algorithm, we find that samples generated using our method cover a wider range of brightness than SeFa (visit the supplementary webpage for examples).

3.5 Application: Selective Semantic Attribute Transfer

In this section, we demonstrate the simplicity of extending our framework to applications other than performing semantic edits of textures. The prototypes and guidance vectors derived from our method can be used to support applications such as selective semantic attribute transfer. This task is inspired by image editing applications such as Photoshop, where a user can select an object and transfer its color to another object. We envision a selective attribute transfer tool where the prototype and guidance vectors guide selecting an attribute from a reference sample and transferring it to another sample.

Figure 3.4 shows a diagram outlining the approach. Say we have a reference sample embedding $\mathbf{w_{ref}}$ and a target sample embedding $\mathbf{w}$, and we want to selectively transfer the attribute of ***Brightness*** from the reference $\mathbf{w_{ref}}$ to $\mathbf{w}$. To do this, we first project both $\mathbf{w_{ref}}$ and $\mathbf{w}$ onto the attribute guidance vector $\mathbf{d}$ between $\mathbf{w_{p1}}$ and $\mathbf{w_{p2}}$. We then

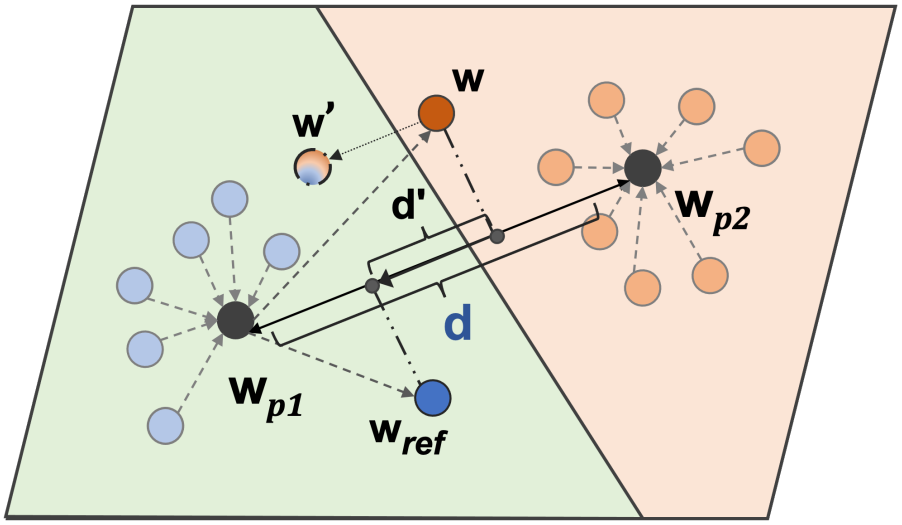


FIGURE 3.4: Semantic attribute transfer from a reference sample $\mathbf{w_{ref}}$ to a target $\mathbf{w}$, with direction vector $\mathbf{w_{p1}} \rightarrow \mathbf{w_{p2}}$ representing, say an increasing level of "Brightness". Both $\mathbf{w_{ref}}$ and $\mathbf{w}$ are projected onto to the direction vector $\mathbf{d}$. The difference vector $\mathbf{d'}$ is used to selectively edit $\mathbf{w}$ to generate $\mathbf{w'}$. $\mathbf{w'}$ will have the same brightness relationship to $\mathbf{w}$ as $\mathbf{w_{ref}}$.

edit $\mathbf{w}$ in the direction of the difference between the projections, namely $\mathbf{d'}$, to create $\mathbf{w'}$. This method not only transfers the ***Brightness*** attribute from $\mathbf{w_{ref}}$ to $\mathbf{w'}$ but also preserves the other unmodified semantic attributes of $\mathbf{w}$ as well as its original structure (position or location of the impact events along the time axis). A formal outline of this algorithm and some results from selectively transferring individual attributes such as ***Brightness*** onto a target sample can be found on our webpage.

3.6 Discussion

The main takeaway from this research is that controllability through labeled training data need not be built-in into the generative model's training and development procedures. Such guidance or controllability (or steering in terms of human-centered tools for creativity) can be induced through post-hoc methods, such as in EBF, or using other post-training algorithms [78, 153, 155]. Enabling steering or guidance in this way can enable the users of such tools to define semantics important to their creative work and not rely on the model developer's definition or understanding of their needs. Using frameworks such as EBF, users can "probe" pre-trained generative models to understand the breadth of the system's capabilities.

Constraining traversal to the latent manifold for $\alpha > 1$: In section 3.4.5.1, we qualitatively study the effect of performing edits using $\alpha > 1$, where α is the interpolation

step size. As seen in equation 3.3, our method assists in a linear traversal of the latent space using the computed direction vector to perform semantic edits on any randomly generated samples. For higher values of α, this linear way of traversing the latent space may result in $\mathbf{w}$ vectors falling outside the latent manifold. Such edited vectors may result in the generation of noisy or unintelligible samples. Other traversal methods may accommodate the local geometry of the latent space so local edits do not escape the latent manifold. One approach is to use more than two prototypes, supported by "in-between" examples, to improve the robustness of our framework. Another approach is investigating incremental non-linear traversal methods (such as in [174]). With such traversal methods, the guidance can be directed to stay within the manifold by incrementally guiding the generation using a sequence of consecutive directional $\mathbf{d}$ vectors. However, it should be noted that such methods will increase the number of computations to be performed during edits or the number of synthetic samples that will need to be manually created.

Manual curation of samples: Although our EBF method has been more effective than algorithms such as SeFa [154] at modifying the user-defined semantic attributes on a texture, some manual curation of synthetic samples is needed to find the relevant guidance vectors. On the other hand, algorithms such as SeFa are automatic and can be applied to any pre-trained GAN without any manual intervention. Thus, exploring the potential of combining SeFa's ability to automatically discover vectors for attribute manipulation with the EBF method to improve the accuracy of editing the semantic attributes will form a productive avenue for research.

Querying GANs using out-of-distribution sounds: The Encoder outlined in section 3.3.2 is trained on masked and amplitude thresholded versions of the real-world training data. In our ablation studies experiments in Section 3.4.5.1, we showed that the Encoder using amplitude thresholding out-performed others, especially for noisy water-filling sounds. This training approach assists in projecting any out-of-training-distribution sounds, such as the parametrically synthesized sounds in our case, to a reasonable part of the latent space, even though the Encoder is not directly trained on such synthetic sounds. Such an Encoder can be extended to query the StyleGAN's latent space using other out-of-distribution sounds, such as the sound generated vocally by users (i.e., query-by-humming approaches). A productive avenue for future work will be to further study the applicability or limitations of our framework in conjunction with vocal queries to guide audio texture generation perceptually.

3.7 Summary

In this chapter, we proposed an audio example-based method to perceptually guide the generation of audio textures based on user-defined semantic attributes. Using a synthesizer to create a few examples, we developed attribute guidance vectors in the latent space of a StyleGAN2 to controllably generate both impact sounds and continuously varying water-filling audio textures. We showed the effectiveness of our method in providing linearly varying controls for texture generation using objective metrics and perceptual listening tests. Furthermore, we applied our method to other signal-processing tasks, namely semantic attribute transfer.

Chapter 4

Audio Morphing with Text-to-Audio Models

Chapter Synopsis

In this chapter[1], we address **RQ2**. Sound morphing combines two sounds to generate novel and perceptually hybrid sounds simultaneously resembling both [2, 31]. In this chapter, we provide means to semantically edit and morph the sound generated by the generative algorithms to allow sound designers to explore the AI model's conceptual representational space better and in a fine-grained way. Using a pre-trained text-to-audio model, we introduce a novel algorithm to granularly morph the semantics of sounds generated by disparate text prompts. We leverage a pre-trained latent diffusion model discussed in Chapter 2 and use the cross-attention layers to generate sound morphs. Using this method, we can smoothly control the semantics in the generated morph using simple fader-like controls. We evaluate our method objectively using text-audio similarity metrics and subjectively using perceptual listening evaluations.

[1]Based on the article pre-print:
Kamath, P., Gupta, C., Nanayakkara, S. (2024). MorphFader: Enabling Fine-grained Semantic Control for Text-to-Audio Morphing through Fader-like Interactions. *(Currently Under Review)*

4.1 Introduction

Diffusion-based [130] text-to-audio (TTA) models have recently exhibited remarkable capabilities in generating sound effects using guidance from text prompts [131–133]. The semantic sound space generated by TTA models is a productive avenue for novel sound exploration and, thus, for developing creative support tools (CSTs) [17] for sound design. However, their capabilities for gradually or smoothly morphing two sounds are relatively unexplored.

In this research, we introduce MorphFader, an interactive technique that utilizes TTA models to morph sounds generated by two different text prompts. Previous work in the image domain has demonstrated the efficacy of using cross-attention layers [113] to perform semantic edits to individual images [175–177]. We expand on these methods and develop a technique for interactive sound morphing and editing by granularly manipulating the cross-attention components using fader-like controls. We technically evaluate our method using text-audio similarity metrics and perform preliminary user evaluation by conducting perceptual listening tests.

In summary, our contributions include -

- A novel interactive technique to *smoothly morph* sounds generated by two disparate text prompts using pre-trained TTA diffusion models.
- A technique to interactively and *semantically emphasize or "weight"* certain word descriptors while morphing.
- A web-based interface to demonstrate our method's effectiveness in generating morphs using simple fader-like controls.

Our method can operate on any pre-trained TTA models without requiring extra training procedures or fine-tuning. Our current web-based interface demonstrates the morphing of two sounds in one dimension. In our future work, we outline interaction designs that utilize our method to morph sounds using 2D or 3D interfaces. A video demonstration of our method and the morphs generated can be listened to on our webpage[2].

[2] `https://purnimakamath.com/thesis-related/chapter_4/`

4.2 Related Work

4.2.1 Morphing in Audio

In videos, morphing is a common cinematic technique where an image of a person gradually transforms into another person in a series of smooth steps. When applied to audio, this technique can generate novel intermediate hybrid sounds and timbres, which can be useful in creating innovative musical compositions and fantastical sounds for sound design [31, 82]. However, most existing systems for morphing are limited to pitched instruments or vocal sounds [13, 31, 178–181] and do not perform well with ambient sound effects (e.g., dog barks) [115]. Previous works show that deep neural networks with specially designed labels could assist in generating smooth morphs [115, 116]. However, such models must be trained or fine-tuned on a small, targeted range of sounds, which limits their applicability outside those sound types.

Techniques for audio morphing can be broadly categorized into two - (1) *dynamic morphing* [182], where the source sound gets continuously transformed to the target sound over some time t, and (2) *repetitive morphing* [183] (also called as *stationary* [182], or *cyclostationary* [181] or *static* [183] morphing in the literature), where a series of intermediate sound morphs are generated, with each progressively containing more features of the target sound and fewer of the source sound.

In our work, we use existing pre-trained diffusion models capable of generating a wide variety of sound effects and develop a technique to generate morphs without additional training or fine-tuning. Further, we adopt the repetitive morphing paradigm to morph sounds generated by two text prompts. This helps us generate novel intermediate hybrid sounds and timbres that, at times, can generate fantastical sounds at each morph step.

4.2.2 Interacting with Generative Models using Text

Recently, diffusion-based text-to-audio (TTA) models have democratized how we generate sounds using AI models. Sound designers of all experience levels can use natural language to leverage AI models in their creative work. In contrast, previously, the sound generation relied on specialized labels designed by the AI model's developers [13, 79, 80] or leveraged the emergent properties of the latent space of the model [78, 184]. Although such models enable building steerable interfaces for CSTs [15, 16, 48, 81, 82], they do not scale well to large datasets compared to TTA models.

For audio, TTA models can be categorized as those that generate music [185–187] and others that generate sound effects, such as AudioLDM [132] and TANGO [131]. Currently, using a text prompt is the only way to interact with these TTA models. In the image domain, for text-to-image (TTI) models, [175–177, 188] outline methods to enhance creativity support offered by text by providing means to either manipulate prompts during the diffusion process or enable additional creative control, such as the use of image sketches [189]. For audio, models such as DITTO [134] enable control over a pre-trained diffusion model by optimizing the diffusion process during inference to achieve an additional creative goal in conjunction with text. While such methods enable building interactivity with the underlying model for better control, their potential for the creative task of morphing two or more sounds has not been explored.

Inspired by these approaches to induce creative control into pre-trained text-based diffusion models, in our work, we enhance interactivity with a pre-trained TTA model beyond simply using text prompts to generate audio morphs. By controlling the interpolation between two prompts through diffusion, we generate novel perceptually intermediate sounds that simultaneously resemble both the source and target-prompted sounds.

4.3 Proposed Method

At the core of our method is a pre-trained text-to-audio (TTA) latent diffusion model (LDM) [137], such as AudioLDM [132]. In Chapter 2, we briefly discussed diffusion models. This section provides a deeper background on LDMs and a further understanding of cross-attention components, which form the basis of our method. Subsequently, we outline our MorphFader method, which uses these components to morph and semantically word-weight two or more sounds.

4.3.1 Latent Diffusion Models

Diffusion models [130] belong to the class of generative AI models that learn to denoise a spectrogram through a series of steps to generate high-quality sounds. While diffusion models generally work directly on the spectrogram representations, LDMs, on the other hand, work towards denoising the latent vector representations of a pre-trained Variational Autoencoder (VAE) [190].

In Figure 4.1 (a), we show a drilled-down schematic of a denoising U-Net [135] for one step of the LDM-based diffusion process. The diffusion process accepts a randomly sampled

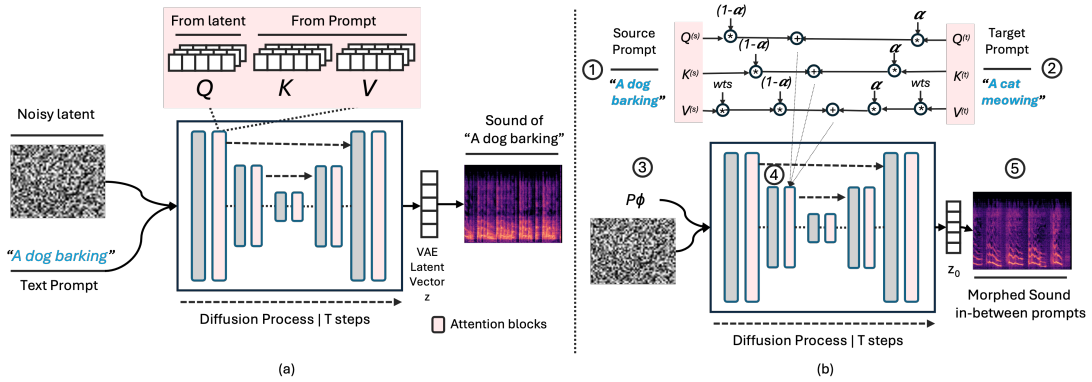


FIGURE 4.1: Schematic outlining the diffusion process and our method. (a) The diffusion process accepts a randomly sampled noise vector and a text prompt. The cross-attention layers in pink are responsible for injecting the text prompt embeddings into the generative model. (b) shows the schematic for our method. A user first generates a sound for the source prompt ① and target prompt ②. We intercept the **Q**, **K**, **V** matrices for both prompts. While generating the morph ③, we inject the interpolated matrices ④ to generate the resulting hybrid sound⑤.

noise vector and a text prompt. Diffusion occurs iteratively in T steps to generate the denoised latent vector z. This latent vector is decoded to a spectrogram using the VAEs decoder network. The spectrogram is converted to an audio waveform using a vocoder [109]. Note that the details on the various aspects of the diffusion process which we do not modify in our method - such as the pre-trained VAE's encoder and decoder, the diffusion forward noising process, the vocoder, as well as the text encoding process - have not been shown in this figure for brevity.

In each step of the denoising U-Net are a series of attention layers [113] (shown in pink in Figure 4.1). More specifically, these are cross-attention layers, where each word in the text prompt "attends to" or affects a specific semantic of the generated sound. For instance, a text prompt "a dog is barking" differs from the prompt "a dog is barking with reverb" in that the latter also pays "attention" to the part of the spectrogram that adds reverb to the generated sound. TTA models use cross-attention layers to inject the text prompts into the generative process. More formally, the components of an attention layer are called query **Q**, key **K**, and value **V**. Cross-attention is formalized as -

$$Cross\text{-}Attention(\mathbf{Q},\mathbf{K},\mathbf{V}) = \overbrace{Softmax\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)}^{attention\ map}\mathbf{V} \underbrace{\qquad\qquad}_{cross-attention\ matrix} \tag{4.1}$$

Where matrix **Q** is the embedded noise vector, and matrices **K** and **V** are embedded vectors of the text prompt (all shown in Figure 4.1 (a)). And d is the dimension of the dot product. The $Softmax$ output of the dot product between **Q** and **K** is referred to as an *attention map*. The dot product of the attention map with **V** is referred to as the *cross-attention matrix*. This cross-attention matrix contains the semantic information from the text prompt and is used to update the spectrogram through the diffusion process. In our work, we focus on manipulating the components of the attention matrices, namely **Q**, **K**, and **V**, on generating morphs and semantically scale the emphasis of words in prompts during morphing.

4.3.2 MorphFader

The intuition behind our method is that the components of the cross-attention matrices carry information concerning the semantic similarity between the text prompt and the generated sounds. By "weighting" or scaling these components, we can semantically emphasize the presence of a descriptor in the generated sound. Similarly, we can generate perceptually plausible intermediate sound morphs by continuously interpolating between the cross-attention components of two prompts.

Our method and algorithm are outlined in Figure 4.1 (b) and Algorithm 2. Say we want to generate a morph between two text prompts - a *source* prompt such as "A dog barking" and a *target* prompt such as "A cat meowing." We first generate the **Q**, **K**, and **V** matrices individually through the diffusion process for both these prompts. We then interpolate these matrices to generate the attention components for the morphed sound. As shown in Algorithm 2, we can interactively control the level of morph or interpolation using a scalar value α, where $0 < \alpha < 1$. This interpolation operation between the matrices occurs at each layer of the U-Net. The layer notations in Algorithm 2 are not indicated for brevity. As α changes from 0 to 1, the morph slowly changes from the source to the target sound.

Furthermore, through MorphFader, we can semantically increase or decrease the emphasis of certain word descriptors in a prompt. For instance, for the prompt "A dog is barking with reverb," by simply multiplying the matrix **V** with a weight vector, we can semantically increase or decrease the "reverb" in the resulting sound.

$$\overline{\mathbf{V}} = \mathbf{wts} \times \mathbf{V} \tag{4.2}$$

Algorithm 2: Generate Morph

Input:

$\mathcal{P}^{(s)}$, $\mathcal{P}^{(\tau)}$ are source and target source prompts;
$\mathcal{P}^{\phi}$ is unconditional prompt (classifier free guidance);
α is the interpolation level. Where $0 < \alpha < 1$;
s is the random seed; DM is the diffusion process;
$\mathbf{wts}^{(s)}$, $\mathbf{wts}^{(\tau)}$ are word weights for each prompt;
$\mathbf{z_T}$ is initial noise vector;

Output:

$\mathbf{z_0}^{(morph)}$: the denoised latent to generate the morph;

Function:

$\mathbf{z_t} \leftarrow \mathbf{z_T}$;
for $t \leftarrow T, ..., 1$ **do**
// Intercept relevant matrices.
// Do for every layer. Layer annotations are not included for brevity.
$\mathbf{z_{t-1}}^{(s)}, \mathbf{Q}_t^{(s)}, \mathbf{K}_t^{(s)}, \mathbf{V}_t^{(s)} \leftarrow DM(\mathbf{z_t}, \mathcal{P}^{(s)}, t, s)$;
$\mathbf{z_{t-1}}^{(\tau)}, \mathbf{Q}_t^{(\tau)}, \mathbf{K}_t^{(\tau)}, \mathbf{V}_t^{(\tau)} \leftarrow DM(\mathbf{z_t}, \mathcal{P}^{(\tau)}, t, s)$;

// Optional: Apply word weights. See equation 4.2.
$\overline{\mathbf{V}}_t^{(s)} \leftarrow \mathbf{wts}^{(s)} \times \mathbf{V}_t^{(s)}$;
$\overline{\mathbf{V}}_t^{(\tau)} \leftarrow \mathbf{wts}^{(\tau)} \times \mathbf{V}_t^{(\tau)}$;

// Interpolate relevant matrices.
$\mathbf{Q}_t^{(morph)} \leftarrow \alpha \times \mathbf{Q}_t^{(\tau)} + (1-\alpha) \times \mathbf{Q}_t^{(s)}$;
$\mathbf{K}_t^{(morph)} \leftarrow \alpha \times \mathbf{K}_t^{(\tau)} + (1-\alpha) \times \mathbf{K}_t^{(s)}$;
$\mathbf{V}_t^{(morph)} \leftarrow \alpha \times \overline{\mathbf{V}}_t^{(\tau)} + (1-\alpha) \times \overline{\mathbf{V}}_t^{(s)}$;

// Run diffusion process with interpolated matrices.
$\mathbf{z_{t-1}}^{(morph)} \leftarrow DM(\mathbf{z_t}, \mathcal{P}^{\phi}, t, s)\{\mathbf{Q}_t^{(morph)}, \mathbf{K}_t^{(morph)}, \mathbf{V}_t^{(morph)}\}$;
$\mathbf{z_t} \leftarrow \mathbf{z_{t-1}}^{(morph)}$
end
return $\mathbf{z_0}^{(morph)}$

Where $\mathbf{V}$ is the value matrix of the source or the target prompt, **wts** is the weight vector, and $\overline{\mathbf{V}}$ is the resulting semantically weighted value matrix.

Our approach of weighting $\mathbf{V}$ achieves similar goals to that of the semantic editing method outlined in [175]. In [175], authors propose to weight the full attention map for performing edits. Instead, we find it more computationally efficient to intercept, interpolate, and propagate individually weighted $\mathbf{V}$ components than the full attention matrix through each layer and per step of the diffusion process while morphing or word-weighting sounds.

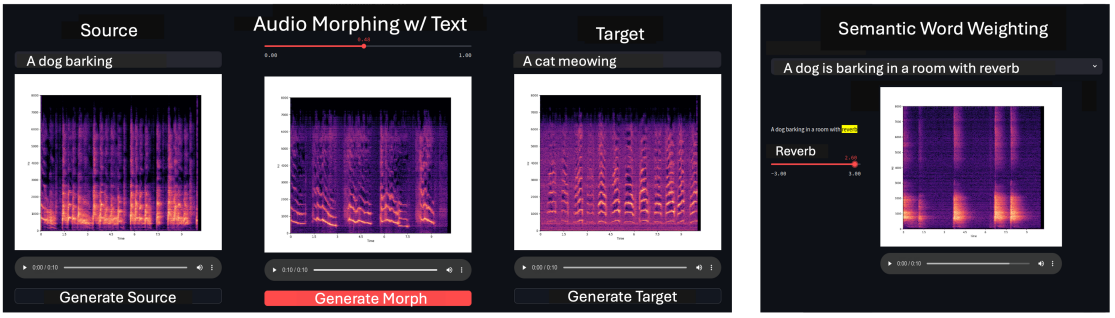


FIGURE 4.2: A screenshot of our web-based morphing interfaces. (a) For morphing using text prompts, a sound designer can individually generate a source sound, such as that of "A dog barking", and a target sound, such as that of "A cat meowing" shown in the interface's left and right columns, respectively. The generation of the audio morph between the two sounds at the center of the screen can be controlled by using a fader control that generates novel intermediate sounds with features resembling both the source and the target sound. (b) For semantically weighting words: The word in the text prompt that is being "weighted" or emphasized is highlighted in yellow. The scalar weight for this word is changed interactively using a fader control.

The interpolated matrices are then injected into the diffusion process during morph generation to create the final morphed latent vector $\mathbf{z_0}$. This vector generates the morphed sound using the VAE decoding process. Thus, by interpolating between the attention components of the two prompts, we can generate fantastical animal vocalizations, such as a morph between a dog's "bark" (source) and a cat's "meow" (target).

4.4 Experiments

In this section, we outline the details for implementing our method, the metrics used to objectively evaluate it, and the experiments conducted for evaluation.

4.4.1 Datasets

We sourced text prompts from a dataset called *AudioPairBank* [30] to evaluate our morphing technique. The AudioPairBank dataset contains over 1123 adjective-noun and verb-noun text-based concept pairs. It associates an adjective or a verb descriptor with nouns to create concept pairs such as a "barking dog" or a "squeaking chair," etc.

The authors of this dataset show that their curated list of semantic adjectives and verb-based concept pairs has many audio files associated with various online datasets, such as FreeSound (FS)[3]. The pre-trained TTA models in this work were trained on datasets that include sources such as FreeSound. Therefore, using prompts from this dataset will generate meaningful (i.e., within-distribution) source and target sound effects to evaluate our semantic word-weighting and morphing task.

4.4.2 Implementation Details

We implement our method over a pre-trained text-to-audio model AudioLDM [132]. AudioLDM is trained on large audio datasets such as AudioSet [152] in conjunction with text-based audio tags and captions. Specifically, we use the *"audioldm_16k_crossattn_t5"* model, which uses cross attention and is finetuned on FLAN-T5 [191] embeddings. Although we demonstrate the effectiveness of our method using AudioLDM, our algorithm can easily integrate with any LDM that uses cross-attention (such as TANGO [131] or Stable Audio [192]).

Our method is developed using Pytorch 2.0 [193]. We run our experiments on an RTX 2080 Ti 11GB GPU. All experiments are set with a constant random seed for the diffusion process, and the samples were generated by running the diffusion process for 20 steps ($T = 20$).

We implemented two web-based interfaces to demonstrate our method's ability to semantically weight words (Equation 4.2) and generate morphs (Algorithm 2) using Streamlit [194]. The controls to change the morph interpolation level α or word weights **wts** are actualized as a slider or a fader control on both interfaces. A demo video of our interfaces, our codebase, and Google Colaboratory notebooks demonstrating the interactivity within our interfaces can be found on our webpage [4].

4.4.3 Evaluation Metrics

Morphing is a creative task; therefore, the resulting morphs are typically assessed based on the subjective aesthetics of the sound. In this regard, Caetano et al. [195] formulate a few objective measures for evaluation, such as 'intermediateness' and 'smoothness', to evaluate morphed sounds. They define 'intermediateness' as a measure that evaluates the

[3]https://freesound.org

[4]`https://purnimakamath.com/thesis-related/chapter_4/`

perceptual plausibility of the generated sounds. They measure if the generated morphs are intuitive and perceptually 'in-between' the source and target. Further, they outline the 'smoothness' of a morph as the ability of the morphing method to gradually and linearly morph the sound from source to target.

To measure the perceptual plausibility of our morphs, we use objective audio quality metrics. Evaluating perceptual plausibility involves determining if the morphed sounds match the same distribution as the AudioSet [152] data. The underlying assumption is that if the objective metrics like FAD and FID were low, the morphed sounds would be closer to the real sound distribution and thus would sound plausible, like they belong to the real world. We further reinforce this objective metric by subjectively evaluating the sounds through listening tests measured using mean opinion scores. We articulate the plausibility of a good morph in a listening test by providing relevant examples of good morphs from existing literature, as outlined in the sections below. Finally, for smoothness, we use perceptual linearity metrics derived from text-audio similarity scores. The technical details of the metrics are as follows:

- ***Fréchet Audio Distance (FAD)***: We use the FAD [39] metric, which is the distance between the distributions of the embeddings of real and synthesized audio data extracted from a pre-trained VGGish [196] model, to evaluate audio synthesis quality as it is consistent with human judgments [39, 110, 197]. Lower values are better.

- ***Fréchet Distance (FD)***: This metric is similar to FAD, but uses state-of-the-art audio classifier PANN [151] for embeddings instead of VGGish. We use this metric as it was the primary metric used to evaluate AudioLDM [132]. Lower values are better.

- ***Inception Score (IS)***: This metric evaluates the quality and diversity of audio based on the embeddings from the InceptionV3 network [40]. Higher values are better.

- ***Smoothness of Morph***: We compute this metric by measuring the linearity of the change in the text-based similarity scores w.r.t the morph interpolation step α. We compute this linearity using the Pearson correlation coefficient (ρ). The text-based similarity score is computed using CLAP (or Contrastive Language-Audio Pretraining model [198]). The intuition behind this measure is based on the principle that if α changes by an X amount, the change observed in the resulting sound should be linearly proportional to X [82]. Higher smoothness values indicate higher intuitiveness or usability of the control.

- ***Mean Opinion Score (MOS)***: Mean opinion score subjectively evaluated via listening test to measure plausibility of the morphed sound. Higher values are better.

4.4.4 Baseline Selection

While selecting baselines for our experiments, we found that existing state-of-the-art toolkits, such as sound morphing toolbox [179], fail for non-pitched sounds [115]. Further, other deep learning methods, such as in [110], generate morphs for only a small targeted range of sounds, such as wind or water. To the best of our knowledge, there is currently a lack of methods to morph inharmonic general-purpose environmental sounds, such as those generated using TTA models. Thus, for our baseline comparison, we selected two handcrafted methods - (1) interpolating or mixing using raw audio waveforms and (2) morphing using engineered text prompts. For the first method, we continuously mix or interpolate between the raw audio waveforms of the source and the target sounds to generate the mix. For the second method, we engineered prompts such as "A morph between <Sound A> and <Sound B> where the level of <Sound A> is at <X>% and level of <Sound B> is at <(100-X)>%".

4.4.5 Experimental Details

We conduct five sets of experiments to evaluate our method. First, we conduct ablation studies to study the effect of each attention component and their respective combinations on the generated morphs. Subsequently, we conducted two experiments to evaluate the quality of morphs and another to evaluate the effect of morphing semantics based on different word types, such as adjectives or verbs.

4.4.5.1 Ablation Studies

In the first experiment, we conduct ablation studies to understand the effect of each participating attention component during the morphing process. That is, we systematically ablate or study the effect of adding or removing each individual $\mathbf{Q}$, $\mathbf{K}$, $\mathbf{V}$ component during the morphing process in Algorithm 2. We conduct seven sets of evaluations, such as using '$\mathbf{Q}$ only' to generate the morphs, or '$\mathbf{Q}, \mathbf{K}$' only, or '$\mathbf{K}, \mathbf{V}$' only, or other combinations thereof. We report this analysis using FAD, FD, IS, and Smoothness metrics.

Based on this, we select the best-performing attention component combination for all subsequent experiments in this chapter.

We randomly sample 100 source prompts from AudioPairBank to generate the sounds for the ablation studies experiment. For each source prompt sampled, we randomly selected another prompt as a target prompt (100 target prompts). We generated sounds for the source prompts and target prompts using AudioLDM. We then interpolated the attention components granularly using our method, using α in steps of 0.1 between the range $[0, 1]$ to generate 11 linearly morphed sounds for each source-target prompt pair. During each of the seven combinatorial ablation experiments, 1100 morphed sounds were generated for evaluation.

4.4.5.2 Morphing Quality Evaluation

This section outlines an objective evaluation experiment and a perceptual listening experiment to evaluate the morphs generated using our method in comparison with the two baselines.

(1) Objective Baseline Comparison: The aim of this experiment is to objectively compare our morphing method with the selected baselines. For this, we generated 1100 linearly interpolated samples using our method following the same procedure outlined in ablation studies. For generating sounds using waveform mixing baseline, we granularly interpolated the source and target prompted raw-audio waveforms to generate 1100 mixed sounds. For morphs generated using engineered text prompts, we crafted 1100 prompts by modifying the level values based on α in the prompt to generate interpolated morphs between the source and target.

We compute two sets of FAD and FD metrics for ablation studies and baseline comparison. We compute FAD-AudioSet and FD-AudioSet, where we evaluate the metrics for the intermediate morphs using 5000 randomly sampled audio files from the AudioSet [152] Evaluation dataset as reference. Additionally, we compute a second set of FAD and FD metrics for the intermediate morphs by using the source and the target sounds as a reference.

(2) Perceptual Baseline Comparison: In this experiment, we aim to perform listening test evaluations to subjectively analyze our method's effectiveness in generating morphs compared with the two baselines. We use mean opinion scores (MOS) for our analysis.

We created the audio morphs for our listening test by randomly sampling 20 source and target text prompt pairs from the AudioPairsBank. We generated morphs using our method and the two baselines for comparison in this test. The participants were presented with the source and target sounds and the three morphed sounds for evaluation in the test.

With approval from our university ethics board, we recruited 18 participants (10 male, 8 female) with a mean age of 28.27 years (SD=4.95 years) for our listening evaluation. Three participants had a background in music composition, and the remaining participants had no experience designing or creating sounds. No reimbursement was provided to the participants for this test.

The listening evaluation was administered online via Qualtrics and can be viewed on our webpage. Participants were emailed a link to the evaluation and could complete the test on their own. They were asked to complete the test in a single sitting and requested to use noise-cancellation headphones during the test. They were also asked to undertake the test in a quiet environment.

In the instructions for the listening test, we first asked participants to hlyellowlisten to a popular example of a good morph[5]. To better evaluate the morphs, we provided them with an instruction: "During the evaluation, ask yourself - *'how would I imagine a baby crying to the tune of a piano?'* And score the option closest to it higher than the rest". For each listening trial, we asked participants to listen to source and target sounds and score each of the three presented morphed sound examples for their perceptual plausibility on a scale from $[0 - 100]$.

4.4.5.3 Morphing Evaluation based on Word Types

This section outlines an objective evaluation experiment and a perceptual listening experiment to evaluate the semantic word-weighting and morphs generated using our method for different word types.

(1) Objective Evaluation: In this experiment, we aim to objectively analyze if different word types, namely adjectives, and verbs, have an effect on the plausibility or smoothness of the generated morphs (Algorithm 2). Further, we analyze if such word types have an effect on the plausibility or smoothness of the semantically weighted or emphasized sounds (Equation 4.2).

[5]We chose the sound of a baby crying morphing to piano from `https://www.cerlsoundgroup.org/Kelly/soundmorphing.html`

For analyzing semantic word-weighting sounds, we randomly sampled 100 adjective-based and 100 verb-based from the AudioPairBank and linearly modified the weights on the adjective or verb descriptors from $[-2, 3]$ to generate the sounds. This generated 6 linearly weighted sounds for each prompt to generate 600 semantically word-weighted sounds. Similarly, we sampled 100 adjective-based and verb-based prompt pairs and interpolated α to generate the morphed sounds to perform this evaluation. Our analysis for this experiment uses the smoothness metric, where we compute the text-based similarity scores for the generated sounds and compute its linearity w.r.t change in interpolation step (in steps of 1 between $[-2, 3]$ for word-weighting experiments and steps of $\alpha = 0.1$ for morphing experiments).

(2) Perceptual Evaluation: Finally, we conduct listening test evaluations to perceptually analyze the effect of word types on semantic word-weighting and morphing using our method.

We conducted two sets of listening evaluations: (1) we evaluated if the semantic word weight changes on adjective/verb descriptors were perceptible, and (2) we evaluated if the morphs generated with $\alpha = 0.5$ sounded perceptually "in-between" the source and target sound.

We randomly sampled 5 adjective-based and 5 verb-based prompts and modified the word weights as before for the word-weighting evaluation. The participants in the test were presented with two sounds: a reference sound and a sound under test. The sound under test was generated by modifying the weights on the adjective or verb descriptor of the prompt by $+1$ or -1.

For the morphing listening evaluation, we randomly sampled 4 adjective-based pairs and 4 verb-based prompt pairs and generated the morphs using our method. The participants were presented with three sounds: a source reference, a target reference, and a morphed sound under test. The morphed sounds were generated with interpolation level $\alpha = 0.5$.

We conducted another listening test (separate from baseline perceptual evaluation) and recruited 17 participants (8 male, 9 female) with a mean age of 28.47 years (SD=5.98 years) for our listening evaluation. Three participants had a background in music composition, and the remaining participants had no experience designing or creating sounds. The listening test was conducted using the Qualtrics survey and can be found on our webpage. No reimbursement was provided to the participants for this test. Overall, our listeners evaluated 20 questions on word weighting and 8 questions on morphing. We also collected some qualitative comments for each question from our listeners.

TABLE 4.1: Ablation Studies

	FAD (↓) AudioSet	FD (↓) AudioSet	FAD (↓)	FD (↓)	IS(↑)	Smoothness (↑)
Q,**K**,**V**	**10.81**	56.68	**0.25**	**5.14**	**5.98**	**0.61**
K,**V**	10.82	**56.61**	0.26	**5.14**	5.96	0.60
Q,**K**	17.53	94.71	7.48	50.79	1.80	0.30
Q,**V**	12.73	81.72	4.87	42.72	2.54	0.41
Q only	17.54	94.71	7.47	50.80	1.80	0.31
K only	27.09	134.35	14.74	96.78	1.00	0.30
V only	12.73	81.72	4.87	42.72	2.54	0.40

For word-weighting, the listeners were asked to evaluate the edited sound with the question *"How has the semantic property changed in the test sound compared to the reference?"*. This multiple choice question included "More", "Less", "No Change," and "Cannot say" as options. Similarly, for morphing evaluation, the listeners were asked to evaluate the morphed sound with the question, *"Does the in-between sample sound like a plausible morph between the source and the target?"*. All questions in this test were multiple choice questions with "Yes", "No," and "Cannot say" as options.

4.5 Results

4.5.1 Ablation Studies

We first study the effect of using **Q**, **K**, and **V** matrices individually while morphing using our method. Table 4.1 shows the FAS-AudioSet, FD-AudioSet, FAD, FID, IS, and Smoothness scores for each combination of the matrices. (↓) indicates that lower values are better. We find that using $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ and $\mathbf{K}, \mathbf{V}$ outperforms other attention component combinations. We use the best performing $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ for all experiments in the remainder of the paper.

TABLE 4.2: Baseline Comparison

	FAD (↓) AudioSet	FD (↓) AudioSet	FAD (↓)	FD (↓)	IS(↑)	Smoo- (↑) thness	MOS (↑)
Ours	10.81	56.68	**0.25**	**5.14**	**5.98**	**0.61** $\pm_{0.03}$*	**50.49** $\pm_{1.66}$
Wav.Mix	**9.13**	**52.19**	0.92	12.88	5.34	**0.61** $\pm_{0.07}$*	29.50 $\pm_{1.91}$
Prompting	11.73	67.10	1.53	18.21	5.20	0.34 $\pm_{0.03}$	45.26 $\pm_{1.90}$

4.5.2 Morphing Quality Evaluation

4.5.2.1 Objective Baseline Comparison

Table 4.2 shows our method's results compared with the selected baselines. (↓) indicates lower scores are better. Our method can generate better-quality sounds in terms of FAD, FD, and IS compared to the baselines. The mixes generated interpolating raw-audio waveforms demonstrate better FAD-AudioSet and FD-AudioSet scores than our method. Interestingly, our method and waveform mixing perform equally well when evaluated on the smoothness metric. A two-way t-test indicates there were no significant differences between the two scores (indicated by '*' in the table, $(t(N = 199) = 0.254, p = 0.799)$). However, by qualitatively listening and comparing the morphs generated by the two methods, we find that the sounds generated by our method generate perceptually novel sounding elements and are not simply an additive mix of the source and the target. We encourage our readers to listen to the sounds for comparisons on our webpage [6].

4.5.2.2 Perceptual Baseline Comparison

Table 4.2 shows the MOS from our listening test. (↑) indicates higher values are better. On average, the participants took 53.9 minutes (Min=14.61 mins, Max=179.7 mins, Std=49.35 mins) to complete the test. Participants rated morphs generated using our method as perceptually better as compared to mixes generated using raw-audio waveforms $(t(17) = 11.52, p < 0.05)$ as well as engineered prompts $(t(17) = 2.70, p < 0.05)$.

In [195], Caetano et al. articulate the difficulty in conducting a perceptual listening evaluation for morphs. No definite objective criteria exist while subjectively evaluating artistic outputs such as morphs. Such evaluations depend upon the participant's personal taste and aesthetics, which is difficult to measure and generalize. Therefore, we conduct perceptual listening evaluations for morphs in this research and augment this evaluation using objective metrics such as smoothness as shown in table 4.2. We encourage our

[6] https://purnimakamath.com/thesis-related/chapter_4/

TABLE 4.3: Analyzing Semantic Word-Weighting based on Word Types

	Smoothness of (↑) Word-Weighting	Plausibility of (↑) Word-Weighting
Adjective-based prompts	0.23 ± 0.03	0.55 ± 0.04
Verb-based prompts	$\mathbf{0.56 \pm 0.06}$	$\mathbf{0.68 \pm 0.06}$

readers to listen to the sounds on our webpage to gauge the effectiveness of our method in comparison with the two baselines [7].

4.5.3 Morphing Evaluation based on Word Types

4.5.3.1 Objective Evaluation

Evaluating Semantic Word-Weighting: Table 4.3 shows scores for the smoothness of word-weighting for both adjectives and verbs. A t-test indicates that there were significant differences between the text similarity scores before (unweighted; word-weight = 1) and after (with weights; word-weight = -2 and 3) the word-weighting for both types of descriptors ($(t(399) = -8.23, p < 0.05)$). This indicates that our method could perform edits to the sounds meaningfully while word-weighting both word types. However, in comparison with each other, weighting verbs in the text prompts was significantly smoother ($\rho = 0.56$) than weighting adjectives ($\rho = 0.23$). Figure 4.3 visualizes the smoothness of interpolation at each step between $[-2, 3]$ for verb and adjective descriptors. The dotted line shows the similarity score at word-weight = **1**, i.e., unweighted generation. Shaded regions show standard error of means computed by bootstrapping over 100 iterations.

This result has some implications when designing controls using adjective- or verb-based text prompts for audio generation. The authors of AudioPairBank note that listeners identify verb-based annotations in sounds better than adjectives. This is because verbs tend to be less subjective and more neutral than adjectives. For example, while annotating training datasets, there is less subjective debate about the presence of a barking sound (verb) than the size or type of the dog (adjective). This subjectivity may also be prevalent in the captions and tags of other large audio datasets. Thus, controls in creative support tools that edit semantics based on verbs would be more effective than semantics based on adjectives.

[7] `https://purnimakamath.com/thesis-related/chapter_4/baseline_linearity_comparison.html`

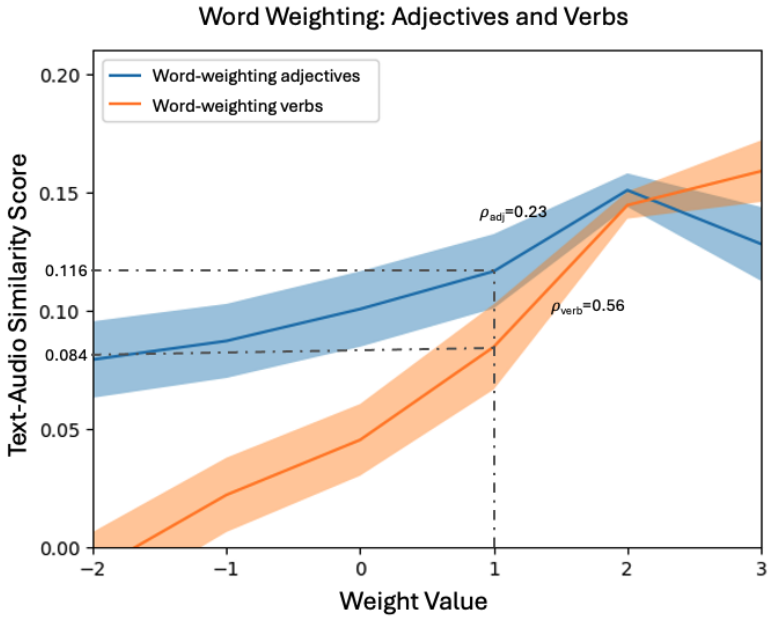


FIGURE 4.3: Plot showing how Text-Audio Similarity Scores change w.r.t weights values between $[-\mathbf{2}, \mathbf{3}]$ while semantic word-weighting adjective and verb descriptors in text prompts.

TABLE 4.4: Analyzing Morphing based on Word Types

	Smoothness (↑) of Morphing	Plausibility(↑) of Morphing
Adjective-based prompts	$0.46 \pm 0.18^*$	$0.55 \pm 0.10^*$
Verb-based prompts	$0.61 \pm 0.15^*$	$0.69 \pm 0.07^*$

Evaluating Audio Morphing: Table 4.4 shows scores for the smoothness of morphing when using prompts with adjectives ($\rho = 0.46$) and verbs ($\rho = 0.61$). We conduct t-tests to validate if the morphs generated using our method interpolate 'away' from the source prompt (indicated by a decrease in text-audio similarity) and 'closer' to the target prompt (indicated by an increase in text-audio similarity). For the source prompt, a two-sample t-test indicated that there was a significant decrease in the text-audio similarity scores after the morphing steps ($(t(99) = 4.33, p < 0.05)$ for adjective morphing; $(t(99) = 6.64, p < 0.05)$ for verb morphing). Similarly, there was a significant increase in the similarity scores for the target prompt ($(t(99) = -3.77,\ p < 0.05)$ for adjective morphing; $(t(99) = -8.45\ , p < 0.05)$ for verb morphing).

We conducted a t-test to validate whether morphs generated by interpolating prompts with verbs were 'smoother' than those generated by interpolating between prompts with adjectives. We found no significant differences between the smoothness scores for both

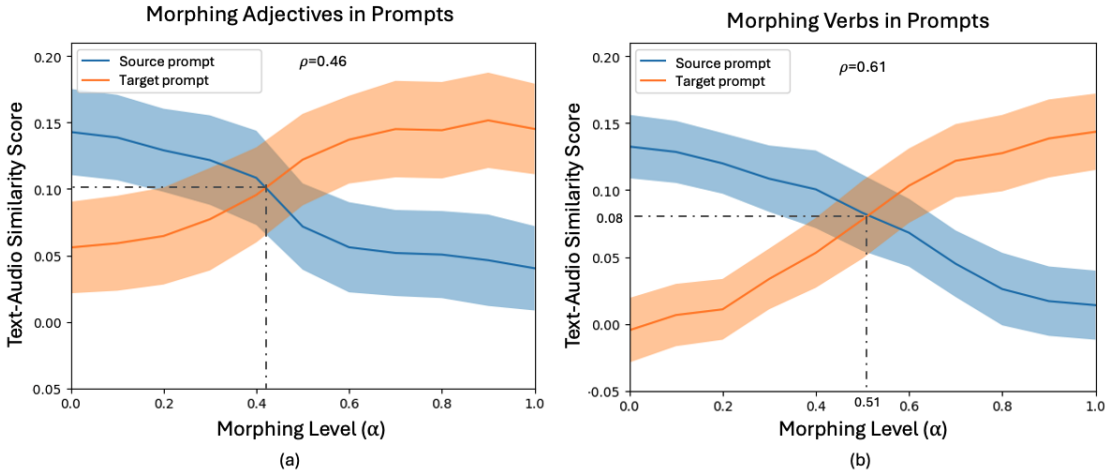


FIGURE 4.4: Plot for Text-Audio Similarity Scores per morphing level α while morphing (a) adjectives and (b) verbs in a sentence from source to target prompt. In both plots, observe how, as the morphing level increases, the similarity scores for the source prompt steadily decrease while increasing for the target prompt. Shaded regions show standard error of means computed by bootstrapping over 100 iterations. The dotted line shows the similarity score when both prompts are equally represented in the morphed sound.

prompts ($p > 0.05$). This indicates that our method can morph both adjective- and verb-based prompts equally well. Figures 4.4 (a) and (b) show the trends in similarity scores while morphing the prompts from source to target. In this figure, we observe that for both adjectives and verb morphing, the text-audio similarity score gradually decreases for the source prompt and gradually increases for the target prompt as α steadily increases from 0 to 1. The dotted lines indicate the scores when the two prompts are equally represented in the sound. Although there were no significant differences in the smoothness values for both types of prompts, the plots visually show that for adjectives, the plots flatten towards α values 0 and 1 more than verbs. This indicates that the range of effective control, the range between 0 and 1 where the morph effectively occurs, is lower for adjectives than for verbs.

4.5.3.2 Perceptual Evaluation

The participants in this listening test took an average time of 44 minutes to complete the evaluation (Min=26 minutes, Max=1 hour, 18 minutes, SD=17 minutes)

Evaluating Semantic Word-Weighting: For word weighting, listeners could correctly recognize verb-weighted changes with an accuracy of 0.68 ± 0.06 and adjectives with an accuracy of 0.55 ± 0.04. A two-sampled t-test revealed that there were significant differences between the accuracies for both descriptors ($t(16) = -2.39, p < 0.05$). Our

listeners could better evaluate semantic changes to verb-based descriptors than adjectives. This finding aligns with our objective technical evaluation in Section 4.5.3.1.

Evaluating Audio Morphing: For morphing, listeners evaluated the morphed sounds between adjective descriptors with an accuracy of 0.69 ± 0.07 and between verb descriptors with an accuracy of 0.55 ± 0.10. A two-sampled t-test revealed no significant differences between the accuracies for both descriptors ($p > 0.05$). Our listeners evaluated morphs created by interpolating between adjectives and verbs equally plausible or "in-between" the source and target sounds. This finding aligns with our objective technical evaluation in Section 4.5.3.1.

4.6 Discussion

The MorphFader algorithm introduced in this chapter uses a pre-trained TTA model to generate plausible-sounding, smooth morphs using two disparate text prompts.

Discussing Results from Ablation Studies: In Section 4.5.1, we performed experiments by ablating the attention components in a combinatorial way and found that both $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ and $\mathbf{K}, \mathbf{V}$ were best-performing combinations. As shown in Figure 4.1, the semantic information from the text prompt is usually represented by $\mathbf{K}$ and $\mathbf{V}$ in the diffusion process. Further, $\mathbf{Q}$ injects the randomly initialized latent noise vector, which, although impacting the generated sound's content, does not control the text-based semantics. Therefore, using $\mathbf{K}$ and $\mathbf{V}$, with or without $\mathbf{Q}$, ensures a greater representation of the semantic attributes of the two text prompts while morphing than the other combinations.

Discussing Results from Morph Evaluations: In Section 4.5.2.1, we objectively evaluated our morphing method with the two baselines of sounds generated using waveform mixing and engineered prompts. Although we qualitatively observe that our method outperforms the waveform mixing baseline, we find that both methods perform equally well when evaluated using objective smoothness metrics. This smoothness metric outlined in Section 4.4.3 is conceptually based on prior work in [115, 167, 195] and is implemented using text-audio similarity scores. When evaluating morphs generated by waveform mixing, the similarity score could be evaluating the presence of individual audio waveforms in the mix instead of the perceptual quality of the overall morph, resulting in a significant score. To address this, we supplemented the smoothness score with other objective metrics like FAD, FD, and IS during the evaluation to illustrate the effectiveness of our approach compared to simply mixing two waveforms. Therefore, we argue for

the need to develop new metrics that can be used to evaluate the perceptual smoothness of morphs, instead of relying on text-based similarity scores as used in this study.

Discussing Results from Semantic Word-Weighting based on Word Types: In Section 4.5.3.1, we evaluated our method of semantic word-weighting prompts with adjectives and verbs using metrics such as smoothness and plausibility of the generated sounds. We found that word-weighting verbs generated smoother interpolations than weighting adjectives. The authors of the AudioPairBank [30] note that human annotators are usually better at identifying actions or verbs in sounds than adjectives. Such subjectivity might be prevalent in all human-annotated datasets (including AudioSet). Therefore, we argue that in most current TTA models, fine-grained editing of verb-based semantics in a sound will be more effective than editing adjectives (as discussed in both Section 4.5.3.1 and Section 4.5.3.2). Therefore, when designing text-based controls for sound design, it is important to consider that verb-based controls may be more usable and identifiable than adjective-based controls.

Discussing Results from Morphing based on Word Types: In Section 4.5.3.1, we compared our method for morphing prompts with adjectives with that for verbs. We found that using our method, we could generate smooth and perceptually plausible morphs for both word types. However, in Figure 4.4, we visually observe that for adjectives, the plots flatten or saturate towards extremely lower and higher values of α more than that for verbs. This indicates that verbs have a greater range between 0 and 1 where effective morphs occur as compared to adjectives. This wider range of effective control for action-describing verbs could be because of human annotator subjectivity prevalent in audio dataset annotations as observed by [30]. Therefore, when using text-based controls for morphing, there will be a greater range of opportunities to explore the design space between verb-based prompts for novel sound discovery than adjective-based prompts.

Opportunities for morphing in modalities other than audio: This chapter discussed how we can generate morphs by exploring the sound space between two text prompts. While this idea was designed specifically for audio, it can be easily extended to designing CSTs for other modalities, such as text. For instance, we interpolate between the cross-attention components generated by the two text prompts for morphing sounds. This method can perhaps be applied to morph the style of the text content generated by a language model. By first generating an attention component for a particular style, say Shakespeare's way of writing poems, we can morph or interpolate any generated text "towards" or "away from" Shakespeare's writing style. Similarly, we could follow this method to modify attention components to emphasize or remove emphasis from (as we

perform semantic word-weighting) a particular affect, emotion, or feeling in the content during generation.

4.7 Summary

This chapter introduced MorphFader, an interactive technique for morphing sounds generated by pre-trained text-to-audio (TTA) models. Our method uses fader-like controls to intercept and interpolate the components of the cross-attention layers within the diffusion process. With no additional training or fine-tuning, our method generates smooth sound edits and perceptually plausible morphs between sounds generated by different text prompts. We validated our approach objectively using text-audio similarity metrics and subjectively through listening evaluations. Through this work, we provided novel interactive ways to explore the semantic sound space generated by TTA models for designing sounds.

Chapter 5

Perceptually Evaluating Descriptive Qualities of Sounds Using Visual Metaphors

Chapter Synopsis

In this chapter[1], we address **RQ3**. Novel AI-generated audio samples are evaluated for descriptive qualities such as the smoothness of a morph using crowdsourced human listening tests. However, the methods to design interfaces for such subjective listening experiments and to effectively articulate the descriptive audio quality under test receive very little attention in the evaluation metrics literature. In this chapter, we introduce novel visual constructs to design interfaces to evaluate the descriptive qualities of sounds generated using deep neural networks. Furthermore, we highlight the importance of framing and contextualizing a descriptive audio quality under measurement using such metaphors. Using both pitched sounds and textures, we conduct two sets of experiments to investigate how the quality of responses varies with audio and task complexities.

[1]With minor modifications from:
Kamath, P., Li, Z., Gupta, C., Jaidka, K., Nanayakkara, S., & Wyse, L. (2023). Evaluating Descriptive Quality of AI-Generated Audio Using Image-Schemas. In Proceedings of the 28th International Conference on Intelligent User Interfaces. IUI '23: 28th International Conference on Intelligent User Interfaces. ACM. doi:10.1145/3581641.3584083

Our results show that, in both cases, we can improve the quality and consensus of AI-generated audio evaluations by using visual constructs. Our findings reinforce the importance of interface design for listening tests and stationary visual constructs to communicate temporal qualities of AI-generated audio samples, especially to non-expert listeners on crowdsourced platforms.

5.1 Introduction

Generative algorithms aim to generate novel audio that matches naturally occurring sounds in their descriptive qualities such as realism, naturalness, or plausibility of the sound [13, 32–36]. Recent studies have emphasized the need for better perceptual evaluation techniques for audio [199, 200]. Automated objective metrics validate audio quality by measuring concepts that can be statistically represented, such as lack of distortion or noise [37–44]. Such metrics are faster to evaluate, but fail to find meaningful differences between descriptive perceptual measures such as realism or goodness of a morph. Consequently, such perceptual qualities are evaluated using subjective listening tests.

Listening tests aim to measure the perceptual quality of audio samples with respect to a ground truth or each other. In-person listening tests require a considerable amount of the researcher's time and effort and are expensive to set up. Thus, there is an increasing push within the audio deep learning community to move towards crowdsourced platforms such as Amazon Mechanical Turk (AMT) to conduct these tests. Platforms such as AMT are fast-paced, task-based marketplaces where participants optimize their time on a task. Thus, concise communication of the task instructions and the audio quality under evaluation becomes increasingly important. AI-generated audio is typically evaluated on quality concepts for sound progression, such as the quality of a morph (or how two sounds are interpolated with each other). While such concepts easily translate back to the ability of the algorithm to disentangle or interpolate within its latent space smoothly, non-experts on crowdsourced platforms may find such technical jargon difficult to understand.

Furthermore, for sounds that are not easily recognizable, such as multi-event, noisy audio textures [36, 92, 201], an ideal audio quality description should explicate the complexity observed in the sound space in a human-understandable way. Outlining such complex qualities verbosely using language makes for lengthy task instructions, which reduces participant interest in such tasks [83] and affects the overall quality of responses. In contrast, for example, image annotation or evaluation tasks often require only a simple 'glance-and-click' action [29]. Our aim in this paper is to assist novice listeners in understanding descriptive audio qualities under evaluation by using visual constructs instead

of language or words and, in turn, help AI researchers collate better and more meaningful responses from such listening tests.

The design of a typical listening test interface involves listening to two or more sounds in comparison to each other or with respect to a reference. As the number of sounds increases (for instance, a MUSHRA test [84] sometimes involves listening and comparing up to 12 sounds with each other), the demands on the listener's audio memory also increase, thus increasing the task's complexity. Recent human computation research on crowdsourcing shows that as task complexity increases, the quality of responses decreases, and participants more frequently abandon such tasks or submit poor quality responses [83, 85]. Thus, another aim of this paper is to design intuitive interfaces using visual constructs to minimize audio task complexity.

Using the metaphors of image-schemas is a promising avenue to visually explicate the descriptive quality of audio and design intuitive interfaces for listening tests. Image-schemas [202] are recurring structures and patterns of our basic sensory-motor experience grounded in our embodied interaction (e.g., walking) with our environment. Our inherent sensory-motor capacities of perceiving space and orientation are employed to understand abstract concepts and perform abstract reasoning. For example, when musicians talk about a composition "..it moves from G to A minor to C.." they apply their sensory-motor understanding of forward-oriented movement to understand chord progressions. Our aim in this research is to apply such image-schemas to evaluating audio in crowdsourced settings.

We can thus formulate our research questions as follows:

RQ3.1 How effective are visual constructs such as image-schemas in communicating the descriptive qualities of AI-generated audio?

RQ3.2 How effective are task interfaces designed using visual constructs such as image-schemas in evaluating AI-generated audio?

We explore these questions by conducting two experiments with 220 participants across different conditions. In the first experiment, we compare the performance of image-schemas and language-based descriptions to articulate the descriptive audio quality of the goodness or smoothness of a morph. In the second experiment, we compare the performance of interfaces designed using image-schemas and language for the complex task of evaluating the perceptual linearity of control parameters. In both experiments, we investigate the effectiveness of each condition by measuring the quality of responses and consensus amongst participants. Through this paper, we aim to provide future

researchers working at the intersection of HCI and audio AI with novel intuitive representations of audio quality, with an application for obtaining crowdsourced evaluation of audio samples. In summary, the main contributions are:

- An application of visual constructs, image-schema, to perceptually evaluate AI-generated audio.
- An open-sourced configurable front-end framework ('Crowd-Eval-Audio') to set up different listening test workflows on AMT.
- Validation of the effectiveness of using image-schemas to conduct listening tests on a crowdsourcing platform.

We show that directional image-schemas assist in evaluating sound progression in AI-generated general audio better than language alternatives.

5.2 Related Work

5.2.1 Visual metaphors for audio

Image-schemas were first introduced by Lakoff and Johnson [203] and later elaborated in [202] as constructs to understand abstract concepts and perform abstract reasoning. Wilkie et al. [204] use the *container* and *source-path-goal* image-schemas (amongst others) while designing music synthesizing interfaces. This was done to improve the user experience and interactions for experienced musicians and novice producers by building intuitive interfaces and reducing the need to possess specialist signal processing domain knowledge. Inspired by this approach, this paper will use the *source-path-goal* image-schema to articulate and communicate the descriptive audio quality under test and design our listening test interfaces. Our approach employs our inherent understanding of motion to indicate how an AI-generated audio sample progressively interpolates, transitions, or morphs from the start to its end.

5.2.2 Task design and clarity of instructions

Researchers have rigorously studied the effect of the quality of task instructions on worker performance and the quality of responses in the context of image annotation and natural

language processing (NLP) tasks on crowdsourced platforms. Gadiraju et al. [85] quantify task clarity based on the measures of goal clarity (what is needed to be done in a task) and role clarity (outlining steps to complete it). They survey crowd workers for ratings on multiple tasks and find that while the intrinsic complexity of a task can be high, the cognitive load associated with it can be reduced by a well-structured task presentation. Additionally, workers reported long sentences and complex words as hindrances to task clarity and performance. Other researchers extend this idea by identifying clarity flaws in descriptions and building algorithms based on natural language processing to identify such flaws [205].

Similarly, Wu et al. [83] systematically investigate the relationship between descriptive metrics (related to the task instructions and descriptions) and prospective metrics (related to workers' task preferences, including worker confidence and enjoyment of the task). They find that though lengthy and descriptive instructions increase worker trust and accuracy in the responses collected, the uptake and, subsequently, worker interest in such instruction-heavy tasks may be low. A balance between task design, instructions, and the number of examples is needed to achieve better descriptive and prospective metrics. K. Chaithanya Manam et al. [86] formulate a multistage framework for crowdsourced task description refinement, where workers assist in creating high-quality instructions based on prior feedback and cross-worker collaboration. Huang and Wu et al. [87] emphasize the need to decompose complex tasks into sub-components split across the same or multiple participants, thus narrowing each participant's focus singularly to a sub-task to collect more meaningful responses.

While improving the clarity of task instructions should be the central aim of any crowdsourced quality evaluation research, our work looks further into interface design to simplify the task complexity and use different visual representations to better articulate the audio quality under test. Furthermore, most current research on task design and clarity of instructions focuses on annotation tasks or tasks related to natural language texts. This paper highlights the importance of a clear task design for descriptive audio quality evaluation. While there is extensive research that focuses on interface design and representation for audio annotation tasks [29, 206, 207], to the best of our knowledge, there is very little research that discusses the effect of using various intuitive task interfaces and audio representations for the evaluation of descriptive audio qualities.

5.2.3 Sound perception studies

In the audio domain, evaluating sounds using crowdsourced platforms is not new. To do this, researchers have extended ITU standards such as MUSHRA [84] to the web. Web-MUSHRA [208] brings the lab-based interface to the web utilizing the standard web audio features available currently on all modern browsers. Cartwright et al. [172, 209] show that web-based MUSHRA can be easily used to conduct listening tests on platforms such as AMT to detect intermediate impairments with results matching a lab-based test. Similarly, research on music similarity judgments [210–213] shows that pairwise comparisons for musical pieces on AMT lead to good results which are comparable to a lab-based setup even when participants in the test had no prior musical knowledge. While these studies provide a basis for meaningfully conducting listening tests on crowdsourced platforms, they usually analyze pitched sounds or music and do not delve into complex sounds such as audio textures. Furthermore, most studies look at either audio annotation tasks [214, 215], pairwise comparisons, or comparisons against a ground truth. In this paper, we look further into analyzing complex tasks to rank order audio samples with respect to references and each other. Additionally, while these studies outline methods to compare audio samples, they do not focus on meaningfully and concisely articulating task instructions or the descriptive quality under test.

Other studies on music tracking tasks showed that visualizing a parameter such as pitch in a live music performance enabled better tracking, especially amongst non-musicians [216]. Our research attempts to build upon this work to produce better responses from naïve listeners by visualizing the audio quality under test on AMT. Regarding tooling, multiple web-based software tools exist that can be used to conduct listening tests online. Some use ITU standards [208, 217] or behavioral experimental design [218], but none focus on visual constructs for audio. The audio-tokens toolkit [219] shares some features with the interfaces we build. While the authors of the toolkit have demonstrated its use in behavioral sciences using speech, its use in evaluating perceptual differences for AI-generated general audio needs further investigation.

5.3 Method

Our aim with this paper is to use visual constructs of image-schemas to explicate the overall temporal quality of audio in a stationary way (RQ3.1) and reduce task complexities arising in multi-sample comparisons by designing interfaces based on such constructs (RQ3.2). We further use recognizable pitched sounds and unrecognizable multi-event

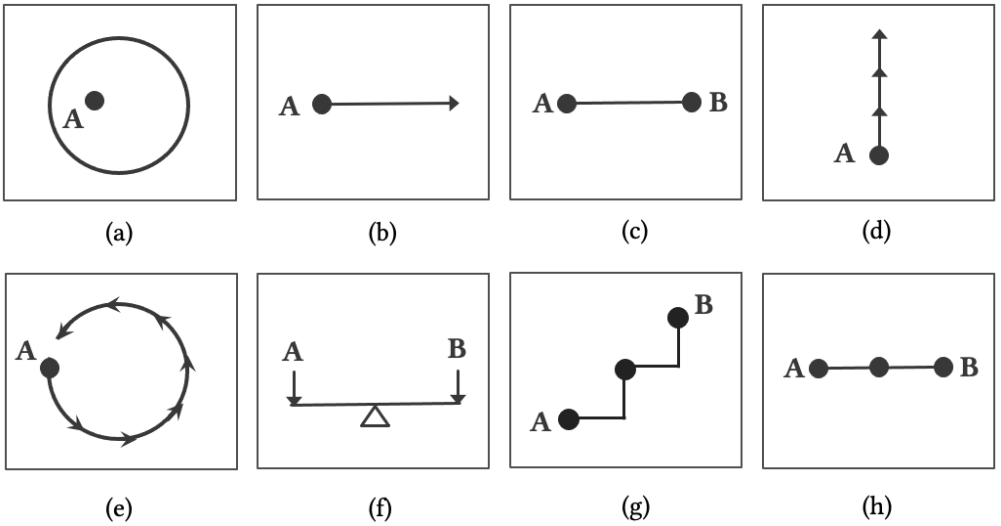


FIGURE 5.1: Some image-schemas visualizations which can be used to understand music based on [204, 220, 221] (a) Container, (b) Path, (c) Linkage, (d) Verticality or Up-Down, (e) Cyclic, and (f) Balance. (g) & (h) show two versions of the source-path-goal image-schema used in this paper.

noisy textures in both our experiments to investigate the ability of such constructs to reduce any complexities observed in the generated sound space.

5.3.1 Image-schemas

Researchers in the audio domain, particularly in music, have long studied and developed theories of musical meaning based on the metaphors of image-schema [204, 221]. Figure 5.1 shows some visualizations of common image-schemas used to understand musical concepts. The image-schema of *containment* can be used to understand chords and keys as containers, which can be connected to other chords/keys using different *paths* or *linkages*. Further, we generally perceive notes to go *up* or *down* in pitch. We can also perceive them to sound identical and yet be an octave apart, thus mapping the melodic notes space onto a *cycle* schema. Similarly, the image-schemas for *source-path-goal* can be used to understand a step-by-step melodic progression.

In this paper, we use the *source-path-goal* image-schema to articulate the descriptive audio quality under evaluation and to design listening test interfaces. Generating novel sound morphs is an important task in generative audio modeling [13, 222]. Morphs are generated by interpolating between two points in a generative algorithm's latent space or parameter space. Such morph progressions are evaluated for descriptive qualities such as perceptual linearity or interpolation smoothness. Thus, the *source-path-goal* is the most applicable metaphor in our context to indicate a morph progression starting at a

selected point in the latent or parameter space (*source*), progressing step-by-step (*path*) towards its end (*goal*).

5.3.2 Crowd-Eval-Audio framework

We conduct listening test experiments to explore RQ3.1 and RQ3.2 on Amazon Mechanical Turk (AMT). Current affordances on crowdsourced platforms such as AMT can be limiting for researchers conducting listening tests as:

- Experiments on AMT are usually set up and administered to participants via a simple web page. Such simple pages make hosting a step-by-step workflow required in an online listening test difficult. Workflows are important to guide participants through multiple steps in a listening test, such as the task overview, outlining the consent details, a hearing screening, or post-task surveys.
- The current task administration on AMT does not afford an experimental design where all participants undergo a fixed set of listening trials or a pre-determined number of randomized trials.
- The range of design elements and task widgets available on the platform is limiting. This can lead to unintuitive and poorly designed interfaces for experiments.

We thus build a front-end-based framework[2] for setting up configurable online listening test workflows on AMT. Our framework needs minimal infrastructure (no database installation required, etc.) and can be easily extended to conduct any experiment on AMT. We developed the framework as a single-page application (SPA) using VueJs, Javascript, HTML5, CSS gradients, and Web Audio API. It can be hosted on any content delivery networks (CDNs) and can be administered on AMT using the External Question API[3].

For the experiments in this paper, all resources were hosted on Amazon Web Services (AWS) CloudFront distribution network to enable fast downloads of audio files and other resources for crowdsourced participants worldwide. Furthermore, we integrate a participant logging service implemented using AWS Lambda and an Aurora serverless database. This logging was implemented to collate unique responses across all experiments by allowing individuals to attempt tasks from only one experimental condition.

[2]Code and sample experiments can be found at - https://github.com/pkamath2/crowd-eval-audio.

[3]https://docs.aws.amazon.com/AWSMechTurk/latest/AWSMturkAPI/

5.4 Experiment 1

In this section, we address RQ3.1 by assessing the effectiveness of using image-schemas to communicate the descriptive audio quality of the 'smoothness or goodness of a morph' of a Generative Adversarial Network's (GAN) [138] latent space in a listening test.

A GAN's latent space is smooth if the sound morphs generated by linearly interpolating between two randomly chosen points in the latent space are perceptually linear. The morphs are said to be non-linear or uneven if the interpolated sounds jump towards their chosen endpoints quickly instead of progressing gradually or if the morphs are of a third class of sounds not within the two chosen endpoints in space. Wyse et al. [116] showed that transforming this latent space using Self Organizing Maps (SOM) produces a more even and smoother morph. In this experiment, we conduct listening tests and ask participants to evaluate the smoothness of morph with and without the SOM ordering using either image-schemas or descriptive language.

5.4.1 Study Design

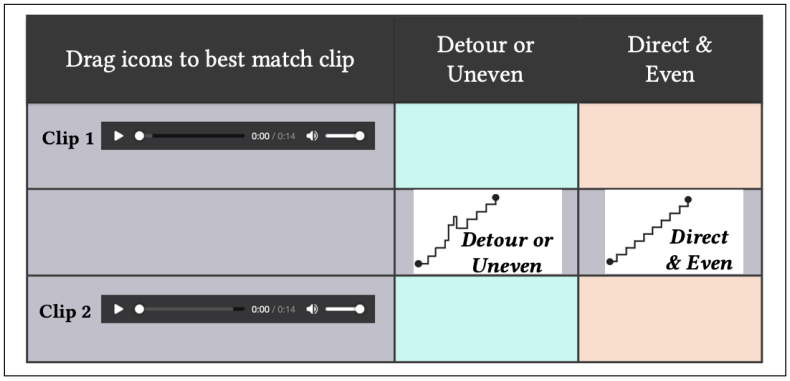
We follow a 2×2 between-subjects factorial design with the following factors -

- Visualization: Using language or image-schemas to articulate the audio quality under test.
- Type of sounds: Pitched sounds or noisy textures.

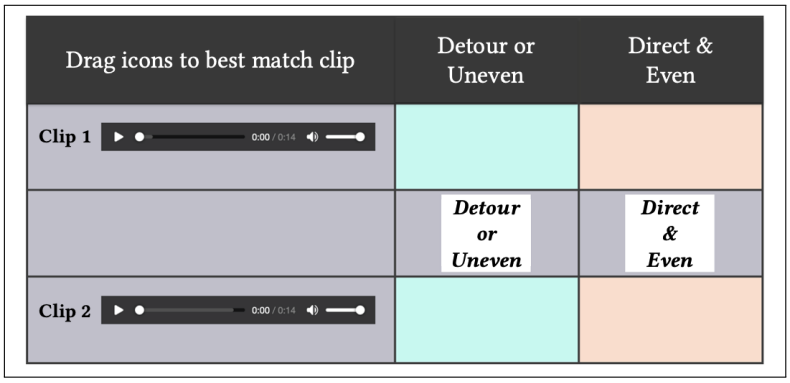
We recruited $N = 30$ participants for each of the 4 experimental conditions on AMT. Under a condition, each participant attempted 3 different trials for pairwise two-alternative forced choice (2AFC) comparisons with randomly selected sounds synthesized from different GANs. A preliminary test (with $N = 3$) was conducted on AMT to analyze and adjust the time allotted and payment for each condition.

5.4.2 Listening Test Interface

We developed a listening test interface for these experiments consisting of a simple 2AFC type of question. Two image-schemas based on the concept of *source-path-goal* were developed to visualize the descriptive quality of smoothness or goodness of a morph. An image-schema icon for ***"Direct & Even"*** conveys the idea of smooth and even morph.



(A) Screenshot of the smoothness of a sound morph experiment using image-schema



(B) Screenshot of the smoothness of a sound morph experiment using language

FIGURE 5.2: Experiment 1 listening test user interfaces.

A different icon for ***"Detour or Uneven"*** indicates that the morph is uneven or detours into a third type of sound. Each image-schema icon was associated with its respective text description ("Direct & Even" or "Detour or Uneven"). Listeners were requested to listen to the two sounds under comparison and drag/drop an image-schema icon near the requisite audio sample. When a participant dragged/dropped one icon toward one audio sample, the interface programmatically moved the other icon toward the other sample under test. This gave the listener visual feedback on how their choice affected the other sound under test in the 2AFC experiment. For the language set of experiments, we outline the descriptive qualities in text without the icons. The language-based interface was designed and implemented to ensure the same functionality as the interface with image-schemas. A schematic of the interface and icons used for both the image-schemas and language conditions are shown in Figure 5.2. The interfaces developed for this experiment, along with their audio samples, can be listened to on our website[4].

[4]https://purnimakamath.com/thesis-related/chapter_5/

5.4.3 Sound synthesis

To generate morphed sounds for the 2AFC comparisons, we train two GANs - one on pitched sounds from the NSynth dataset [79] and another using a noisy texture dataset [223]. The GANs were trained as outlined in [116]. We selected brass and reed instruments from MIDI pitch numbers $64 - 76$ from the NSynth dataset to train the GAN for pitched sounds. For textures, we selected 4 different classes of sounds from the texture dataset. To generate the morphs, we randomly selected two points, A and B, in the latent space and sampled 20 evenly-spaced points between them to generate the first clip. Next, the space between the two selected endpoints was remapped using the SOM (as in [116]) and resampled to generate the second clip. Each interpolated point in the latent space generates a stationary morph sample between A and B. Finally, all 20 generated samples are concatenated to create a single audio file presented to a listener for evaluation. Each audio sample created in this way is approximately 14 seconds in duration. We generated 3 sets of samples, with and without remapping, for the audio trials in this experiment.

5.4.4 Participants

The 120 participants recruited for this experiment were paid \$1.40 for completing the test. Participants were allowed to attempt our experiments if they had a 95% approval rate with over 1000 approved HITs. The mean task completion times were 7.27 ($SD = 3.26$) and 7.22 ($SD = 3.17$) minutes for image-schema and language-based conditions, respectively. Each participant was allowed to complete a task only from one experimental condition.

5.4.5 Procedure

Participants were requested to sit in a quiet place and use a pair of headphones during the experiment. They were first presented with a hearing screening as outlined in [172]. During the screening, participants were presented with two audio samples containing different tones generated at random frequencies between 55Hz and 10kHz and were asked to count the number of tones. Participants who completed the hearing screening and correctly estimated the number of tones were allowed to attempt the task. The hearing screening ensured that the participants were of normal hearing, were using a pair of headphones, and were in a quiet environment while taking the test.

Next, the participants were presented with the instructions for each condition and asked for consent. Subsequently, a listening test was presented to the participants depending on their assigned condition. All audio trials within each task were randomized to reduce any ordering effects. After completing the test, they were asked to complete a post-test survey, which included questions on their listening equipment and surrounding environment. They were also asked for comments on the complexity of the listening test.

5.4.6 Measures for evaluation

The measures outlined below build upon the analysis methodologies from [29, 201].

- **F-Score**: We use F-Scores as a quality measure for our experiments. F-Score is a harmonic mean of precision and recall used as a binary classification measure for this experiment.
- **Pairwise agreement**: We use participant agreement as another measure of the quality of responses in our experiments. We build upon the pairwise agreement outlined in [29] to measure the consensus amongst our listeners. This measure can be formalized as:

$$agreement = \frac{1}{N(N-1)} \sum_{i,j=1}^{N} F_{i,j}, \tag{5.1}$$

  Where N is the total number of participants within a condition and $F_{i,j}$ is the pairwise combined F-score between two participants i, j within that condition.
- **Test-retest reliability**: To estimate the minimum number of participants needed to obtain stable results for our experiments, we measure the test-retest reliability of our results. We randomly divided our participants into two groups and measured Pearson's correlation coefficient between vectors of quality versus condition for each group. Then, we repeated the procedure 1000 times for a range of sample sizes (from 2 to 30) and analyzed the reliability coefficient.

All score distributions, confidence intervals (CI), and standard errors of means are visualized and reported by bootstrap (1000 samples). For every bootstrap iteration, a set of participants equal in number to the participants who took longer than the average time to complete each trial were sampled with replacement.

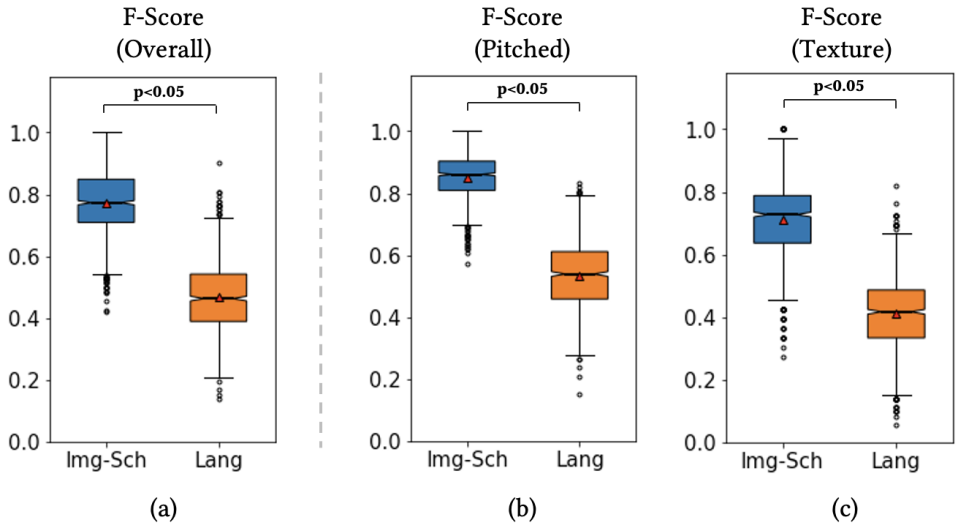


FIGURE 5.3: Results of Experiment 1 for pitched and texture sounds based on the type of visualization.

5.4.7 Results

The responses we collected via our experiments on AMT did not conform to the conditions of normality. We thus conduct non-parametric tests to analyze the results. We first test the stability of our responses using test-retest reliability measures. Our test-retest reliability coefficient for the pitched sounds condition increased from 0.39 at $N = 2$ to 0.95 at $N = 18$. For textures, the reliability coefficient increased from 0.24 at $N = 2$ to 0.95 at $N = 24$. Thus, $N = 30$ is sufficient for analyzing all conditions in this experiment.

Effect of visualization on quality of responses

For this experiment, Figure 5.3 (a) plots F-Score as a measure of quality for the experiment conducted using image-schema ($Mdn = 0.781$) and language ($Mdn = 0.462$) for both pitched and texture sounds. Figures 5.3 (b) and (c) plots F-Scores individually for pitched sounds ($Mdn = 0.860$ and $Mdn = 0.538$ for image-schemas and language conditions) and textures ($Mdn = 0.727$ and $Mdn = 0.416$ for image-schemas and language conditions) respectively. A Sheirer-Ray-Hare test for the quality of response comparison across visualization and sound types showed that visualizations had a significant effect on the overall quality of responses ($H(1, 118) = 20.62, p < 0.017$, adjusted for Bonferroni correction $\alpha = 0.05/3 = 0.017$). A post-hoc Mann-Whitney test found that image-schemas had a significant effect on the quality of responses for both pitched sounds ($U(N_1 = 30, N_2 = 30) = 674.5.0, p < 0.05$) and textures ($U(N_1 = 30, N_2 = 30) = 637.5.0, p < 0.05$). Therefore, for experiments with simple 2AFC tasks evaluating

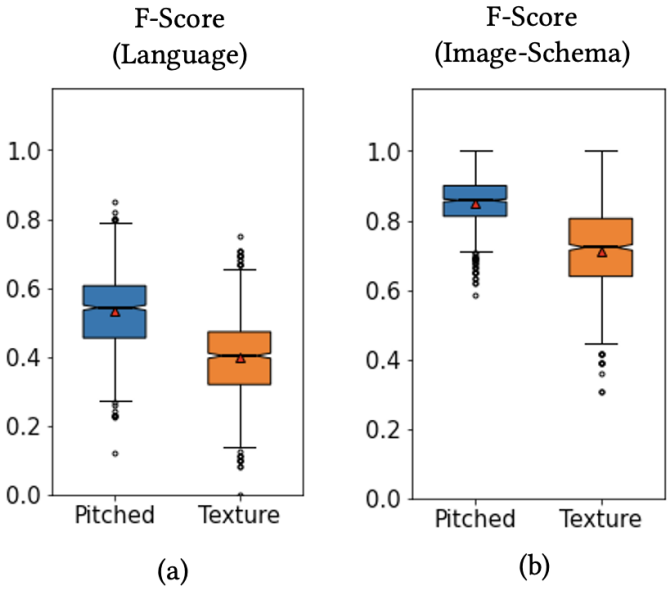


FIGURE 5.4: Results of Experiment 1 for language and image-schemas conditions based on sound type.

descriptive qualities of audio samples, using image-schemas significantly improved the quality of responses collected for all sound types.

Effect of visualization for sound types

Figures 5.4 (a) and (b) plot the F-Scores for pitched sounds and textures based on language and image-schema conditions. Interestingly, even though pitched sounds are generally considered more recognizable than noisy textures, participants evaluated both sound types comparably. There were no significant differences found in their evaluation using language ($U(N_1 = 30, N_2 = 30) = 522.5, p = 0.268$) or image-schemas ($U(N_1 = 30, N_2 = 30) = 491.0, p = 0.497$). Therefore, for simple 2AFC type of experiments, complexity in sound space does not significantly affect the quality of responses submitted in an online listening test within the same visualization condition.

Pairwise Agreement

Table 5.1 shows the median pairwise agreement between participants across trials in this experiment. The results indicate that experiments using image-schemas showed significantly better agreement as compared to language-based conditions.

To examine the effect of the number of participants on the overall agreement in this experiment, we plot the aggregated median pairwise agreement by simulating an increase

TABLE 5.1: Pairwise participant agreement for Experiment 1.

Sound Type	Experimental Condition	Median Agreement [95% CI]
Pitched Sounds	Image-Schema	0.897 [0.850, 0.944]
	Language	0.549 [0.502, 0.624]
Textures	Image-Schema	0.803 [0.707, 0.898]
	Language	0.384 [0.292, 0.515]

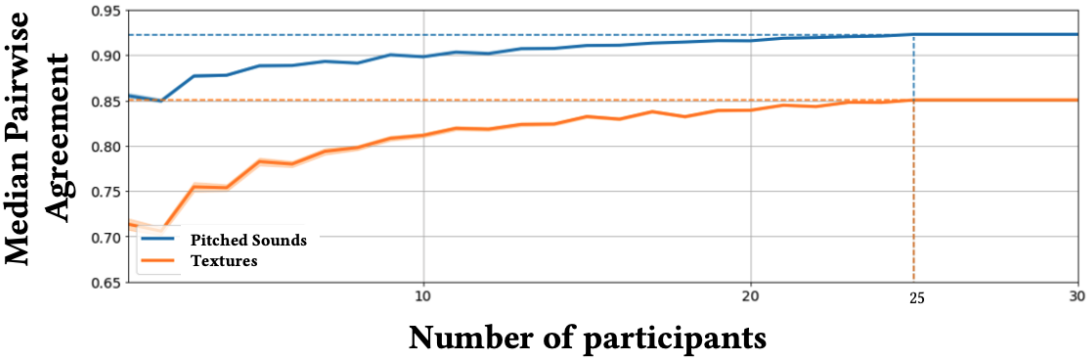


FIGURE 5.5: Median pairwise agreement for pitched sounds and textures under the image-schemas condition as the number of participants increases for Experiment 1. The dotted lines indicate the number of participants needed to reach full agreement for this condition.

in the number of participants for the image-schemas condition as shown in Figure 5.5. We adapt this method from [29]. First, we shuffle the order of the responses from all participants across all trials and then progressively add the responses to calculate the overall median agreement for each sound type and repeat the process 1000 times. The dotted lines show the number of participants needed for maximum consensus for each condition. The plots indicate that we need at least 25 participants to reach the maximum consensus for that condition in this experiment for both pitched sounds and textures. This implies that for simple 2AFC tasks, especially in the absence of ground-truth labels, the number of participants needed to evaluate sounds with a significant agreement does not depend upon the complexity of the sounds under test.

Relationship between task time and quality of responses

We study the effect of the time participants take to complete a task on the quality of responses submitted. For this, we first sort the participant responses in an increasing order of time taken to complete the trials. We then analyze the correlation between the time spent on the trials and the average quality (F-Score) of response. For pitched

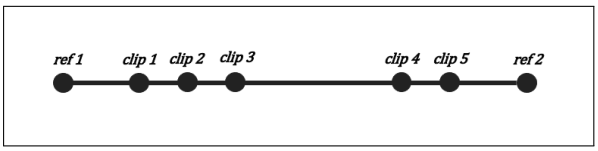
sounds, under both visualization conditions, the time taken to complete the task strongly correlated with the quality of responses submitted ($r(28) = .717, p < 0.05$ for image-schemas condition and $r(28) = .819, p < 0.05$ for the language condition). For textures under the language condition, time spent on task negatively correlated with quality ($r(28) = -.817, p < 0.05$). For the image-schema condition, these sounds moderately correlated with quality ($r(28) = .407, p < 0.05$). Therefore, participants who spent a longer time on a task could submit better responses for recognizable pitched sounds. For unrecognizable noisy textures, participants who spent a longer time on the task submitted lower-quality responses when using the language condition than the image-schemas condition. This implies that staying longer on the task evaluating textures might have increased the cognitive load and confusion amongst the participants, leading to poorer responses when using the language-based conditions, which was slightly alleviated when using image-schemas.

5.5 Experiment 2

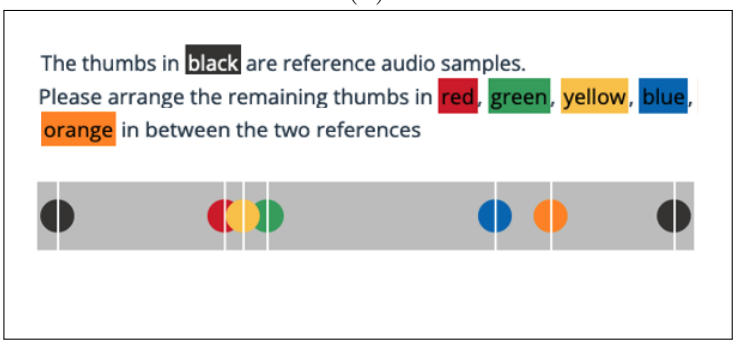
In this section, we address RQ3.2 by using image-schemas to design interfaces for a listening test. We conduct an experiment to evaluate the perceptual linearity of sounds generated by linearly varying control parameters of a generative model. Controllable or conditional synthesis is an important task in generative audio modeling, which typically comprises a deep neural network trained on audio data in conjunction with some conditional parameters. For example, generative models conditioned on pitches from different musical instruments can be explicitly controlled to generate different timbres for a pitch [13]. Generally, control parameters used to train such models are varied linearly in the parameter space. Their effect, however, in the generated audio is not always guaranteed to be perceptually linear [36, 167, 224]. There is thus a need to evaluate the perceptual sensitivity to the linear parametric variation of such AI-generated sounds.

5.5.1 Study Design

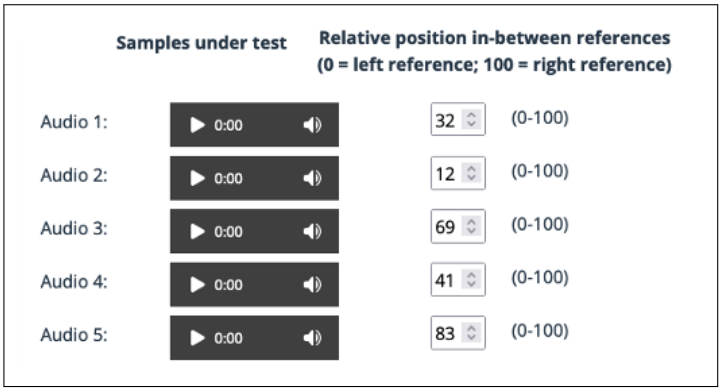
As with Experiment 1, this experiment applies the same recruitment criterion and follows a 2×2 between-subjects factorial design using the factors of visualization and type of sounds. Once again, a preliminary test (with $N = 3$) was conducted to analyze and adjust the time allotted and reimbursements for this experiment.



(A)



(B)



(C)

FIGURE 5.6: Experiment 2 image-schema design is shown in (a), and the screenshot of the implemented interface using image-schemas is shown in (b). A language-based interface was implemented as shown in (c).

5.5.2 Listening Test Interface

To conduct this experiment, we designed and implemented an interface to rank-order and proximally space some audio samples. This test aimed to present 7 short audio samples - 2 references and 5 sounds under test - and ask the listeners to order the 5 samples perceptually between the two references. For the image-schema condition, a listening test interface followed the *source-path-goal* schema as shown in Figure 5.6 (a). All audio samples were visually represented as thumbs along a *path* in between the *source* and *goal* reference sounds. Figure 5.6 (b) shows a screenshot of the user interface using image-schemas. Black thumbs represented the reference endpoints at the left and right extremes. Colored thumbs along a slider represented the remaining sounds under test. The participants were asked to hover their mouse pointers over the thumbs to listen to

the sound samples and to drag the thumbs along the slider to position them with respect to the static and immovable references as well as each other. The 'hover to listen' and 'drag to position' interactivity were implemented to reduce fatigue amongst listeners from having to click a web audio component to play each sound multiple times while positioning the sounds between the references.

Given the number of comparisons participants needed to perform, we split this task into 3 sub-tasks to mitigate its complexity. First, the participants were asked to listen to and position the sounds as discussed above. Then, they were allowed to listen to the 'arrangement' they created by clicking a button that played each audio sample in the order in which they were positioned along the slider. This step was intended to aid the participants in positioning the samples better by listening to them in an automated sequence. After this step, they were asked to listen to the arrangement again and fine-tune each sample's position based on the reference endpoints and their proximity to their neighbors. Participants were allowed to submit the task only after completing all steps. The interfaces developed for this experiment, along with their audio samples, can be listened to on our website[5].

The language-based interface, shown in Figure 5.6 (c), was designed and implemented to ensure the same functionality as the screens with image-schemas. Here, participants listened to audio samples presented using the HTML5 web audio interface and entered the ranks or distances using number drop downs between 0 and 100. The two rank-order and proximity based on distance 'arrangement' sub-tasks (just as in the image-schema trials) played the audio samples in an automated sequence based on the order indicated by the numbers in the drop downs.

5.5.3 Sound synthesis

We select two audio samples to explore RQ3.2 with varying sound complexities. The pitched samples were generated using the Syntex [225] DS_BasicFM_1.0 synthesizer which generates a frequency modulated sine wave governed by the algorithm $y[t] = sin(2\pi * cf * t + mI * sin(2\pi * mf * t)))$, where cf is center frequency, mI is the modulation index and mf is the modulation frequency. To create this dataset, we fix the modulation frequency and modulation index to $\sim 7.5Hz$ and 12.5, respectively and linearly vary the signal's center frequency between $\sim 330Hz$ and $\sim 660Hz$ (the left and right references, respectively). The center frequencies for the remaining 5 other samples were randomly selected between the two references. Each of these 7 audio samples was 2 seconds long.

[5]https://purnimakamath.com/thesis-related/chapter_5/

For texture samples, we used a recorded sound made by water filling a container, with the container's fill level as the control parameter guiding the progression of the audio samples. The left and right references were chosen with fill-level=0 (empty container) and fill-level=1 (full container). The five other samples under test were random fill levels between the two references.

5.5.4 Participants & Procedure

We recruited $N = 25$ participants for the 4 experimental conditions, resulting in 100 participants. They were paid $1.20 for completing the task. Each task had one perceptual ordering trial. The mean task completion times were 6.63(SD = 2.23) and 7.31(SD = 3.5) minutes for image-schema and language-based conditions. As in Experiment 1, participants were allowed to complete a task only from one experimental condition.

5.5.5 Measures for evaluation

We analyze the results from this experiment by first transforming the proximity or distance-based values captured from the participant's responses to ranked data. We use the measure of F-Score and agreement (formalized in section 5.4.6) on this ranked data to evaluate the quality of responses. In addition to these measures, we use -

- **Cosine Similarity Score**: The rank-based F-Score fails to account for the variance in the actual distance values captured (e.g., while a participant's response of [1,2,3,4,5] matches in rank to a ground truth of [11,28,32,49,51], it differs largely in the actual proximal distance/spacing captured). To capture this variability in distance, we treat each participant's response as a 5-dimensional vector and find its cosine similarity with the ground truth. Cosine similarity for two vectors is formalized as -

$$similarity\ score = \frac{x \cdot y}{\|x\|_2 \|y\|_2} \tag{5.2}$$

  where x and y are the participant's response and ground truth. The numerator is their dot product, and the denominator is a scalar product of their Euclidean (L2) norms.

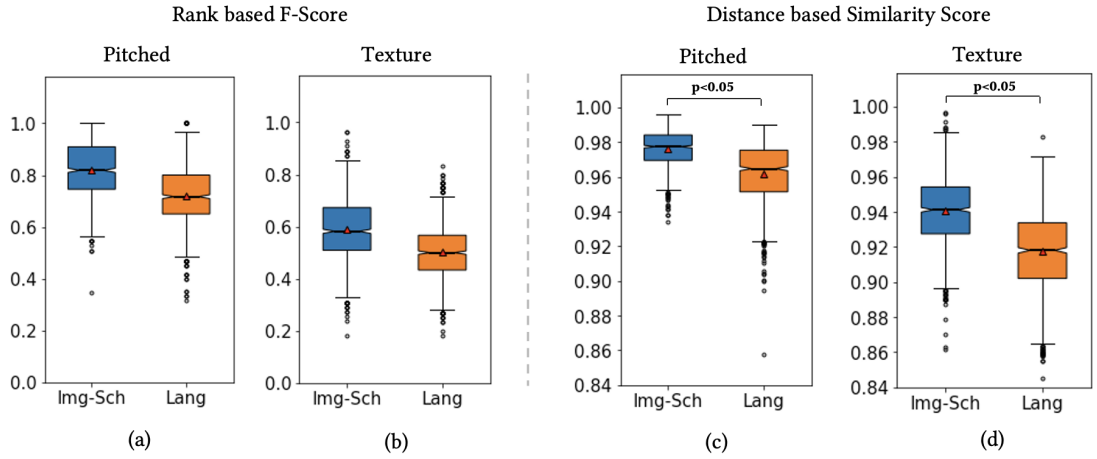


FIGURE 5.7: Results of Experiment 2 for pitched and texture sounds based on the type of visualization.

5.5.6 Results

As with Experiment 1, we use non-parametric tests to analyze our data for this experiment. Our test-retest reliability coefficient for the pitched sounds condition increased from 0.18 at $N = 2$ to 0.95 at $N = 22$. For textures, the reliability coefficient increased from 0.25 at $N = 2$ to 0.95 at $N = 18$. Thus, $N = 25$ is sufficient for analyzing all conditions in this experiment.

Effect of visualization on quality of responses

For this experiment, Figure 5.7 (a) and (b) plots aggregated rank-based F-Scores for the responses for each sound type. Overall, the experiments with image-schemas did not result in significant differences using rank-based measures ($Mdn = 0.82$ for pitched sounds and $Mdn = 0.58$ for textures) in comparison with language-based experiments ($Mdn = 0.72$ for pitched sounds and $Mdn = 0.5$ for textures). This implies that most participants in both image-schemas and language-based conditions could rank-order the audio samples correctly.

To further study the difference between the two conditions, we tested our hypothesis using distance-based similarity measures defined in section 5.5.5. Figure 5.7 (c) and (d) plots this similarity measure for each sound type. A Sheirer-Ray-Hare test for the quality of response comparison across visualization and sound types showed that visualizations had a significant effect on the overall quality of responses ($H(1, 98) = 7.13, p < 0.017$), adjusted for Bonferroni correction $\alpha = 0.05/3 = 0.017$). In post-hoc Mann-Whitney tests, both pitched sounds ($U(N_1 = 25, N_2 = 25) = 400.5, p < 0.05$) and textures

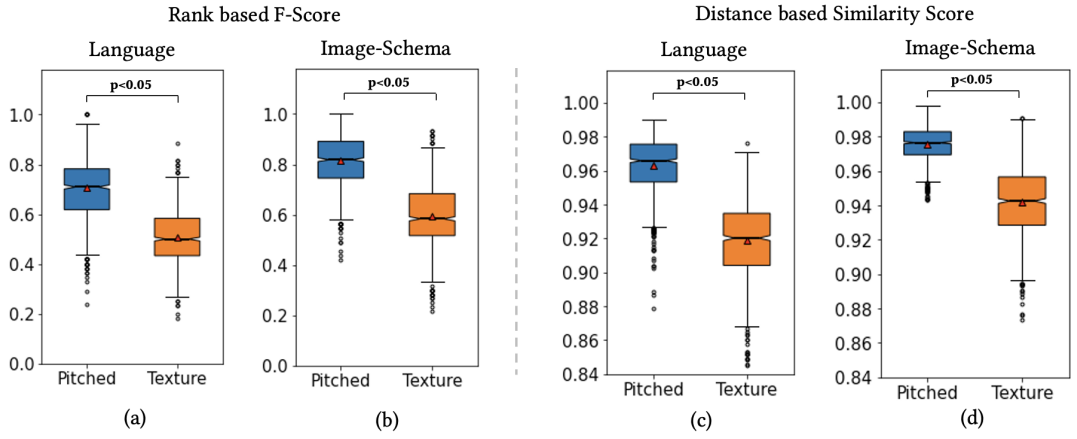


FIGURE 5.8: Results of Experiment 2 for language and image-schemas conditions based on sound type.

TABLE 5.2: Pairwise participant agreement for Experiment 2.

Sound Type	Experimental Condition	Median Agreement [95% CI]
Pitched Sounds	Image-Schema	0.896 [0.850, 0.941]
	Language	0.813 [0.731, 0.859]
Textures	Image-Schema	0.630 [0.476, 0.783]
	Language	0.498 [0.417, 0.651]

($U(N_1 = 25, N_2 = 25) = 410.0, p < 0.05$) report significant differences in evaluation when using image-schemas. Thus, interfaces designed using image-schemas were better at capturing differences in spacing or proximity of the sound samples than language-based interfaces.

Effect of visualization for sound types

Figure 5.8 shows the rank-based F-Score and distance-based similarity measures for pitched and texture sounds for both visualization conditions. Even though the texture sounds used in this experiment (water filling a container) were recognizable and not very noisy, pitched sounds were evaluated significantly better than textures when using both rank-based data ($U(N_1 = 25, N_2 = 25) = 404.5, p < 0.05$) as well as distance-based similarity scores ($U(N_1 = 25, N_2 = 25) = 430.0, p < 0.05$).

Pairwise Agreement

Table 5.2 shows the median agreement for each condition for this experiment. The conditions using image-schemas for both pitched sounds and textures reported significantly better agreement amongst participants than using language. $U(N_1 = 25, N_2 = 25) = 514.0$, $p < 0.05$ and $U(N_1 = 25, N_2 = 25) = 421.0$, $p < 0.05$ for pitched sounds and textures respectively. As in Experiment 1, we also analyze the effect of the number of participants on the overall aggregated agreement for the image-schema condition. The maximum consensus in this experiment is reached for pitched sounds by aggregating agreement from 9 participants. For textures, we needed 18 participants for maximum consensus. Therefore, for complex tasks, the number of participants needed for consensus varies depending upon the type of sounds under test. Recognizable pitched sounds need fewer participants as compared to textures.

5.6 Discussion

5.6.1 Use of image-schemas for audio quality description

We explored RQ3.1 by conducting an experiment with a simple 2AFC type of task and using pitched sounds and textures of longer duration ($\sim$ 14 seconds). We found that by visually articulating the descriptive audio quality under test, we improve the overall quality of responses collected via listening tests for these conditions. Using image-schemas, we employ conceptual models [202], which are both language and experience-agnostic and thus provide means to understand audio quality concisely. These findings reinforce previous research on the clarity and brevity of instructions [83, 85] and extend it to audio evaluation.

We term the pitched sounds used in this experiment as more recognizable than noisy textures as they had some semantic meaning associated with them. For instance, the timbre and/or MIDI pitch of the pitched sounds used in this experiment changed as the sounds progressed in time. In section 5.4.7, we saw that such pitched sounds were evaluated comparably to noisy textures under both visualization conditions. This implies that the quality of responses in a simple 2AFC experiment does not depend on the sound space complexity introduced due to the type of sound being evaluated in a test.

5.6.2 Use of image-schemas for designing listening test interfaces

We explored RQ3.2 by experimenting with a complex rank-ordering and proximity spacing type of task and using pitched sounds and textures of shorter duration ($\sim$ 2 seconds). The interfaces designed using image-schemas afforded the listeners the ability to visually "see" the reordered clips as compared to the language-based condition and thus might be the reason why they performed better. Interestingly, the participants using both interfaces could rank-order the sounds equally well. The advantage image-schemas condition had over language-based interfaces was its ability to allow participants to capture proximal distances between the sounds better. This can be attributed to the fact that the slider-based image-schema interface allowed for better visual positioning of the sounds under test in relation to each other as compared to the number of drop-downs in the language condition. Therefore, it is sufficient to use language-based interfaces for simple rank-ordering tasks and image-schema-based interfaces for more granular proximity spacing tasks.

In this experiment, participants evaluated pitched sounds better than textures under both conditions. This could be attributed to the fact that textures have more perceptual feature dimensions that could be used for evaluation compared to pitched sounds. For instance, for the texture sample of water filling a container, several other perceptual features, such as the rate of the water filling, the material, and the resonances of the bucket, etc., co-vary with the fill-level [226] confounding the parameter under evaluation. Furthermore, most participants on AMT are novice listeners and may be more intuitively familiar with pitched sounds than multi-event textures. Also, they may be more comfortable working on tasks with speech transcription, sound event detection, or music evaluation and less familiar with rank-ordering tasks for textures, thus lowering the quality of responses collected.

While in Experiment 1, the number of participants needed for maximum agreement did not depend upon the sound type, in Experiment 2, pitched sounds needed fewer participants than textures (see Section 5.5.6). This finding has practical implications when selecting the number of participants in a listening test. Thus, when task complexity is high in a listening test, a larger sample size is needed for unrecognizable multi-event soundscapes to match in agreement with recognizable pitched sounds evaluated under similar conditions.

Measuring perceptual linearity on a continuous scale, as done in Experiment 2, can be modeled as discrete pairwise comparisons as suggested by prior work in other domains [227, 228]. It should be noted that while discrete pairwise comparisons are easier to evaluate, we cannot record some important perceptual distance-based differences between the samples under evaluation. Thus, for AI-generated audio, we find pairwise comparisons are great for rank-ordering tasks, and interfaces designed using image-schemas are advantageous for more granular perceptual distance-based evaluations.

5.6.3 Amazon's Mechanical Turk as representative crowdsourced platform

Besides AMT, the audio AI community uses crowdsourcing platforms such as Zooinverse[6] and Crowdflower[7] to conduct audio evaluations. Each platform's affordances drive the type of experiments that can be conducted and the data quality collected. For instance, Zooinverse is a research-focused volunteer-driven platform, where participants' motivation to contribute may not be driven by monetary incentives [206]. Comparatively, many participants on platforms such as AMT or Crowdflower are driven by the payment structure associated with the task as income generation is their primary motivation [229–231]. Furthermore, Zooinverse-like platforms connect researchers and participants more intrinsically via their project discussion boards, which assists in a deeper connection and communication of the research compared to the asynchronous email-based communication afforded by AMT. Projects set up on Zooinverse are usually long-running and undergo a review by the Zooinverse team, who are assigned experts to assist researchers with best practices for setting up their experiments. Such affordances potentially allow for better overall quality of responses from volunteer-driven platforms than paid crowdsourcing.

We chose AMT to conduct our experiments because it supports short-term evaluation tasks and the convenience of experiment administration and setup. Though AMT does not afford between-subjects experimentation out-of-the-box, we set up our listening tests to log participant actions and disallow them from attempting multiple tasks. In addition, we explore communicating tasks and instructions, acknowledging asynchronous communication between participant and researcher. Our findings should hold even in a collaborative environment provided by platforms such as Zooinverse.

[6] https://www.zooniverse.org/
[7] https://www.crowdflower.com/

5.6.4 Other Applications of image-schemas

Image-schemas assist in visually, spatially, and statically representing the temporal attributes of artifacts, such as environmental audio. These metaphors can be extended to other modalities, including video, text, speech, and haptic effects. When analyzing narratives in videos and text, we can assess the comprehensibility of the sequence of events or the overall "shape" of the narrative using these visualizations. This approach is promising for evaluating various AI-generated linear or non-linear narratives in text or videos.

Besides visualizing the narrative sequences, image-schemas can also be used to evaluate the affect (emotion) the narratives induce. By integrating image-schemas, such as up/-down, container, and source-path-goal, participants can determine the sequence of emotions induced through the temporal content in video, text, or speech. The image-schemas for containers can represent various affect categories, while the directional schemas (up-/down) can indicate the intensity of the emotion. Additionally, the source-path-goal schema can assist in capturing these emotions directionally over time.

Another promising application of image-schema metaphors is for evaluating AI-generated haptic effects. Haptic effects are brief touch experiences used to convey immersive experiences for consumers in AR/VR applications. Visual metaphors can be employed for AI-generated haptic effect assessments by prompting participants to spatially sketch the temporal responses of the haptic actuators placed on different areas of the skin or body.

5.7 Summary

In this chapter, we highlighted the importance of using the visual metaphors of image-schemas in designing listening test interfaces for AI-generated audio. We introduced novel visual constructs to evaluate sound progression in rank ordering and pairwise comparison types of tasks and verified their effectiveness in improving the overall quality of responses in a listening test. Furthermore, we discuss the implications of using such visual constructs to evaluate sounds with varying complexities. Our findings shed light on the value of using such stationary constructs to communicate the temporal quality of audio for future researchers working at the intersection of generative audio modeling and human-computer interaction.

Chapter 6

Understanding opportunities for generative models in sound design practice

Chapter Synopsis

In this chapter[1], we address **RQ4**. Recently, many studies have adopted generative AI to assist in sound design co-creation. Most of these studies focus on the needs of novices and less on the pragmatic needs of sound design practitioners. This chapter aims to understand how generative AI models might support sound designers in their practice. We designed two interactive generative AI models as Creative Support Tools (CSTs) and invited nine professional sound design practitioners to apply the CSTs in their practice. We conducted semi-structured interviews and reflected on the challenges and opportunities of using generative AI in mixed-initiative interfaces for sound design. We provide insights into sound designers' expectations of generative AI and highlight opportunities to situate generative AI-based tools within the design process. Finally, we discuss design considerations for human-AI interaction researchers working with audio.

[1]With minor modifications from:
Kamath, P., Morreale, F., Bagaskara, P. L., Wei, Y., & Nanayakkara, S. (2024). Sound Designer-Generative AI Interactions: Towards Designing Creative Support Tools for Professional Sound Designers. In Proceedings of the CHI Conference on Human Factors in Computing Systems. CHI '24: CHI Conference on Human Factors in Computing Systems. ACM. doi:10.1145/3613904.3642040

6.1 Introduction

Generative audio models for music [14, 132, 232] have been well studied for their potential to support co-creation in the human-AI interaction literature [15, 16]. And yet, despite the growing adoption of AI models as co-creation tools for music production [45], very few empirical studies exist to assess their potential to offer new possibilities to the practice of sound design.

Most human-AI interaction studies for audio focus on the applicability of steerable generative AI interfaces to empower novice users in their creative goals [15, 16, 46–48]. Expert sound design practitioners spend years developing their creative design process and building inventories of sounds to apply in their next design project [4]. As such, their needs, expectations, and ways of working with AI-based tools necessarily differ from those of novices. Thus, in this paper, we aim to explore: *How can generative AI-based co-creation tools assist expert sound designers in their creative practice?*

We developed two interactive generative AI models as Creative Support Tools (CSTs) [17, 50] to explore the potential of this technology in assisting sound designers. As in [233], we use an experimental design strategy of deploying interfaces in real-world contexts to provoke discussions and answer research questions. We deployed our CSTs with sound designers to gather information about their expectations of AI, as well as the current challenges and opportunities for generative AI in their practice. Further, we captured the designer's interpretations of AI by designing the interactivity with our CSTs by incorporating elements of "use-qualities" [53] (also see Chapter 2, Section 2.2.4) such as pliability and ambiguity [89, 91]. While we developed two CSTs in this study, we did not aim to compare them with each other. Instead, we aimed to provide designers with two unique ways of interacting with AI-based tools to gather their reflections [74, 233] on using those tools.

We introduced our CSTs to nine professional sound designers and asked them to apply them in a creative endeavor in their practice. We conducted semi-structured interviews with the participants to reflect on their creative goals and the sounds they created using the CSTs. We gained three key insights through inductive reflexive thematic analysis [234] of the interviews:

- First, we outline *an AI-assisted sound design process* where we find how sound designers situate AI-based tools within their design process in practice. While performing creative tasks, we found that sound designers used AI models for performing fast iterations to create novel sounds and as an alternative to manual field

recording activities. They also used such sounds as layers to give the perception of plausibility to unreal sounds.

- Next, we found how sound designers *worked with unpredictability and ambiguity* and developed an intuition for interacting and controlling AI-generated sounds. We also found that designers often realized or understood failure modes in AI-generated output and worked towards ways of using ambiguity in their sound design.
- Lastly, we furthered our understanding of *sounds designers' expectations of generative AI* to build convincing cinematic experiences in terms of creator agency and owning the creative process

In summary, our contributions are three-fold: (1) we developed a novel understanding of generative AI in supporting creative exploration for the practice of sound design; (2) we developed two AI-based CSTs for future studies on using audio generative AI as a tool for sound designers; and (3) we offered five design recommendations for future human-AI interaction research for sound design.

6.2 Audio Generative AI CST Design

To develop our two CSTs, we adopt an approach from the "XAI for arts" community, where researchers working at the intersection of explainable AI (XAI) and arts developed novel ways to explore latent spaces of GANs for creative endeavors [74, 75]. In particular, we use the Example-Based Framework (EBF) [184] (from Chapter 3) and the Semantic Factorization (SeFa) algorithm [154] to explore the latent space of an unconditionally trained StyleGAN.

We designed the interactivity for our interfaces based on the principles for AI controllability outlined in Weisz et al. [235]: by using (1) domain-specific controls (for *interface-1*), and (2) technology-specific controls (for *interface-2*). Domain-specific controls use audio descriptors or acoustic parameters to control the generation from an AI model. Technology-specific controls, on the other hand, are generic controls that depend on the generative algorithm and are not necessarily related to the audio domain. Such technology-specific controls allow users to perform manipulations or edits directly in the latent space of the generative model. They are typically effective in making changes to the semantic attributes of a sound.

Both interfaces used the same underlying StyleGAN model and differed only in how the generation was controlled. Further, both interfaces provided opportunities to interact

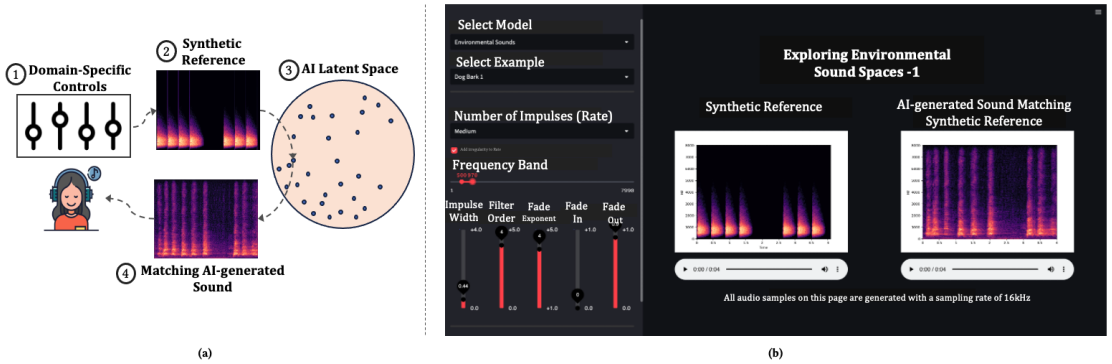


FIGURE 6.1: A conceptual diagram **(a)**, and screenshot **(b)** of interface-1. **(a)** A sound designer can use the domain-specific controls from ① to generate a synthetic reference sound seen in ②. This synthetic reference sound is used to "query" or "search" the latent space of an AI model shown in ③ to generate a matching AI-generated sound in ④. **(b)** The screenshot shows the placement of the controls and the synthetic and generated sounds as viewed by the designer on the web interface. Please see Appendix B.2 for a link to a Google Colaboratory version of this interface, and B.4 for image attributions.

with two StyleGANs - (1) one trained on a dataset of 'Hits & Scratches' called the *Greatest Hits Dataset* [166], and (2) another trained on a dataset of 'Environmental Sounds' from the *DCASE 2023 Foley Sound Synthesis Challenge* [127]. Using the 'Hits & Scratches' model, the sound designers could generate and explore a small set of timbres related to the impact sounds made by a drumstick hitting various hard and soft surfaces. Using the 'Environmental Sounds' model, the sound designers could generate and explore more complex timbres and sounds such as dog barks, footsteps, gunshots, motor vehicles, rain, and keyboard clicks. Further, we added some preset sound configurations on both interfaces, which participants could test during the study. These presets included timbre parameter settings, such as impact sounds on hard and soft surfaces or environmental sounds like a medium-sized dog barking.

All underlying AI models were built and trained using Pytorch [236] and were running on a single RTX 3090 GPU. The interfaces were built as web-based technologies such as Streamlit[2] and ReactJS[3] to run on web browsers for ease of access. Please see appendix B.2 for architecture and implementation details for both interfaces.

6.2.1 Interface-1 - Using domain-specific controls

For *interface-1*, we employed the use of domain-specific controls [235] based on acoustic parameters such as frequency band, impulse width, fade-in, fade-out, etc. to guide the generation of the sounds. For this interface, we use the EBF framework [184] outlined in Chapter 3. EBF uses a set of domain-specific controls to create a synthetic sound using signal processing techniques. This sound is then used to "query" or "search" the latent space of the StyleGAN for a matching, AI-generated sound. A conceptual diagram and screenshot of this interface are shown in Figure 6.1. We designed this interface with the use-quality of pliability [53, 89]. Pliability, as described by Löwgren, refers to the responsiveness of the design material in assisting the designer in iteratively refining the generated artifact to match their creative goals. *Interface-1* in this study is designed to enable sound designers to iterate and generate sounds by modifying a synthetic proxy. A user of this interface conveys their ideas to the AI model by designing a synthetic sound. The AI model, in turn, uses the synthetic sound to search and generate a matching, more realistic sound. The resulting audio for both the synthetic reference as well as the AI-generated sounds is displayed on the webpage. Additionally, we provided visual feedback to the users by displaying the spectrogram for each sound along with the audio on the webpage. We include this spectrogram visualization to allow the participants to focus on the spectromorphology of the sounds [101], or how the frequencies in the sound change or morph over time.

While we designed this interface to provide opportunities for reflection [74, 91] by giving greater flexibility in generating multiple types of synthetic sounds, not all synthetic references resulted in meaningfully matching AI-generated sounds. This unpredictability in the AI-generated sounds is due to the limitations of the training data used to train the GAN. We allowed this unpredictability on this interface by design to gather our participants' intuition about AI limitations.

6.2.2 Interface-2 - Using technology-specific controls

For *interface-2*, we employed the use of technology-specific controls [235] based on the SeFa algorithm outlined in [154]. In SeFa, dimensions for controlling generation are extracted by performing an eigendecomposition of the learned weights of the StyleGAN. That is, using eigendecomposition, the weights matrix of the StyleGAN are factorized into basis vectors which can then be used to perform latent space manipulations to

[2] `https://streamlit.io/`
[3] `https://react.dev/`

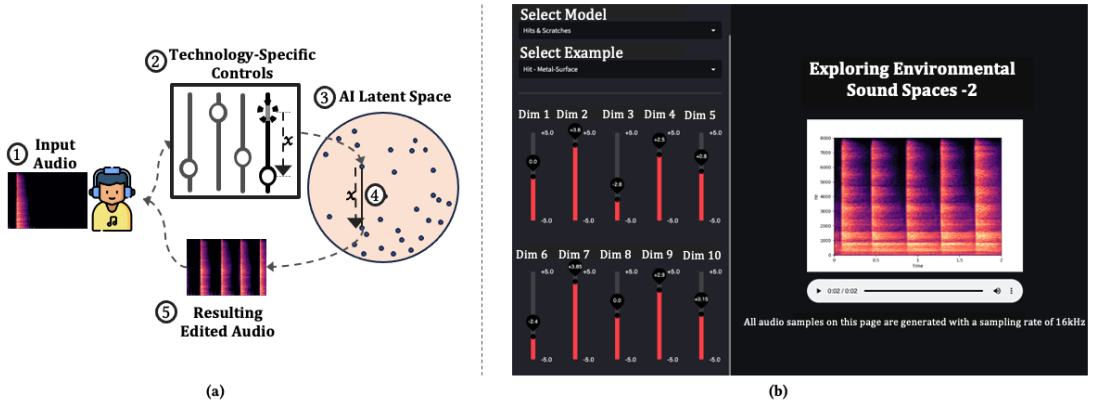


FIGURE 6.2: A conceptual diagram **(a)**, and screenshot **(b)** of interface-2. **(a)** To edit the audio in ① such that the number of impacts in the sound increases, a sound designer can use the technology-specific controls extracted from the latent space of a StyleGAN shown in ② to perform direct latent space manipulation shown in ③ and ④, resulting in the edited audio sample in ⑤. **(b)** The screenshot shows the placement of the controls and the generated sounds as viewed by the designer on the web interface. Please see Appendix B.2 for a link to a Google Colaboratory version of this interface, and B.4 for image attributions.

control semantic audio descriptors on a sound. Such semantic dimensions are usually unlabeled and are typically open to user interpretation of them. Users usually interpret each semantic dimension by performing and observing a few edits made by changing a dimension on the sound. We chose the top 10 dimensions (top 10 eigenvalues after eigendecomposition, see appendix B.2.2) found by the algorithm to perform sound edits on this interface. A conceptual diagram and screenshot of this interface are shown in Figure 6.2. As for interface-1, we displayed the spectrogram along with the resulting audio on this interface.

We designed this interface with the use-quality of ambiguity [53, 89, 91]. That is, we designed this interface to provide opportunities for reflection [74] by leaving the dimensions unlabeled. We allowed the designers to interpret this ambiguity in the dimensions based on the intuition they developed while engaging with the controls on this interface.

6.3 User Study

6.3.1 Participants

We recruited nine professional sound design practitioners (six male, two female, and one preferred not to say) for this study through snowball sampling. We used this sampling

strategy to reach academic and professional sound designers working in the industry. Starting with the authors' existing network, we asked individual participants whether they knew other practitioners interested in participating in our study. In our email, we indicated the study would take at least 1.5 hours to complete. Our sample size was thus pragmatic based on the number of sound designers willing to invest time in this study. Participants had diverse backgrounds in sound design, from designing sounds for products, movies, music, and games to creating sound for data sonification projects (Table 6.1). The median self-reported years of experience in sound design was 10 years (Min = 3 years, Max = 48 years). They were offered USD45 gift cards as a token of appreciation for their time in the study.

Table 6.1: Participant Details

ID	**Country**	**Experience**	**Description of Sound Design Experience**
P1	New Zealand	8 years	Sound design for visual media such as short films, documentaries, and games. Using sound as an aid and embellishment to story-telling. Experience in recording and mixing music. Undertaken audio post-production work.
P2	Spain	48 years	Sound designer and electronic music composer. Focused on audio perception and programmatic ways of creating sound. Worked on programmable synthesizers and libraries for various platforms. Educator for sound art and design. Currently focused on audio AI research.
P3	Hong Kong SAR (China)	9 years	Sound design for movies and animated films. Audio post-production for TV programs. Experience recording and mixing music, and foley sound effects.

Continued on next page

Table 6.1: Participant Details (Continued)

ID	Country	Experience	Description of Sound Design Experience
P4	New Zealand	7 years	Original sound creation and implementation for online and theatrical films, games, music production, and live performances. Field recording. Post-production work includes dialogue editing, sound mixing, audio restoration, and foley mixer. Educator for sound design.
P5	Netherlands	20 years	Designed sound to build brand experiences for various international brands and airport authorities. Designed "sonic identities" for brands ranging from sound installations for their public spaces as well as designing product sounds. E.g., the sound of a car's engine, doors opening or closing, etc. Focussing also on data sonification projects.
P6	Italy	3 years	Sound design for vehicle or gardening simulation video games working directly with environmental soundscapes. Designing quad ambiance and sound effects, and implementing them in the game engine.
P7	Germany	10 years	Sound designer and composer. Designed sounds for over 40 games. Also worked on sound design for films as well as some movie trailers.
P8	Singapore	10 years	Electroacoustic music composer using Ableton Live and FL Studio.
P9	Singapore	47 years	Music composition for ambient/rock and experimental/avant-garde genres. Worked for theatre and other projects that are in between sound design and music. Co-leader for a desktop Foley system. Also, writing music software.

6.3.2 Procedure

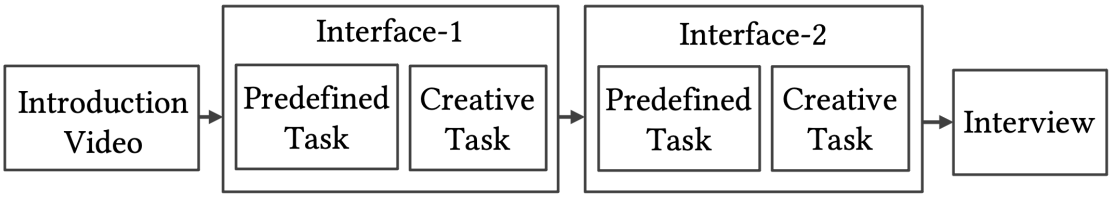


FIGURE 6.3: Overview of the study procedure

Figure 6.3 shows an overview of the procedure. The participants were sent a link to a web page outlining the task instructions[4]. This web page included a 3-minute introductory video explaining the tasks and providing a brief overview of the interfaces. To minimize any order effects, participants were randomly assigned into two groups. The first group attempted the tasks with interface-1 first before interface-2. The second group performed the tasks in the reverse order.

We first asked participants to complete a short, close-ended, predefined task to familiarize them with our interfaces. Subsequently, we asked them to complete an open-ended creative task to generate sounds they might use in their own practice or performance. As our participants were located in different parts of the world, they were asked to perform these tasks at their own pace and time and record their screen activity when performing the open-ended creative task. This approach was adapted from the video-cue recall method [51, 237] from the interactive arts literature for our purpose.

We subsequently conducted a semi-structured interview to gauge the participants' experience and feedback on the generative AI interfaces. Our server logs indicated that overall, the participants spent a median of 46.28 minutes (Min=24.11 minutes, Max=2 hours, 31.6 minutes, SD=44.46 minutes) exploring and familiarizing themselves with the interfaces. Participants recorded their screen activity when performing their open-ended creative tasks as instructed. For these creative tasks, the median screen recordings for each interface were 2.44 minutes long (Min=1 minute, Max=20.25 minutes, SD=5.36 minutes). We asked the participants to send us their screen recordings before the interview. The interviewer watched the screen recordings before conducting the interview and highlighted parts of the recording where participants employed different exploration strategies when using the interfaces. We discussed the participants' creative goals during the interview using the highlighted parts of the recordings as discussion prompts. The recordings were used as discussion prompts only and not as data for analysis. All

[4]See link on our webpage - `https://purnimakamath.com/thesis-related/chapter_6/#task-instructions`

interviews were conducted remotely and lasted for a median of 40 minutes (Min=32 minutes, Max=60 minutes, SD=10.19 minutes). Please see Appendix B.1 for the interview questions.

6.3.3 Data Analysis

Due to the exploratory nature of this work, we chose an inductive, reflexive thematic analysis (TA) approach [234] for analyzing the interview transcripts. One author conducted all interviews. Two authors (including the interviewer) collaboratively analyzed the data using a bottom-up approach. We first familiarized ourselves with the transcripts individually and independently reading them at least twice. We then coded the transcripts with quotes relevant to our research objectives. Next, we collaboratively combined and refined our codes using Atlas.TI[5]. As recommended in [238], we use a semantic coding strategy during our coding process where each code captures a semantic observation. For instance, a quote from a participant such as "some randomness (in the AI-generated output) is always refreshing" is considered as one code. Quotes from other participants making similar observations may also be tagged to the same code. This code, amongst similar other codes, is then organized under a theme such as "Non-determinism assists creativity". Through this process, we iteratively refined and identified 76 codes. We use affinity diagramming to assist us in collaboratively organizing the codes into 12 themes. These themes are organized under the 3 sections in the results section below.

In reflexive TA, meaning is not *"excavated"* [239] from the data, but is subjectively generated through a researcher's interpretation of the data [234]. This nature of the analysis makes it difficult to formalize a sample size or define data saturation (or the minimum number of participants needed before stopping data collection) [239]. Thus, instead of defining data saturation for this study, we resorted to deliberately seeking a varied group of participants based on their geographic location, background in sound design, and years of experience in sound design. With this, we tried to gather diverse views and opinions of AI during our study.

6.4 Thematic Analysis Findings

In the following three subsections, we organized the themes from our inductive, reflexive thematic analysis into three meta-themes: (1) An AI-assisted sound design process; (2)

[5] `https://atlasti.com/`

Working with unpredictability and ambiguity; and (3) Sound designers' expectations of AI for sound design.

6.4.1 An AI-assisted sound design process

6.4.1.1 Fast iterative exploration

Sound designers are constantly seeking new sounds to use in their work. *"Like if you're working on a sci-fi game, then you can't just use run-of-the-mill sounds. And just so people are always looking for new sounds, like a new palette so to speak"(P1)*. Some commonly shared frustrations our participants observed in their current design process were around the manual processes of creating new sounds on tight deadlines or low budgets. Creating and manipulating new sounds takes time, and it can be frustrating as *"a lot of back and forth happens when someone (a client) has something on their mind that they can't verbalize and then you're trying to figure out what they want"(P1)*. In such cases, being able to quickly and iteratively create novel sound samples using AI is beneficial.

> ***P1**: "It's really useful to be able to go through 20 iterations in less than half the time that it would take me to do it in the traditional way. And then because you can adjust so many parameters so quickly, then you're not stopping and changing things. You're not editing waveforms, you're not changing plugins. So I think it is really useful [...] I think people tend to overstate what creativity is. But to me personally, it is to be able to go through a lot of things quickly and to select the right bit of sound for that purpose."*

6.4.1.2 An alternative source to field recording

Often, sound designers sourced new sounds by field recording them and further processing them to develop new sound palettes. Such manual recording activities can be frustrating as they cannot always control a recording situation. *"You can't tell everyone in a city 'Be quiet for a second. I need to record this thing"'(P4)*. Typically, a 5-second recorded audio takes a couple of hours to clean, denoise, and process before use. In such cases, AI-generated sounds can be considered as a suitable and convenient alternative for *"finding interesting source material" (P7)*.

> ***P4**: "Most likely it would be I spend a day with the interface making a bunch of sounds and I just record all of them. I'd delete the ones that I don't think will be useful and I'd keep all the rest [...] I'd almost treat this like field recording in a sense, but instead of me actually going outside to record it, I am going into this interface to capture it."*

6.4.1.3 Creating unreal but tangible sound palettes

The bulk of sound in a film is usually added in post-production [2]. Sound designers typically develop and use a custom palette of sound effects for each film [9]. *"In sci-fi movies [...] we want to give people the kind of 'metal' feelings. That this world is made from science, and not really an actual world. To feel that it's a different world compared to my living world" (P3).* Thus, designers are often on the lookout for unreal, but plausible-sounding sound elements that assist in building immersive experiences for the consumers of such media. Using AI-assisted sound design tools in this study, designers were able to create such fantastical or alien, but tangible sounding palettes.

> ***P4**: "Obviously you can make stuff like this in a synthesizer, but the problem is it sounds like a synthesizer, it doesn't sound real.[...] And while this (AI-generated sound) doesn't sound like something that's real, because it's in some way based on something that is a real recording, it still has a kind of tangible quality to it. And that's kind of what the value is. You can make synthetic sounds that still sound somewhat like there's a real object doing it."*

Although the models used in this study were not trained with the goal of generating unreal sounds, the interactivity encoded in them enabled the designers to generate such sound palettes. Five designers *(P1, P3, P4, P5, P7)* noted that the generative AI tools were better used for generating such sounds rather than replicating real-world recordings.

> ***P7**: "We are always on the hunt for those kinds of elements where we can layer something that actually exists with something that does not exist to enhance immersion for the consumer. Those are the elements that are more interesting for me personally. If I want to have the recording of a falling tree, I can just go out and record it. I don't need a tool for that."*

Further, six sound designers *(P1, P3, P4, P5, P7, P8)* we interviewed said that they rarely used sounds from their own libraries or external databases as-is in their projects.

They usually processed the recordings through 'effect chains' (i.e., using a Digital Audio Workstation (DAW) to process sounds through a chain of effects such as adding/removing distortion, reverb, etc.) to fit the requirements of different projects. They found using the generative AI tools in the study useful as part of such effect chains. The interactivity in the tools could be used to extract textural components from various sounds, which can be used as layers to enrich other recorded or synthesized sounds.

> ***P7****: "(Describing their creative task result) For me, that would be like a sci-fi layer or that could be used in some trailers when there is something popping up. Or when a spaceship flies by. You can use that as a sweetener."*
> ***Interviewer****: "What is a sweetener?"*
> ***P7****: "Yes, say you have a sound, but then you put something (.) on top of it like spices. And then it's like, wow! That's new!"*

6.4.1.4 Annoying, but Fun!

Both AI-based tools in this study embodied non-determinism in controlling the generated sounds by using either synthetic sound queries (interface-1) or unlabeled dimensions (interface-2). This nature of the AI-based tools was appreciated by our designers for their ability to allow exploration and serendipitous discovery of novel sounds, even when the sounds were not in line with the participant's original task goal. For instance, when performing his open-ended task with interface-2 *P2* said:

> ***P2****: "I understood that I was exploring and there was some discovery. So every once in a while you'll hear me say, 'Oh, I like that!'. Even though it wasn't necessarily exactly what I was looking for, it had something that I liked"*

Further, most designers noted that while this exploratory nature of the AI-based tool was fun, it would be annoying or frustrating to work with it on task-oriented work regularly, especially on a deadline.

> ***P4****: "Well, one thing I found fun was seeing how the AI responded to the synthetic reference and how it didn't listen to me, right? So sometimes I made a change and it didn't quite reflect that and I found that interesting. But if I worked with this every day and I was on a project with a deadline and I really wanted it to listen to me, then I'd imagine it would stop being*

fun and it would start becoming frustrating trying to get it to do those specific things."

6.4.2 Working with unpredictability and ambiguity

6.4.2.1 Exploration strategies

For interface-1 (domain-specific controls), the general exploration strategy we found amongst designers was following a 'broader first, then narrower' strategy. For instance, participant P5 said she would experiment broadly first, say using a wider range of frequencies, and then narrow down to the specific perceptual outcomes she had in mind by employing *reduced listening* (reduced listening is when designers concentrate on the sound for its own sake, as a sound object, independently of its causes or meaning [240]).

> ***P5**: "And as I say in the synthetic reference, this worked quite well because the sounds were like (MIMICKING THE SOUND OF A CICADA TRILLING), so I selected frequencies that are typical without too much thinking. Let's say higher frequencies. I did everything quite rough, not knowing the system and then trying to achieve this to get as closer as I could (to the goal)."*

For interface-2 (technology-specific controls), to understand the parameter space they were exploring, participants employed multiple strategies such as - (1) simply playing around with each control and observing its effect on the generated sound (*P1*, *P2*, *P5*, *P8*, and *P9*), or (2) by using a 'Systematic Change without Compounding' where the parameters are reset to the original positions first and only one parameter is changed at a time to observe or isolate its change *(P4)*, or (3) by using a 'Min-Max' strategy by observing the generated output at the minimum and maximum limits of a parameter's range *(P3, P7)*. While P3 used the 'Min-Max' strategy to clearly isolate the change made by a parameter, P7 used that strategy to see how far he could push a control to get something *"new or weird" (P7)* out of it.

Overall, we observed that the participants who approached the exploration with both interfaces systematically discovered new sounds and were generally satisfied with the exploration, even when the outcomes did not match their original goals. One participant (P6), who reportedly approached the exploration randomly and without a goal, found it difficult to get any satisfactory output and gave up performing the task. Although other participants discovered interesting new sounds from their explorations, they expressed

their desire for more predictability in the controls so as to be able to use the tools regularly.

6.4.2.2 Opportunities from ambiguity

As outlined in Section 6.2, both interfaces used the same underlying AI model but with different interactivity mechanisms governed by different levels of ambiguity to control the generation of sounds. All designers in this study noted that while both interfaces could generate unpredictable outputs from the AI models, the controls on interface-1 (domain-specific controls) were more intuitive and comprehensible than those on interface-2 (technology-specific controls). This was primarily because interface-2 had (1) unlabeled controls and (2) a higher number of controls than those on interface-1. When using interface-2, some designers noted that the exploration seemed like a *"trial and error" (P7)*. In contrast, others *(P2, P3, P4, P5)* found that this *"lowest form of control" (P4)* gave them greater opportunities for exploration as there were more control parameters to *"twist" (P3)*.

> ***P4**: "One thing, of course, is it's less intuitive in the sense that nothing's labeled, [...] but by not giving it a name, it actually made more sense in a way, because you just see that as an abstract quality, the AI is doing something with it. So just naming them arbitrarily kind of made you pay attention to what they were actually doing more and not expecting something that it wasn't going to do. The lack of specificity makes it feel open in a different way."*

Further, when using interface-2, all designers expressed the need to be able to label the dimension based on their preferences. Designers gravitated towards labeling the dimensions based on either semantic changes *(P1, P6, P8)* or acoustic changes *(P2, P7)* they observed in the generated output.

6.4.2.3 Modes of working with audio interfaces

Although designers indicated that labeling dimensions would enable them to use the interfaces better, especially when using interface-2 (technology-specific controls), two designers *(P3, P5)* reported that they relied on listening to understand the role of each parameter, even when using labeled controls on interface-1 (domain-specific controls).

Such designers built an intuitive knowledge about the effect of each control parameter on the generated sounds and did not rely on the descriptions provided on the interface.

> ***P5**: "Usually I don't even read descriptions much. I just listen to what comes out. It's a nicer way of exploring for me. And then when I'm familiar, I can control it."*

Further, while using interface-1, we noticed five designers *(P1, P2, P3, P5, P8)* stopped listening to the synthetic sounds and focused on listening to the effect of the parameter change directly on the matching AI-generated sound itself. Reading and observing the changes on the synthetic spectrogram was sufficient for them to understand the effect of their changes. Thus, they could focus more on the effect of their changes on the AI-generated output.

> ***P2**: "Since [..] the AI-generated (sound) was really what I was exploring, [..] and I can read the spectrograms well enough to know that I just didn't have to go through that intermediate step. So spectrograms were helpful in kind of building out what the goal was."*

Finally, two designers *(P3, P8)* found it easier to create atomic units of sounds, such as a single impact sound or a single dog bark. Then, fixing and editing that single unit's important semantic and perceptual aspects and looping or repeating it in a DAW. This gave them better control of the creative process in adjusting the variability of the sounds to their liking.

> ***P3**: "I want to create the sound that is, actually can be used in my work. I think it should be one - how to say, one should sound, not (THUD THUD THUD THUD). Only one (THUD). If I need more of this, I can copy-paste (loop or repeat it in a DAW)."*

6.4.2.4 Understanding unpredictability of the response

As outlined in section 6.2.1, for interface-1 (domain-specific controls), we gave greater flexibility in generating the synthetic sounds, while not all synthetic sounds resulted in meaningfully matching AI-generated sounds. For instance, the *Greatest Hits* dataset was limited by a certain range of rate of impact (number of impact events per second). When

designers tried to query the AI model for higher rates, the model generated unpredictable responses. During our interviews, we discussed the nature of the generated sounds and asked our participants if they understood the reasons behind the AI's unpredictability due to its limitations. Three participants *(P2, P4, P5)* were familiar with the idea that AI was limited by its training data. Participant *P2* had experience with building and using AI models, and participants *P4* and *P5* were familiar with popular generative models such as ChatGPT [19] or DALLE-2 [20]. Participants' prior experience with the limitations of generative AI across different modalities might have made it easier to reconcile their understanding of the failure modes in our interfaces, especially when the changes they made did not align with their expectations. For instance, while explaining the unpredictable response from interface-1, P4 said:

> ***P4**: "Often, changes in synthetic reference didn't clearly correlate to the changes in the AI-generated sound. [...] Sometimes the fade-out parameter didn't really do that much to the AI-generated sound."*
> ***Interviewer**: "Can you tell me why you think that is happening?"*
> ***P4**: "Why? Not exactly sure why it wasn't following along exactly, but I'm guessing it's because it's trained on a certain kind of response that already has a certain type of fade-out innate in it, and so when you change the fade-out, there's only so much it can change based on what kind of input it has had."*

6.4.3 Sound designers' expectations of generative AI

6.4.3.1 Cinematic effect over accuracy

Through our interviews, we found that interviewees focussed mostly on the overall perceptual aspects of the sounds they worked with. Aspects such as where the sound originated from were not necessarily important to them. For instance, although we set up our AI CSTs to generate 'Hits & Scratches' impact sounds made by a drumstick, the sound designers used the models to create novel base sounds and *sweeteners* for footsteps *(P1)*, fantastical 'adolescent monsters' *(P3)*, trilling cicadas *(P5)*, sci-fi whooshes and flying machines *(P7)*, and as layers over percussive drum beat *(P8)*.

> ***P1**: "I think the most important thing, whether it's movies or games, is not accuracy so much, but immersion. So the footsteps that you hear in a movie, do not sound like that in the real world. Like, if you punch someone*

> *in the real world, it doesn't sound anything like what it does when Harrison Ford punches someone. The whole point (of sound design) is immersion and entertainment."*

6.4.3.2 Creative agency and ownership

Currently, most research in generative AI focuses on building omnipotent intelligent agents that can do it all—agents that can create art or compose music directly instead of being an enabler for creativity. While tools with greater AI agency would work well for novice users, for sound design experts, there are more opportunities for AI as an enabler rather than a creator in itself.

> ***P4**: "So a lot of other AI seem to be trying to replace a creator so that someone can get sounds who don't know how to make them, whereas this one seems more useful for someone who already knows how to make sounds but just wants to add to their arsenal by having another tool."*

In [4], Susini et al. emphasized that sound design as a practice is not just concerned with generating new sounds, but is also associated with a designer-led research-oriented design process grounded in psychoacoustics and sound cognition. Although most generative AI systems focus exclusively on the generation of new sounds, they do not focus on *"what the sound should do, or what it should be" (P5)*. As such, the results from our interviews suggest that the best use of AI is as a Creative Support Tool, as a part of a larger creative process owned and controlled by the designer.

> ***P5**: "I would like to keep the ownership of the creative process. I imagine the sound as it should be because it comes from a long research [...] The creative design process is much more than making the sounds. It is more about knowing what you want and finding the right tools.[...] So if the AI is also part of the research process, it could have good ideas."*

Finally, our results indicate that AI algorithms have the technological capability to provide means for creators with novel ways of creating sound for their work, which traditional signal-processing techniques cannot do. For instance, in our study, we observe two such instances where designers were able to discover novel base sounds for their sound palettes during exploration or extract *sweeteners* or textural components to layer over other sounds (see section 6.4.1.3). This capability to modify audio signals in novel ways gives creators greater opportunities to create new artifacts.

> ***P4**: "The approach where it is more about creating the individual units of sound rather than the finished product of sound, makes much more sense. It seems at least to be more achievable than what AI seems to be doing in the visual space. Because it doesn't always necessarily understand composition, it gets things roughly in place. What I've seen on people using AI for sound is that it's good to get good approximations, but not necessarily always to do things all the way."*

6.4.3.3 Need for focus on AI for sound design

Most current research in audio synthesis focuses on music and speech production, and very little work exists to model environmental sounds [241]. This feeling was conveyed by P4 during the interview:

> ***P4**: "A lot of the applications you're seeing right now are kind of in the infant stages a lot of the time. From what I've seen so far in sound there haven't been that many great uses of AI so far, at least ones commercially available or available on the market. And a lot of that, I think is because they're taking a more music approach where they're trying to streamline the job of a music producer."*

Further, given the recent surge in text-to-audio models, two designers *(P4, P5)* felt that AI models that needed to be prompted using text would be a barrier for sound design, which needs granular, continuous, and *"intimate control" (P2)* to design sounds. Developing controls over AI models where designers can *"leverage their current skills" (P5)* instead of learning newer ways to prompt AI models would be more beneficial for creator use.

6.5 Discussion

In this study, we sought to investigate how generative AI technologies could support sound design practitioners in their creative work. We found that AI-based CSTs could assist sound designers in their creative process by providing means to iterate over ideas quickly, generate fantastical and novel-sounding elements, and reduce the need to manually source individual artifacts via field recording for their creative work. Further, we found that although the unpredictability of controlling the AI-generated artifacts assisted

in the serendipitous discovery of new sounds, the exploratory nature and unpredictability in controlling the generation could hinder task-oriented work. Further, in our study, the sound designers employed various strategies while exploring the design space generated by the AI-based CSTs. These strategies helped them better understand the limitations of the generation capabilities of AI-based tools. Finally, while AI algorithms are usually incentivized to accurately replicate real-world sounds, in contrast, we found that sound designers were more interested in the overall perceptual aspects of the sound than its accuracy. We thus found that AI-based CSTs could easily be integrated as part of a larger creative design process owned and controlled by the designer. Such CSTs can produce novel sound elements that sound designers can incorporate into their compositions as layers over other sounds or use as individual components for a better cinematic effect than the accuracy in their compositions.

6.5.1 AI assistance in the practice of sound design

Recently, human-AI interaction researchers have been increasingly interested in understanding how mixed-initiative creative interfaces (MICIs) [66] can be applied in a work setting in different domains of creative work [235, 242, 243]. In our work, we respond to these questions in the context of sound design by proposing a mode of working with generative AI where designers perform exploration and creation using AI-based CSTs. Findings from our exploratory study suggest that such tools can assist in a fast iterative exploration (section 6.4.1.1) to help sound designers find novel sounds to use in their work. This finding is in line with some recent research on CSTs in the visual domain, in music composition, and in storytelling where algorithmic tools were used predominantly for idea generation [15, 16, 24, 27, 244, 245]. Further, such AI-based tools can generate synthetic surrogates of real-life sensory information (such as, in our case, field recordings (section 6.4.1.2)) which can constitute realistic and convincing alternatives to this information. Consequently, (sound) designers could save the time and resources needed to obtain this information in the first place. This observation could be extended beyond the realm of sound and also include visuals and other sensory modalities.

In [23], researchers note that while the unpredictability (section 6.4.2) emerging from AI-based tools supports creativity, it could be a hindrance to task-oriented creative work. We further this understanding for sound design (section 6.4.1.4) and find that sound designers might overcome this limitation by performing exploration (section 6.4.2.1) as a separately focused task [246], by employing *"reduced listening" (P5)*, to *"build a library" (P4)* of novel sound palettes for use in their projects. The possibility of using CSTs in this way to generate novel individual units of sounds, instead of entire compositions, gives

professionals another tool *"in their arsenal" (P4)* and more ownership of their creative process (section 6.4.3.2).

6.5.2 Constrained and Unconstrained Randomness

Previously, researchers have investigated the role of constrained and unconstrained randomness in interactive systems on user experience [247, 248]. In [247], using an example of a music-listening interactive system, the authors observe that, at times, unconstrained randomness can contribute to rich user experience (such as serendipity). They also note that this positive experience depends upon the size of the audio library, where large-sized libraries can have detrimental effects on the listener experience. In such cases, adding constraints to randomness (by constraining content) allows the users to manipulate or control the affective state of their user experience. We observe this duality of unpredictability and constraint in our study. Our impact sounds 'Hits & Scratches' model was smaller and more constrained in terms of the variety of sounds generated compared to the 'Environmental Sounds' model, which generated sounds from seven classes. Our participants found models with large variances in timbres, such as the environmental sounds model, detrimental to targeted creative exploration. For instance, participant P7 reported: *"The variety of sounds that I got out of the (environmental sounds model) was very extreme. I think that a tool that offers such a broad variety of results is like a two-edged sword."*

Further, our interface-1 was constrained in terms of providing means to explore the AI's latent space using only synthetic sounds, compared to interface-2, which provided means for unconstrained exploration directly in the latent space of the model. While using our CSTs, P6 reported: *"(Interface-1) was just like playing with an old synthesizer or something. It was quite easy to grab things and just tweak them and see what happened. (With interface-2) none of these settings did anything I was expecting at all."*. Our findings thus indicate that constraints implemented by either smaller models (such as the 'Hits & Scratches' model) or by using synthetic sounds for steering the CSTs assisted designers in better understanding the capabilities of AI (see section 6.4.2.4) than when using larger models or interface-2.

6.5.3 Reflections on designing and implementing AI-based tools for sound design

On selecting interactive AI models: While we implemented two CSTs in this study, our aim was not to compare them with each other but to provide our participants with two unique ways of interacting with the underlying AI model. While selecting algorithms for interactivity, we aimed to explore algorithms that worked primarily in a post-hoc fashion (i.e., worked on existing pre-trained GAN models). We found that using methods such as SeFa [154], we could integrate any available pre-trained GAN models from existing marketplaces [249–251]. Further, using methods such as EBF [184], enabled us not only to use domain-specific controls for exploration but also additionally constrain multi-class large audio models using class-based soft constraints [160]. Using these soft constraints, the designers could target their exploration to a part of the latent space oriented toward that class. Thus, we found both these methods effective in providing a wide range of exploration options [49] within our CSTs. Such methodologies for designing interactivity over AI models can be easily extended to other modalities, such as images. In light of the recent environmental impact [252] due to the training of large generative AI models, we suggest future CSTs, for all modalities including sound design, could make use of existing pre-trained models by leveraging such post-hoc methods for interactivity.

On visualizing sounds: While designing our interfaces, we visualize the spectrogram of the generated sound because the controls on both interfaces modified the spectromorphology [101] of the sound. Interestingly, through our interviews, we found that these visualizations provided means for the designers to describe their creative goals in spectromorphological terms. For instance, participants used terms such as *"seeing the individual events" (P2)*, *"fade-in is quite long" (P4)*, or *"removing the initial transient and softening it to leave the body and tail" (P4)*, etc. Previously, researchers in the explainable AI (XAI) for arts [75, 253] used latent space visualizations to *explain or debug* their creative goals. We build upon this work and suggest that spectrogram visualizations could provide a great way for designers to communicate their creative goals and understand AI-based CSTs' output.

6.5.4 Ambiguity in interactive user control

Interface-2 in this study was designed based on the use-quality of ambiguity. We deliberately left the dimensions unlabeled on this interface to allow the designers to interpret them based on their intuition. The ambiguity in the dimensions made the exploration

"more open (P4)"(section 6.4.2.2), and different participants came up with different semantic or acoustic explanations for the effect of each dimension on the edited sound (section 6.4.2.1). Participant P6 reported that Dimension 6 on the interface seemed to semantically change if the sound source was *"outside or inside the room"*. Further, P1 reported that Dimension 7 and 10 were similar to acoustic high-pass and low-pass filters, and P3 commented that Dimension 10 changed the pitch of the sound. By naming the dimensions differently and using semantic or acoustic labels, the designers could use the sound design space in their creative work in a personalized way.

Further, with interface-2, participants had to adopt a more varied number of strategies to meaningfully explore the sound design space (section 6.4.2.1) compared to interface-1. Therefore, although interface-2 opened up more personalized avenues for the designers to interact with the AI, the ambiguity in the dimensions got in the way of its *agentive flow* [254], a highly engaging state of interacting with an AI-based CST. The ambiguity in the controls in interface-2 made the designers focus more on the intricacies of the system rather than on their creative output.

6.6 Design recommendations for human-AI interaction in sound design

In this section, we outline five design recommendations for interactive generative AI. We specifically reported some quotes capturing rich insights from our expert practitioners to inspire our readers.

DR1: Design interactivity using intuitive controls

From among our participants, *P2* and *P9* had extensive prior experience designing audio interfaces, synthesizers, and programming desktop foley systems. Their advice on designing a good perceptually relevant set of controls for sound synthesis systems is as follows. They suggest a good control should be:

- **Perceptually monotonic**: If you moved a control forward to change the sound by an X amount, then moving it more in the same direction should do more of X.
- **Perceptually linear**: This principle builds upon monotonic controls. If you moved a control by an X amount in the forward direction and then moved it

the same amount in the reverse direction, both changes to the sound should be perceptually the same.

- **Perceptually orthogonal**: If you had multiple controls, a change in one control should be independent of others.

These principles are especially important when developing technology-specific controls (as on interface-2), as these controls are extracted by an algorithm from the latent space of a generative model. We thus propose future human-AI interaction researchers focus on constraining such algorithms to yield specific changes based on these principles.

DR2: Variety is a two-edged sword

The general trend in large language models or image generation research is to build large overarching generalizable AI models that cater to generating a large variety of images, art, or text. A similar trend is observed in audio, where a large audio model generates music, environmental sounds, and speech [132, 255]. Such large audio models can perform well as tools for exploration but are less useful for task-oriented work. This is particularly due to the complexity of the learned latent space. Small changes in the parameter space of such models can lead to large perceptual changes in the generated sounds. Participant P7 termed this variety as a "two-edged sword". We thus propose that future interactive AI applications for sound design focus on allowing designers to explore smaller models trained on a more targeted range of sounds. Or provide means to constrain the exploration of large audio models based on class, semantics, or other perceptual aspects of the sound (see section 6.5.3).

DR3: More cinematic effect than accuracy

In section 6.4.3.1 and 6.4.1.3, we showed that our participants valued perceptual aspects of the generated sounds and the AI's ability to generate 'unreal but tangible' sound palettes, more than the accuracy or the origin of the sound. Currently, most audio AI algorithms objectively incentivize the replication of real-world sounds. While real-world sound replications are useful as an alternative to field recording (section 6.4.1.2), they will have very limited use in being able to generate novel sound palettes. We thus propose that there is value in pursuing a research approach where AI models *"do not replicate real life too well" (P4)* and can extract textures and patterns from sounds. This approach would give artists and creators more creative tools in their arsenal rather than simply automating the generation of real-world sounds they can record easily.

DR4: Seeing sounds as an alternative to listening

Previously, Cartwright et al. [29] demonstrated that when using visual representations of sounds such as spectrograms, they collected better annotations for sound events than when using audio alone. Visual spectrogram representations of the sounds allowed annotators to 'glance-and-click' on the sound events while listening, which improved the accuracy of the collected annotations. In our study, we make a similar observation. Sometimes, the designers used the spectrograms on the interfaces as a proxy for listening. *"It's very nice to have the spectrogram because this gives you a good forecast. It is a good shortcut to imagine how it will sound like so you can even not listen to it" (P5).* We thus propose that using such visual representations of the sounds can reduce the cognitive load associated with making small edits and stopping to listen to the generated sounds, especially when doing exploratory work.

DR5: Improving the explainability of dimensions

As observed in sections 6.4.2.3 and 6.4.2.2, although most designers found the ambiguity in dimensions a hindrance to task-oriented work, they observed that giving them the ability to personalize the dimension names would improve the usability of such tools and the explainability of the dimensions (especially with interface-2). *"With the 10-D interface, I found myself wanting to change the label after I explored it so that I could remember what it did for me"* (P2). Further, in our conversations with P6 and P7, we observed that for understanding and learning controls on synthesizer interfaces, designers usually relied on not just the names of the controls but also their ranges and units of control. For instance, units such as 'dB per octave' are associated with filtering frequencies. P6 observed that on interface-2, all dimensions operated in a range of [-5, +5] with no units, which made it difficult to memorize the function of each control. We thus propose future human-AI interaction research to encompass dimensional controllability for sound models to rescale the ranges and adjust or assign units on controls to fit existing techniques on commercial synthesizer interfaces.

6.7 Summary

This chapter investigated how sound designers can use generative audio AI models in their creative practice. We designed and implemented two interactive audio AI CSTs and invited nine professional sound designers to apply the CSTs in their practice. Through

semi-structured interviews, we gathered insights on how to situate AI-based tools in the sound design process, the sound designer's ways of working with unpredictability and ambiguity in AI, and their expectations of generative AI-based tools. Further, we reported five design recommendations for future interactive AI-based creative support tools for sound design. Through this work, we hope to bring focus to this area of interactive audio AI and explore opportunities to improve AI assistance in the practice of sound design.

Chapter 7

Discussion & Conclusion

In this chapter, we synthesize the insights gained from the past chapters on designing and evaluating AI-based Creative Support Tools (CSTs) for sound design in response to this thesis's aims, research questions, and grounding in human-centered design philosophy. We conclude by outlining the limitations and future directions for this research.

7.1 Summary of Findings

In this section, we revisit our aims and research questions to synthesize the overall contributions and present them below.

7.1.1 Aims

This thesis aims to investigate approaches to designing and implementing steerable AI-based CSTs for sound design. It also investigates ways to perceptually evaluate such steerable models and build our understanding of the challenges and opportunities of applying such models in a practice-oriented sound design environment.

To achieve these aims, we applied principles from the Human-Centered AI (HCAI) framework to develop CSTs aligned with the creative process and workflows grounded in sound design theory. This work demonstrates novel ways to design and evaluate CSTs for sound design and creativity by:

- **Supporting exploration** by enabling means to explore the AI-based CST's design space to assist in novel sound discovery. Using frameworks outlined in this thesis, sound designers can explore this design space to search for sounds based on their semantic properties of interest by using synthetic sounds to "sonify" (artificially re-creating) or sketching their creative intent.
- **Steering by way of interactive controls** by facilitating interactivity with the AI-based CST for performing creative tasks in sound design, such as sound morphing. Using frameworks outlined in this thesis, sound designers can steer AI-based CSTs to edit semantics or morph two or more sounds in a fine-grained way.
- **Designing interfaces for non-audio experts** when conducting perceptual listening evaluations for descriptive audio qualities using AI-generated sounds on crowdsourced platforms.
- **Designing for creative engagement in practice** by employing the use-qualities of pliability and ambiguity in AI-based CSTs. By enabling exploration for novel sound discovery and using emergent properties of the latent space as ambiguous and unnamed controls, this work demonstrates how sound designers interpret the controls and explore the AI-generated design space for their creative work.

This work expands on the human-centered AI framework, bringing new knowledge to designing and evaluating such systems for creativity using audio, specifically for the creative pursuit of sound design.

7.1.2 Research Questions

[**RQ1**] How can we perform exploration using generative audio models trained on unlabeled data to generate environmental sounds using user-defined semantic attributes?

In Chapter 3, we introduced a novel guidance framework, "Example-Based Framework" (or EBF), to find user-defined attribute guidance vectors in the latent space of a GAN trained on audio textures. Through comprehensive objective and subjective metrics, we demonstrated that this framework enabled us to perform fine-grained semantic edits to the generated sounds. This method enabled novel ways to explore the latent space of the StyleGAN trained on unlabeled textures, such as impact sounds and the continuously varying texture of water filling a container. We also demonstrated the simplicity of extending this framework to other creative tasks, such as semantic attribute transfer, where users can select a semantic attribute on a randomly generated texture and transfer that

attribute to another texture sample. Using frameworks such as EBF, users can "probe" pre-trained generative models and engage in exploration to understand the breadth of the system's capabilities in a human-centered way.

[**RQ2**] How can we build steerable generative audio models that support creative sound design tasks such as audio morphing?

In Chapter 4, we introduced the creative task of morphing sounds for sound design. We used an existing pre-trained latent diffusion-based text-to-audio (TTA) model and designed fine-grained controls over discrete text prompt tokens while morphing two or more sounds. We outlined the "MorphFader" algorithm to morph sounds by interpolating the cross-attention matrices generated per layer and per diffusion step. Through objective and subjective metrics, we demonstrated that this framework enabled us to perform continuous, granular morphs between two text prompts. This method enabled novel ways to creatively explore the text-based semantic space generated by TTA models. We developed interfaces using MorphFader to demonstrate the ability of our method to generate plausible and semantically relevant audio morphs in real-time.

[**RQ3**] How can we perceptually evaluate audio generated using generative audio models for their descriptive semantic qualities using non-experts on crowdsourced platforms?

In Chapter 5, we first motivated the need for perceptually evaluating sounds generated by steerable generative audio models for their descriptive qualities of sound progressions, such as smoothness or goodness of the morphed sound or realism or plausibility of the generated sound. Based on the metaphors of image-schemas we designed visual constructs and interfaces to evaluate sound progressions or morphs in rank ordering and pairwise comparison tasks. We conducted experiments on a crowdsourced platform with non-expert listeners who may or may not have music or audio backgrounds. Using both pitched sounds and audio textures, we verified the effectiveness of visual constructs in improving the overall quality of responses collected in a listening test.

[**RQ4**] How can steerable generative audio models assist professional sound designers in their creative practice?

In Chapter 6, we studied how generative audio-based CSTs can assist professional sound designers in their creative practice. We designed and implemented two interactive generative audio models as CSTs and asked nine sound design professionals to apply the

CSTs in pre-defined and creative tasks. Through semi-structured interviews, we asked the participants to reflect on their use of the generative models to help us gather information about their expectations of AI and the current challenges and opportunities for generative models in their practice. This qualitative methods study helped us outline an AI-assisted sound design process. We also developed an understanding of how sound designers worked with unpredictability and ambiguity in the AI-generated output. We also outlined five design recommendations to support the creative task of sound design for future researchers of audio AI-based CSTs.

7.1.3 Synthesized Contributions

In this section, we propose new concepts to extend the knowledge and theories of HCAI and creativity support and combine design insights from the previous chapters for designing future human-centered AI-based CSTs for sound design.

7.1.3.1 Exploration by "Sonic Sketches"

A key proposition of this thesis is enabling exploration of the AI-based CSTs design space based on user-defined semantics. This is especially due to the lack of datasets with strong semantic labels for training AI-based CSTs. The novel approaches outlined in this thesis enable exploration by capturing the sound designer's creative intent in the form of synthetic sounds—an approach that we define as exploration by "sonic sketches".

As the name suggests, sketches are simplified caricatures of real-world phenomena. For sound, we define sonic sketches as signals representing the designer's creative intent by synthetically synthesizing the semantic aspects of the sound they want to generate. As shown in Chapter 3, such synthetic sounds can be generated by the parametric acoustic synthesizers [93–96] and physical modeling techniques [97–99] from the domain of audio signal processing.

Previously, in Chapter 2, HCAI principle of "capturing intent" [64] and other interaction design techniques of visual "sketching" [62] for exploring creativity and using sketches for rapid iterative exploration [65] were introduced. This thesis builds upon these techniques to propose a design paradigm for AI-based CSTs for sound exploration. The framework outlined in Chapter 3 relies on a synthetic sound generator, which can be granularly controlled to generate sonic sketches. While such sonic sketches are representative of the designer's creative intent, they lack the realism and the textural aspects of real-world sounds. The framework uses the sonic sketches to "search" or "query" the AI-based

CST's latent space to produce a matching real-world adjacent sound. Typically, the parameters to control the generation of environmental sounds can vary greatly. For example, when generating dog barks, the designer may want to control the pitch of the bark, while for footsteps, the semantic control might involve changing the material of the floor. Exploration based on sonic sketches is especially useful for such CSTs as it enables designers to creatively explore the design space without the limitations induced by pre-defined labels or controls devised by the algorithm's developers.

7.1.3.2 Novel "Exaptations" for Creativity Support

Chapter 2 of this thesis discusses how human creativity occurs in a conceptual space. Creativity can be combinatorial (i.e., combining the previous two ideas to create something new), exploratory, or related to transforming existing ideas into something new. As such, human-centered CSTs must be designed to support novel ways of creating new artifacts. Typically, generative AI models are trained for a specific task. Different models and frameworks exist that conditionally generate sounds [13, 116], others that morph sounds [115] or generate sounds using text-based controls [131, 132]. In lieu of designing newer algorithms specifically designed to cater to every new task, in this thesis, we outline an approach to extend the functionality of existing, pre-trained large foundational audio models through a method we term "exaptation for creativity support."

With origins in evolutionary biology, the term exaptation refers to repurposing an idea, concept, or system for newer uses that expand on its original purpose. In Chapter 4, we leverage a TTA model trained to generate sounds using a single text prompt to expand its use to morph two or more sounds without additional training or fine-tuning procedures. Similarly, in Chapter 6, we leveraged a pre-trained model trained on multiple classes of environmental sounds and developed class-based "soft constraints" [160] for exploration. Using these soft constraints, the designers could target their exploration to a part of the latent space oriented towards those constraints, thus enabling a more focused and deeper exploration. In both these cases, AI-based CSTs designed specifically for generating sounds were *"exapted"* for some additional creative tasks such as sound morphing or constrained exploration.

In light of the recent environmental impact [252] due to the training of large generative AI models, we suggest future CSTs, for all modalities including sound design, could make use of such exaptation techniques over existing large foundational pre-trained models by leveraging such post-hoc methods for interactivity instead of training newer models for that specific purpose.

7.1.3.3 Visual Approaches to Designing and Evaluating Sounds

Another key proposition of this thesis is to design CSTs and perceptual listening test interfaces for novices and experts. One of the approaches outlined in this thesis involves using visual spectrograms to demonstrate the generated audio output on a CST. We also develop a method to use visual metaphors of image-schemas [203] to visually articulate the quality of sound progression in a listening test with non-audio experts.

In Chapter 6, CSTs used in the study with expert practitioners visualized the spectrogram of the generated sound. As the interactive control panels on the interfaces modified the spectromorphology [101] of the sound (or how the frequencies in the sound change or morph over time), the visualizations were meant to provide feedback to the designers on the effect their parameter changes had on the synthetic sounds they were editing. Interestingly, our interviews found that these visualizations provided means for the designers to describe their creative goals in spectromorphological terms. Sound design experts defined the use of the controls in visual terms, such as "seeing sound events", or defining the impact sounds in terms of its "long body and tail" etc. (see Chapter 6, Section 6.5.3). Further, designers used the spectrogram visual as a proxy for listening, especially while performing rapid edits and explorations using the CSTs. They found it easier to visually observe the effect of their parameter changes on their creative explorations instead of stopping and listening to every small change they made.

Thus, we propose that using such visual representations of the sounds can reduce the cognitive load associated with making small edits to the generated sounds, especially when doing exploratory work. Previously, researchers in the explainable AI (XAI) for arts [75, 253] used latent space visualizations to *explain or debug* their creative goals. We build upon this work and suggest that spectrogram visualizations could provide a great way for designers to communicate their creative goals and understand AI-based CSTs' output.

Similarly, in Chapter 5, static visual metaphors were used to describe the temporal quality of audio in a crowdsourced listening test with non-audio experts. The visual metaphors were also used to design interfaces for complex rank ordering and proximity spacing tasks. Using such metaphors enabled us to collect better-quality responses in both cases.

Previously, Cartwright et al. [29] demonstrated that when using visual representations of sounds, they collected better annotations for sound events than audio alone. Visual representations of the sounds allowed annotators to 'glance-and-click' on the sound events while listening, which improved the accuracy of the collected annotations. Similarly, in

our experiments, we used visual metaphors to employ language and experience-agnostic conceptual models [202] to define audio quality. We thus expand on prior research and suggest that using the metaphors of image-schemas provides a visual means to understand temporal audio quality concisely.

7.1.3.4 Creative Engagement with AI-based CSTs

In Chapter 6, we qualitatively studied how AI-based CSTs can assist sound designers in their creative work. Sound design practice is highly technical and artistic in nature [2], thus such practitioners know how to work with digital sound editing tools and also have a keen artistic understanding of developing sound palettes and background scores associated with the film or game. Thus, one of the central tenets of this thesis was to evaluate AI-based CSTs for sound design for their creative engagement and use-qualities [53, 89] of pliability and ambiguity rather than metrics such as efficiency or error-free performance.

The novel approaches for exploration outlined in this thesis enable a pliable exploration of the AI-generated design space using synthetic sounds (or sonic sketches as described previously). The advantages of this pliability use-quality are demonstrated by the designers' use of an AI trained on simple impact sounds to rapidly and iteratively explore the design space to generate cinematic effects that perceptually resembled trilling cicadas, footsteps, sci-fi whooshes, and sounds made by fantastical monsters (see Section 6.4.3.1). In comparison, AI-based systems trained using labels in a supervised way are tightly coupled to the semantics in the labeled data and may not be as pliable or malleable for the purpose of exploration. We therefore recommend using pliable exploration techniques, such as those based on sonic sketches, for better creative engagement with an AI-based CST.

When using the CST with the element of ambiguity [89, 91] in the design of the dimensions in Chapter 6, different participants came up with different semantic or acoustic explanations for the effect of each dimension on the edited sound. Ambiguity in controls forced the participants to engage closely with the CST and participate in meaning-making. This led to participants developing varied exploration techniques, as well as personalizing the dimensions based on their own understanding of its effect. Typically, non-determinism is considered to be detrimental to the user experience of an AI system [90]. In contrast to such prior work, we found that ambiguity in design allowed for more serendipitous discovery of novel elements of sound (Section 6.4.1.4) and made the exploration more pliable or open (Section 6.5.4) for the sound designers during their creative task in the study. Furthermore, the designers found ways to work around this

ambiguity by performing creative exploration as a separate focused task and building inventories of novel sounds from the exploration for use in their creative projects.

7.2 Limitations

7.2.1 Semantic Exploration for Music and Other Sounds

The "Example-Based Framework" (EBF) in this thesis derives semantic guidance vectors using synthetic sound sketches to perform semantic exploration in the latent space of a StyleGAN. While we demonstrated the efficacy of the EBF method for perceptually guiding the generation of audio textures, a potential limitation of extending this approach to other sound types is in the parametric synthesizer (in Chapter 3, Section 3.3.3). Our current parametric synthesis technique is limited by its ability to model sounds based on object resonances or physical parameters of the interacting objects. Newer approaches must be developed to approximate the synthetic sound queries needed for navigating the latent space of other sound types, such as timbres from musical instruments and speech. A potential avenue for such procedural synthesis models can be found in [225].

7.2.2 Approaching EBF-like Semantic Edits using Text-to-Audio models

The two technical contributions from this thesis, the EBF method, and the MorphFader method outline novel ways to semantically edit sounds in a fine-grained way. While EBF operates on smaller GAN-based models trained on a targeted range of sounds, MorphFader, on the other hand, can semantically edit sounds generated by a large TTA model. Although we developed two methods to perform such edits on the generated sounds, in this thesis, we do not perform experiments to systematically compare them to each other.

In our experiments for evaluating EBF in Chapter 3, we restricted ourselves from comparing our method with TTA models because the datasets we used in the study were unlabelled and not associated with text captions needed to train TTA models. Further, we refrained from using off-the-shelf TTA models for comparison as their training data (such as Audioset [152]) significantly differed from the training data distribution

under the purview of the EBF method. Although we are unable to perform a systematic comparison of our method with TTA models, on our supplementary webpage[1] we demonstrate some text prompts that assist in achieving the semantic editing goals of our framework using TTA models such as AudioGen [256] and AudioLDM [132]. For impact sounds, we designed prompts by describing the material properties of the impact surface and certain acoustic properties of the sound. Similarly, we described the container's material properties and fill level for water filling.

While it should be noted that well-engineered prompts would lead to better results, with the prompts we used (on our supplementary page), we observed that editing a prompt considerably changed not just the semantic attribute being edited but also other attributes of the sound. For instance, modifying an existing prompt by adding a Rate feature such as 'fast' considerably changed other aspects of the sound, such as Brightness, and removed the 'long sustain' from the originally prompted sound. This could be because, in TTA models, text prompts could be entangled with multiple semantic attributes of the sounds. To alleviate this effect, in Chapter 4, we elaborate on semantic word-weighting to emphasize certain word descriptors while morphing two or more sounds. Although word-weighting could be used to edit semantics granularly, such methods are limited by the existing text captions in training datasets and might not be useful when modifying user-defined semantics such as "brightness" as done with EBF. Further systematic experimentation is needed to study these methods for editing any user-defined semantics on the generated sounds, semantics that perhaps do not exist (or are out of distribution) of the training data text captions. This study could be conducted in comparison to or in conjunction with sound-based frameworks like ours.

7.2.3 Perceptually Evaluating Sounds using Visual Constructs.

In Chapter 5, we introduced novel visual constructs of image-schemas to evaluate sounds generated by steerable audio models. We designed and implemented interfaces using such stationary constructs to communicate the temporal quality of audio in a listening test on crowdsourced platforms such as Amazon's Mechanical Turk (AMT). In this section, we outline some limitations of this work.

Use of Modern Browser Features. All experiments in Chapter 5 rely on the availability and use of modern internet browser features such as HTML5 components, drag/drop,

[1] https://purnimakamath.com/thesis-related/chapter_3/#t-to-a

sliders, CSS gradients, and variables. Experiments with the visual image-schema constructs will not render correctly without these features. Therefore, before participants started working on the tasks in our experiments on AMT, we performed browser checks and allowed only participants who used compatible browsers to attempt our trials. Given the diverse pool of crowdsourced participants with varying desktop and browser installations, it is difficult to ascertain how many qualified participants could not attempt our tasks because of this limitation.

Generalizability to Other Datasets. There are some limitations to generalizability due to the datasets we use in these experiments. In this paper, we choose a set of sounds from varied sources - pitched instrument sounds from an existing benchmarked dataset [79], noisy textures [223], a set of programmatically synthesized pitched sounds [225] and recorded sounds (water filling a container). While we try to cover multiple sources of sounds - from noisy textures to sounds with events and pitched instrument sounds - more experimentation with larger datasets is needed to systematically generalize these results to other sounds.

Development of image-schemas. This thesis explores using the *source-path-goal* image-schema for articulation and interface design as both experiments evaluated sound progression. There is a need to design and develop new icons and interfaces depending on the evaluation measure in a listening test. Further, more exploratory research is necessary for evaluating audio qualities of stationary sounds using other visual metaphors such as *containment* or *linkage* [202–204] etc. A potential future avenue for research surrounds the development of a generalized, pluggable web interface or visual library for metaphors for such audio evaluations.

7.2.4 Sound Design Practice with Rapidly Evolving Generative Models Landscape

In Chapter 6, we investigated how sound designers can use generative audio models in their creative practice. Generative audio research is evolving rapidly with newer innovations in building larger, faster, and better-quality generative audio architectures. While we use StyleGANs [126] to build our CSTs, alternatives based on model architectures such as Diffusion [132] are emerging as potential alternatives. Although we have tried to keep our inferences on assessing the potential of generative models for sound design free from any technical constraints or usability issues, new modes of interactivity will change how designers perceive and use such models. Therefore, more research will be needed

in the future to understand how the practice of sound design evolves along with newer models.

We conducted our study with nine professional sound designers from diverse geographic, years of experience, and sound design backgrounds. With this, although we present a rich description of how such CSTs can be used by sound designers in a work setting, given the qualitative nature of our study, our findings might not generalize to broader populations. Further, the study was conducted where the participants used the CSTs for only a few tasks. Our future work will focus on capturing patterns of usage as well as studying the different parameter exploration strategies in depth in a professional work setting, over longer periods, and in various phases within the sound design project cycle.

Finally, in our study, we focused on exploring the strategies used by sound designers to discover new sounds when using AI-based CSTs designed with pliability and ambiguity. However, it is also important to study these exploration strategies based on the background, years of experience, and types of projects developed by sound designers. For example, we need to investigate whether sound designers who typically create sounds for fantasy movies by editing or "bending" existing sounds make more use of ambiguity in interfaces compared to designers who do not work on such projects. Additionally, we need to examine how this ambiguity affects the sound designers' experience with these interfaces based on their years of experience, types of projects, and their design background (such as designing sounds for games, films, or products).

7.3 Future Work

7.3.1 Real-Time Generation

Although the algorithms outlined in Chapters 3 and 4 can generate sounds based on user-defined semantics, they can generate sounds only of a pre-defined length. For instance, the examples we demonstrated in this thesis were either two seconds (for the Example-Based Framework), four seconds (DCASE Challenge), or ten seconds (for MorphFader) in length. As discussed in Chapter 2, this pre-defined duration or length of sound generation is due to using architectures, such as StyleGANs and Latent Diffusion Models (LDMs), that train on 2D spectrogram representations. Certain applications, such as those in the music domain, need continuous sound generation with fine-grained semantic control in real-time. Such real-time synthesis is useful in applications of musical performances and data sonification experiments [57].

As outlined in Chapter 2, RNNs and Transformer architectures are designed to autoregressively generate samples based on past states. While such models are effective in the next sample generation and can be used to continuously generate sounds indefinitely, training them on raw audio samples is prohibitive memory and performance-wise. Recently, audio encoding algorithms such as Encodec [257], SoundStream [258], and Descript Audio Codec (DAC) [259] have been able to generate embeddings for general-purpose audio, which can be used as an alternative to raw audio when training Transformers and other autoregressive models to generate sounds in real-time. Developing algorithms that generate sounds of indefinite lengths, assist in semantic discovery, and perform sound edits and morphs with parametric response times [116] will be a productive avenue for future work.

7.3.2 Multi-Dimensional Interactivity for Morphing Interfaces.

In Chapter 4, we developed and evaluated a method for morphing and semantic word-weighting sounds generated using TTA models. A productive avenue for future work is to improve the proof-of-concept interfaces developed in this thesis and study the usability and applicability of such interactive techniques for sound design.

The interfaces for MorphFader currently provide means to continuously interpolate between two prompts or perform sound edits using fader-like controls. Such fader controls resemble the knobs or tracks in most Digital Audio Workstations (DAWs). With the increasing sophistication of generative algorithms for sound generation, novel interfaces are needed to encode the multi-dimensional interactivity enabled by the algorithms. Previously, Tubb et al., [260] evaluated multi-dimensional ways to build *expressive* (or fine-grained control of perceptual properties) interfaces for sound creation or editing. They proposed a 2D XY touchpad and 3D Leap Motion-based controls for such creative tools and found that creators could successfully use such controllers for their creative work (although with some practice). Further, Wyse et al., [116] create a new musical instrument they call a "Trumpinet" by morphing the timbres of a Trumpet and a Clarinet to create musical compositions. They explore a GAN-generated 2D sound space guided by the timbre and pitch of the two instruments to generate individual pitches for the new instrument for their compositions.

We take inspiration from such 2D and 3D interfaces and propose future work to develop morphing interfaces for sound design, such as in Figure 7.1. By morphing between prompts along the X-axis and weighting between their semantic word descriptors for

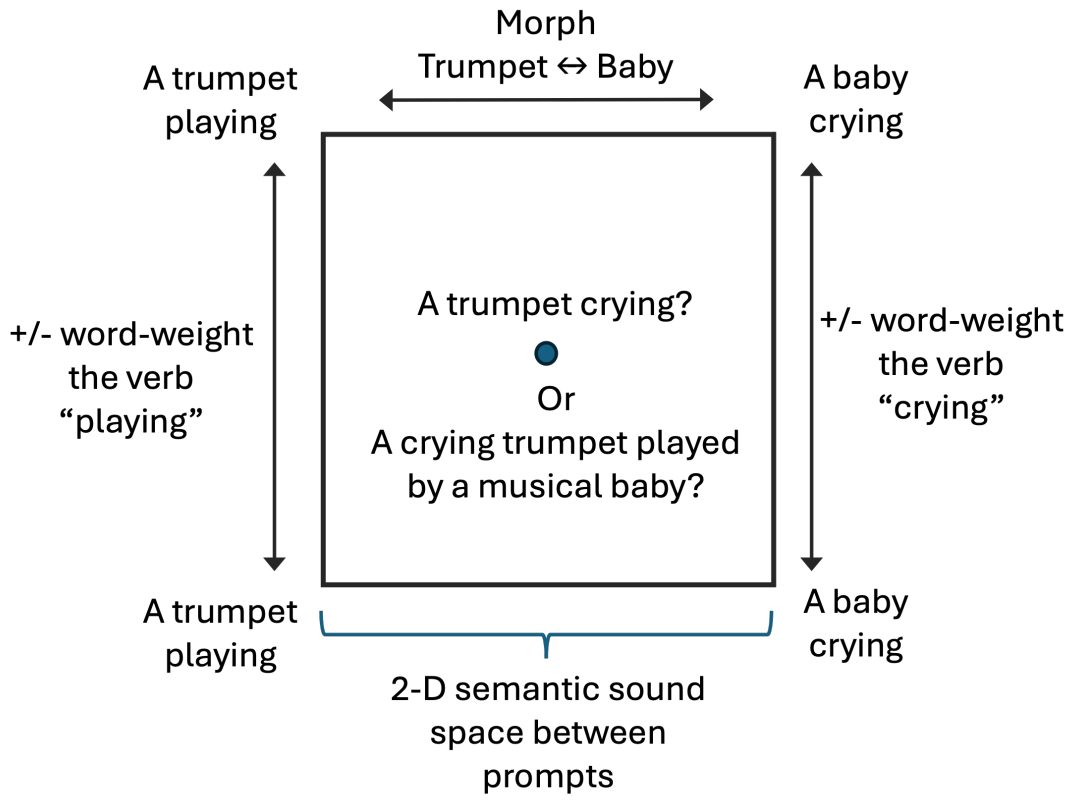


FIGURE 7.1: Future work: A conceptual sketch for the design of a 2D interface to explore the semantic sound space generated using text prompts and TTA models.

each prompt along the Y-axis, we can provide sound designers better opportunities to explore the semantic sound space between text prompts and their word descriptors than when using simple 1D interfaces. Furthermore, this idea can be extended to 3D interfaces, where the X-Y dimension can morph between multiple text prompts, and the Z-axis can be used to weight their semantic word descriptors.

While such 2D or 3D interfaces provide effective sound space exploration strategies, a few issues arise when implementing them. The current TTA models (AudioLDM) used in MorphFader take 2-3 seconds to generate semantically word-weighted sounds and around 5-10 seconds to generate morphs. This delay in sound generation and exploration may impact the user's experience using these 2D or 3D CSTs. While the sound designer can instantly click on any point within the 2D interface, there will be a delay until they can listen to the sound and further explore the space iteratively. Further, MorphFader explores linear trajectories within the design space between two text prompts. For a better interactive CSTs, other exploration strategies, such as spherical interpolation, must be evaluated and enabled within this design space.

7.4 Final Remarks

In this thesis, we investigated novel human-centered approaches to designing, implementing, and evaluating steerable generative models and Creative Support Tools (CSTs) for sound design. Through this work, using steerable audio models, we designed affordances on CSTs that enabled sound designers to convey their creative goals based on semantically relevant attributes or properties of the environmental sounds they want to generate. Furthermore, we enabled support for creative tasks such as sound morphing to allow designers to explore the sound space generated by the generative models in novel and creative ways. We developed ways to perceptually evaluate the steerable models and their creative output using subjective listening tests on crowdsourced platforms. Finally, we studied the challenges and opportunities of applying such models in a practice-oriented sound design environment.

While the research and literature on generative models for audio is growing, we believe the new perspectives and human-centered approaches outlined in this thesis will provide a conceptual and theoretical foundation for future researchers working at the intersection of human-centered AI and sound design.

Appendix A

Supplementary Material - Technical Architectures

A.1 GAN Loss Functions

In Chapter 2, Figure 2.2, we introduced G as the generator and D as the discriminator or the critic. Further, $z \in \mathcal{Z}$ is the latent space vector. The generator G is trained to fool the generator by creating samples that closely resemble real data by minimizing $log(1 - D(G(z)))$. Thus, training a GAN is said to be the minmax game between G and D over the function —

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \in X}[log\, D(x)] + \mathbb{E}_{z \in \mathcal{Z}}[log(1 - D(G(z)))] \tag{A.1}$$

That is, during backpropagation, we update the weights on the Discriminator network by performing *gradient ascent* (to maximize the loss) and *gradient descent* on the weights of the Generator (to minimize).

To stabilize the GAN training procedure, [261, 262] propose using Wasserstein distance or "Earth movers distance" metric along with a gradient penalty (a regularization term) while computing the discriminator loss. The discriminator loss is thus formalized as —

$$L = \underbrace{\mathbb{E}_{z \in \mathcal{Z}}[D(G(z))] - \mathbb{E}_{x \in X}[D(x)]}_{\text{Discriminator Loss based on Earth Movers Distance}} + \underbrace{\lambda\, \mathbb{E}_{\hat{x} \in \hat{X}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2]}_{\text{Regularization term}} \tag{A.2}$$

Where $\hat{X}$ is a distribution of uniformly sampled linear combination of training data and data sampled using the generator for the same points. In this thesis, we do not modify the GAN loss functions. We use this loss formulation throughout this thesis and different versions of GANs (as seen in the subsequent sections). A more nuanced and detailed description of the loss formulations can be found in the original literature, such as [138, 261, 262].

A.2 DCASE Challenge Technical Report

A.2.1 Introduction

For the DCASE Challenge 2023 Task 7 (Track B), Foley Sound Synthesis, we submit two systems, (1) a StyleGAN conditioned on the class ID, and (2) an ensemble of StyleGANs each trained unconditionally on each class separately. We quantitatively find that both systems out-perform the task 7 baseline models in terms of FAD Scores. Given the high inter-class and intra-class variance in the development datasets, the system conditioned on class ID is able to generate a smooth and a homogeneous latent space indicated by the subjective quality of its generated samples. The unconditionally trained ensemble generates more categorically recognizable samples than system 1, but tends to generate more instances of out-of-distribution or noisy samples.

Generative audio algorithms using deep neural networks aim to generate novel audio that matches naturally occurring sounds in their qualities such as realism or plausibility of the sound. Recently, there has been a focus on developing such models for inharmonic sounds such as those of environmental audio. Such synthesis models are useful for generating background environmental sound scores for movies, games, and automated Foley sound synthesis. The task in DCASE Foley Sound Synthesis challenge [127] this year is to generate sounds of seven sound classes with high fidelity and diversity. There are two tracks - tracks A and B - in this challenge, each using a curated dataset and with or without external resources outlined on the challenge webpage[1]. Our submission is for **track B**, i.e., using only the development dataset and without the use of any external resources (audio data or pre-trained models).

For our submission, we use a type of Generative Adversarial Network (GAN) [138] called StyleGAN2 [125, 126] trained from scratch on log-magnitude spectrogram representations of the environmental sounds in the dataset. Generally, GANs learn a distribution of the

[1]https://dcase.community/challenge2023/task-foley-sound-synthesis

TABLE A.1: DCASE Challenge System Details

	Class	w/z-**dim**	**No. Mapping Layers**	**Training Iters (kimgs)**	**Training time (∼days)**
System 1 (Conditional. One model, all classes.)	All Classes	512	8	1200	2.125
System 2 (Unconditional. Individual models for each class.)	Dog Bark	128	4	1600	2.5
	Footstep	128	4	2600	4.7
	Gunshot	128	4	3200	5
	Keyboard	128	4	2200	3.95
	Moving Motor Vehicle	128	4	800	1.29
	Rain	128	4	1600	2.4
	Sneeze/Cough	128	4	1800	3.79

sounds in the dataset, such that random sampling within the learned latent space generates novel audio samples matching the fidelity of the real-world training data. StyleGANs are designed to further improve the quality of the generated sounds by better disentangling the factors of variations observed in the dataset using an intermediate latent space. Such architectures are inspired by the style transfer tasks and learn the intermediate latent space using a set of affine transforms called the mapping network.

We submit two systems for this challenge - (1) System 1: A conditional StyleGAN2 trained on the entire development dataset and conditioned on the class-IDs using one-hot encoding, and (2) System 2: An ensemble of unconditionally trained StyleGAN2 networks, one for each class of sounds in the dataset. We empirically decide the values for certain hyperparameters of the StyleGAN2 architecture (e.g., the dimensionality of the latent space and the number of layers in the network) depending on the number of classes being modeled in the system. We report the Fréchet Audio Distance [39] scores for each class per system on the training set. We describe each system in detail in the following sections.

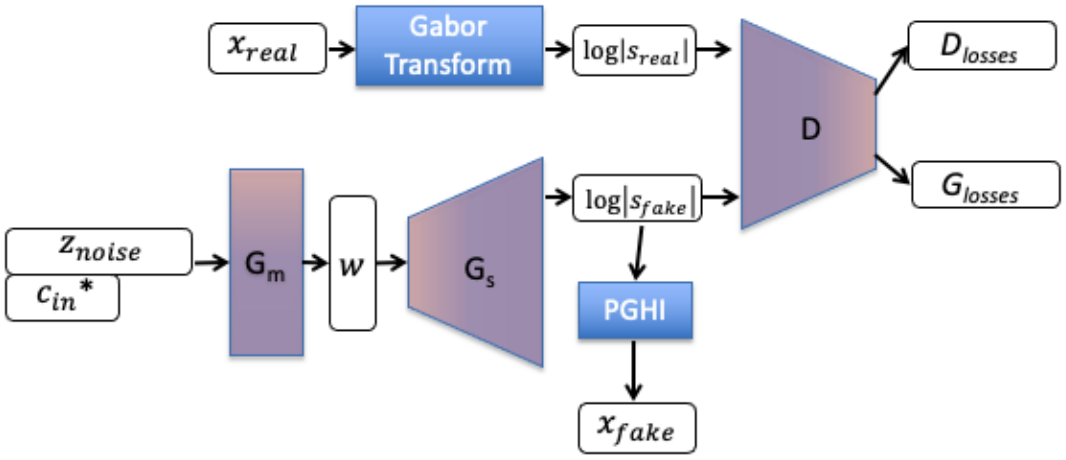


FIGURE A.1: A schematic outlining the main components in our submission for both System 1 and System 2. Conditioning vector c^*_{in} is applied only to System 1.

A.2.2 System Overview

Figure A.1 illustrates the main components within our submission. We use StyleGAN2 in conjunction with log-magnitude spectrogram representations generated using Gabor transforms [168]. Previously, Gupta et al. [110] showed that using the phase gradient heap integration method (PGHI) [111, 168] for phase reconstruction during spectrogram inversion is an effective way to reconstruct sharp and clear transients in the resulting sounds. As most of the environmental sounds in the development dataset in this challenge include sound events with sharp attacks and transients (such as Dog Barks or Footsteps), we use the Gaussian windowed log-magnitude spectrogram representations during training and PGHI for high-fidelity spectrogram inversion in conjunction with StyleGAN2.

We use StyleGAN2 from Nvidia's official codebase[2] and adapt it to train using audio spectrograms. In this report, we elaborate mostly on the Generator of StyleGAN2 as most of our changes to the official repository focus on that component. As shown in Figure A.1, we use the Gaussian windowed log-magnitude Short-time Fourier Transform (STFT) of an audio sample x_{real} to train the GAN. The aim of the generator is to synthesize an audio sample x_{fake} which resembles x_{real}. The generator samples from the Z latent space to synthesize x_{fake}. Specifically, a StyleGAN2's generator can be modeled as a two functions - a mapping network or a set of fully connected layers $G_m(.)$ that maps a d-dimensional latent space $z_{noise} \in R^{d_z}$ to an intermediate $w \in R^{d_w}$ space

[2]https://github.com/NVlabs/stylegan2-ada-pytorch

and a synthesis network $G_s(.)$ that maps the resulting w vector to the spectrogram space $s \in R^{f \times t}$. Here d_z, d_w is the dimensionality of the Z and $\mathcal{W}$ space respectively. And f, t are the number of frequency channels and time frames of the generated spectrogram.

A.2.3 Experimental Setup

For System 1 (conditional), we set both d_z and d_w to 512. The number of fully connected layers in the mapping network G_m (or the number of affine transforms before w vectors are generated) is set to 8. For System 2 (unconditional ensemble), we set d_z and d_w both to 128 and number of layers in the mapping network to 4. Further, we use a batch size of 4 or 8 (depending upon the resource availability on our shared compute infrastructure during training) to train the networks.

All our models were trained either on a single RTX 3090 24GB GPU or the National University of Singapore's high-performance compute infrastructure (shared single Nvidia Tesla V100 32 GB GPU). The training details with respect to the number of epochs or iterations and the time taken are outlined in table A.1.

A.2.3.1 Dataset

For this task, we used only the development dataset outlined in the challenge description [127, 128]. The dataset consists of environmental sounds from 7 classes. Classes such as Dog Bark, Footstep, and Gunshots contained multi-event sounds with sharp transients, whereas classes such as Rain or Motor Vehicle contained more noisy sounds. Each sound sample was 4 seconds long and sampled at 22,050 Hz. We generate the Gaussian windowed log-magnitude spectrogram with stft_channels = 2048, n_frames = 1024 and hop_size = 128.

A.2.3.2 Data Augmentation

GANs are powerful generative architectures but need large datasets to model the distributions effectively. The number of samples per class in the development dataset was very small, with an average of ∼46 minutes per class. We thus augmented our development dataset using one of two simple strategies - zero-pad, and wrap-around, before training our unconditional System 2. Note that no data augmentation was done for the conditionally trained System 1.

For all audio samples in training that contained events lasting less than 2.5 seconds (detected by simply thresholding), we applied the zero-pad augmentation strategy. On closer observation of the nature of the audio samples under each class, multiple samples had sound events lasting only a few seconds with zero-padding for the remainder of the sample (e.g., some Dog Barks and Gunshot samples). To augment such samples, we shifted the sound events along the right of the time axis, while padding the beginning of the sample with zeros. For sounds that had events lasting more than 2.5 seconds (e.g., Moving Motor Vehicle), we used the wrap-around strategy where we simply wrapped around and shifted the samples along the time axis after removing the padded silences during augmentation. We applied these augmentations to each audio file 10 times, which augmented our training data by a factor of 10 for each class.

A.2.3.3 Evaluation Methodology

We use the Fréchet Audio Distance(FAD) [39] to evaluate the quality of our synthesized audio for both systems. This metric measures the distance between the distributions of training data and the synthesized audio based on their VGGish embeddings. We synthesized 100 samples for each class and computed the FAD score against the entire training set for that class. Further, this score was computed for multiple checkpoints during training. We selected 2-3 checkpoints based on best FAD scores and then subjectively evaluated by listening (internally within the research team) to the synthesized audio for artefacts such as smearing of the attack transients in the samples and recognizability of the sounds. We eventually selected the model which generated more recognizable sounds than others and qualitatively preserved the transients for submission for this task irrespective of their FAD scores.

A.2.4 Results & Discussion

Table A.2 shows the FAD scores for both System 1 and 2. Standard error of means computed by bootstrapping 10 times. Scores marked with * are higher than the baseline in the task. Mean FAD scores for System 1 and 2 were 6.50 and 4.02 respectively.

TABLE A.2: DCASE Challengen FAD Scores

	Class	**FAD Scores(↓)**
System 1 (Conditional)	Dog Bark	5.34 ± 0.76
	Footstep	5.06 ± 0.34
	Gunshot	9.98 ± 0.66*
	Keyboard	3.94 ± 0.26
	Moving Motor Vehicle	14.26 ± 0.82
	Rain	5.30 ± 0.52
	Sneeze/Cough	1.65 ± 0.08
System 2 (UnConditional or per-class)	Dog Bark	3.80 ± 1.09
	Footstep	3.30 ± 0.21
	Gunshot	4.40 ± 0.36
	Keyboard	3.38 ± 0.18
	Moving Motor Vehicle	7.05 ± 1.27
	Rain	4.21 ± 0.38
	Sneeze/Cough	2.02 ± 0.11

While System 2 (unconditional ensemble) organizes its latent space according to the variances in each individual class (intra-class variance), System 1 (conditional) has an additional task of organizing its latent space according to both inter-class as well as intra-class variances in the dataset. The implications from this on the quality of generated sounds is two-fold - (1) though System 2 shows lower FAD scores than System 1, the latent space generated by System 2 has 'holes' [263] in the latent space which generate out-of-distribution (OOD) or noisy sounds. This nature of the latent space can be attributed to the high intra-class variance in the sound samples in the training set. (2) Although System 1 does not generate many OOD sounds and has a homogenous or smooth latent space as compared to System 2, it generates more sounds which can be subjectively

mis-categorized (i.e., the 'holes' in the latent space are filled with sounds from another class or category). For instance, some System 1 synthesized Gun Shot sounds, such as machine gun sounds, sound like Keyboard clicks. In this regard, System 2 generates more categorically recognizable sounds.

Further, while training System 1 (conditional), we observe that all classes do not train equally through the training iterations. While training for longer epochs, some classes, such as Dog Barks, tend to overfit while other classes such as Gun Shots are still generalizing to the distribution.

A.2.5 Limitations

The StyleGAN2 architecture was originally developed to a learn latent distributions for images. As such, this architecture trains using square (same height and width)images. To adapt this architecture to audio, we design square spectrograms by zero padding the raw audio and selecting a specific number of frequency channels and time bins. Our future work will involve modifying this architecture to use spectrograms of any number of frames and frequency channels.

Appendix B

Supplementary Material - Understanding Opportunities for Generative Models in Sound Design

B.1 Semi-structured Interview Questions

As discussed in section 6.3.2, our interview consisted of three parts:

- Participant's background and experience: Through these questions, we focused on capturing the participant's experience with sound design
  - Can you describe some of the projects that you typically work with?
  - Can you describe with an example some of the typical tasks you perform while designing sounds?
  - What is the most annoying part of your design process?
- Their expectations of generative AI: These questions captured the participant's outlook and past experience with generative AI.
  - What do you feel about AI?
  - What were your expectations from this AI-based sound design tool?

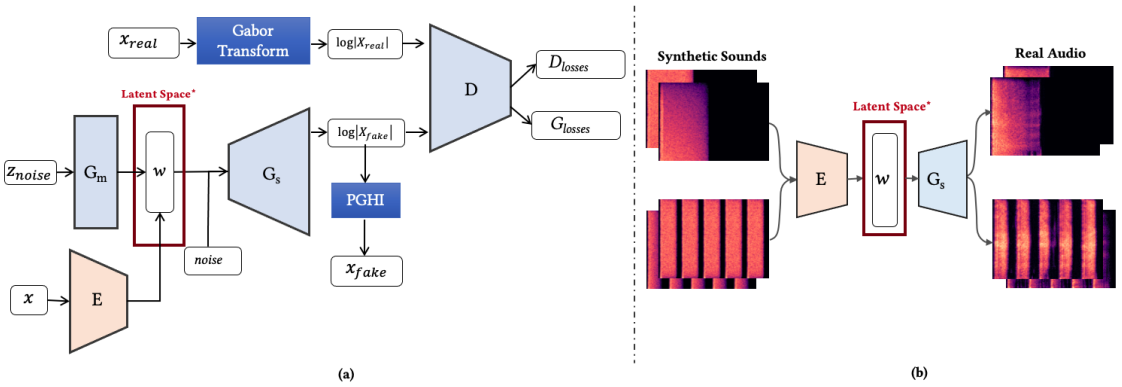


FIGURE B.1: Architectural components driving the audio AI interfaces used in the study

  - Can you describe an ideal AI-based tool for sound design?

- Based on the creative task: Through these questions, we capture the participant's experience and feedback using the AI-based sound design tools in this study. For this part of the interview, we use the screen recordings as discussion prompts.
  - Can you explain what you wanted to do in the open-ended task and how did you go about achieving it?
  - Did the outcomes from your tasks match your expectations?
  - How do you think such AI-assisted sound design tools fit into your design process?
  - What did you find most frustrating to do?
  - What do you want AI-assisted sound design tools to do more?

B.2 AI-based CST Architecture Details

The two interfaces in this study, interface-1 which uses domain-specific controls, and interface-2 which uses technology-specific controls, use the same underlying trained StyleGAN. The StyleGAN architecture is shown in Figure B.1 (a), where G_s is the generator (synthesis network), G_m is the mapping network, and D is the discriminator. E is a GAN inversion network adapted from [184] for interface-1. A StyleGAN2's generator can be modeled as a function $G(.)$ that maps a latent space $\mathcal{Z}$, where $\mathbf{z} \in \mathbb{R}^{\delta_z}$, to the higher dimensional spectrogram space $\mathbf{S} \in \mathbb{R}^{f \times t}$, such that $\mathbf{S} = G(\mathbf{z})$. Here δ_z is the dimensionality of the $\mathcal{Z}$ space and f, t are the number of frequency channels and time frames of the generated spectrogram respectively. StyleGANs further learn an intermediate representation $\mathcal{W}$, where $\mathbf{w} \in \mathbb{R}^{\delta_w}$, between that of $\mathcal{Z}$ and $\mathcal{S}$ via a mapping network $G_m(.)$.

This intermediate latent space further disentangles factors of variation as compared to the latent $\mathcal{Z}$ space [126]. Further, a synthesis network $G_s(.)$ maps the $\mathbf{w}$ vector to a spectrogram $\mathbf{S}$. Note that in this paper, whenever we refer to the term "latent space", we mean the $\mathcal{W}$-space generated by the mapping network G_m.

We set $\mathcal{Z}$ and $\mathcal{W}$ space dimensions δ_z and δ_w both to 128 and use 4 mapping layers in the Generator for all our experiments. Further, we use the log-magnitude spectrogram representations generated using a Gabor transform [168](n_frames= 256, stft_channels= 512, hop_size= 128), a Short-Time Fourier Transform (STFT) with a Gaussian window, to train the StyleGAN2, and the Phase Gradient Heap Integration (PGHI) [111] for high-fidelity spectrogram inversion of textures to audio [110]. All sounds generated using both interfaces were normalized to -14dB for loudness using pyloudnorm [264]. The codebase for the interfaces, StyleGAN, and Encoder is on GitHub as follows:

- Both interfaces in this study: `https://github.com/augmented-human-lab/audio-design-toolkit`
- StyleGAN: `https://github.com/pkamath2/audio-stylegan2`
- GAN Encoder: `https://github.com/pkamath2/audio-latent-composition`

A Google Colaboratory version of our interactive Creative Support Tools can be found here: `https://pkamath2.github.io/chi2024-resources/`

B.2.1 Interface-1

Apart from StyleGAN, interface-1 is powered by two additional components: (1) a GAN Encoder or inversion framework, and (2) a synthetic sound generator. The code for both is adapted from [184]. While GANs map the latent space to real-world sounds, GAN Encoders learn the inverse, i.e. they map real-world sounds to latent space embeddings. This technique is especially useful when we want to "query" or "search" sounds within the latent space using approximations (or synthetic sounds). The synthetic sounds are generated by passing Gaussian noise $\mathcal{N}(0, I)$ through band-pass and fade filters. The parameters to generate the synthetic sounds are actualized as sliders on the interface. Figure B.1 (b) shows a conceptual diagram for the components behind interface-1. The synthetic sounds are encoded into the latent space to derive their $\mathcal{W}$-embeddings. These embeddings are then passed through the generator to generate realistic AI-generated sounds matching the synthetic sounds.

B.2.2 Interface-2

We utilize the semantic factorization algorithm (SeFa) [154] to derive technology-specific controls from the latent space of the StyleGAN in this study. The SeFa method is a closed-form, unsupervised method for latent semantic discovery. It decomposes the pre-trained weights of G_m of the StyleGAN using eigendecomposition to find vectors for controllability. The SeFa algorithm returns δ_w (128 in our case) dimension vectors and their corresponding singular values. We fetch the top 10 vectors (vectors with the highest singular values) and display them on the interface. The vectors are actualized as sliders on the interface for users to interact and perform edits directly in the latent space of the GAN.

Both interfaces were developed using Streamlit and ReactJs. Streamlit is a Python library that enables frontend applications to connect to Python-based machine learning models easily. The ReactJs-based frontend communicates with the Python backend using Websockets.

B.3 Acoustic Parameters on Interface-1

The list of acoustic parameters on the interface-1 are:

- Impulse width: Parameter value decides how long the impact sound 'rings' or lasts along the time axis.
- Rate: Controls the number of impact events in the sound along the time axis.
- Frequency band: Frequency range of the bandpass filters. Controls the brightness of the sound. Higher frequencies sound brighter, such as impact sound on a hard metal surface. Lower frequency ranges sound duller, such as impact sounds on soft materials such as a cushion or a sofa.
- Filter order: Determines the frequency roll-off. Used in conjunction with the frequency band. Higher filter orders have a steeper roll-off and transition between the frequency bands.
- Fade In: Controls how the sound transitions from zero to full strength.
- Fade Out: Controls how the sound transitions from full strength to zero.

B.4 Attribution for icons and images

The author created most of the images used in this chapter using various drawing tools. Some visual icons were sourced from the following websites:

- In Figure 6.1: the sound designer icon is sourced from Flaticon.com; the domain-specific controls icon is sourced from a "slider" icon by Inggit Jaya from thenounproject.com.
- In Figure 6.2: the sound designer icon is sourced from Flaticon.com;

Bibliography

[1] Sandra Pauletto. The voice delivers the threats, foley delivers the punch: Embodied knowledge in foley artistry. In *The Routledge Companion to Screen Music and Sound*, pages 338–348. Routledge, 2017. 1

[2] Leo Murray. *Sound design theory and practice: Working with sound*. Routledge, 2019. 1, 15, 57, 116, 137

[3] Michel Chion. *Audio-vision: sound on screen*. Columbia University Press, 2019. 1, 16

[4] Patrick Susini, Olivier Houix, and Nicolas Misdariis. Sound design: an applied, experimental framework to study the perception of everyday sounds. *The New Soundtrack*, 4(2):103–121, 2014. 1, 2, 4, 15, 16, 106, 122

[5] David B Anderson and Michael A Casey. The sound dimension. *IEEE spectrum*, 34(3):46–50, 1997. doi: 10.1109/6.576008.

[6] David Moffat, Rod Selfridge, and Joshua D Reiss. Sound effect synthesis. In *Foundations in Sound Design for Interactive Media*, pages 274–299. Routledge, 2019. 1, 2

[7] David Lewis Yewdall. *The practical art of motion picture sound*. Routledge, 2012. 1

[8] Budhaditya Chattopadhyay. Reconstructing atmospheres: Ambient sound in film and media production. *Communication and the Public*, 2(4):352–364, 2017. doi: 10.1177/2057047317742171. URL `https://doi.org/10.1177/2057047317742171`. 1

[9] Anna Wiener. The weird, analog delights of foley sound effects, 2022. URL `https://www.newyorker.com/magazine/2022/07/04/the-weird-analog-delights-of-foley-sound-effects`. 1, 116

[10] Philip Rodrigues Singer. The art of foley [accessed: 17 april 2024]. URL http://www.marblehead.net/foley/whatisitman.html. 1

[11] Ric Viers. *The Sound Effects Bible: How to create and record Hollywood style sound effects*. Michael Wiese, 2008. 1

[12] Mark Cartwright. *Supporting novice communication of audio concepts for audio production tools*. Northwestern University, 2016. 2

[13] Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. GANSynth: Adversarial neural audio synthesis. In *International Conference on Learning Representations*, New Orleans, Louisiana, United States, 2019. ICLR. URL https://openreview.net/forum?id=H1xQVn09FX. 2, 4, 19, 24, 26, 28, 37, 59, 80, 85, 94, 135

[14] Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. Musiclm: Generating music from text, 2023. 2, 106

[15] Ryan Louie, Andy Coenen, Cheng Zhi Huang, Michael Terry, and Carrie 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*, CHI '20, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450367080. doi: 10.1145/3313831.3376739. URL https://doi.org/10.1145/3313831.3376739. 2, 4, 19, 59, 106, 124

[16] Ryan Louie, Jesse Engel, and Cheng-Zhi Anna Huang. Expressive communication: Evaluating developments in generative models and steering interfaces for music creation. In *27th International Conference on Intelligent User Interfaces*, IUI '22, page 405–417, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391443. doi: 10.1145/3490099.3511159. URL https://doi.org/10.1145/3490099.3511159. 2, 4, 19, 20, 59, 106, 124

[17] Ben Shneiderman. Creativity support tools: Accelerating discovery and innovation. *Commun. ACM*, 50(12):20–32, dec 2007. ISSN 0001-0782. doi: 10.1145/1323688.1323689. URL https://doi.org/10.1145/1323688.1323689. 2, 5, 6, 18, 19, 21, 58, 106

[18] Karn N Watcharasupat. Controllable music: supervised learning of disentangled representations for music generation. *Master Thesis*, November 2021. Available at https://dr.ntu.edu.sg/handle/10356/153200. 2, 45

[19] Open AI. Introducing chatgpt, 2023. `https://openai.com/blog/chatgpt` [Accessed: 29 August 2023]. 3, 121

[20] Open AI. Dall.e 2, 2023. `https://openai.com/dall-e-2` [Accessed: 29 August 2023]. 3, 121

[21] John Joon Young Chung and Eytan Adar. Promptpaint: Steering text-to-image generation through paint medium-like interactions. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–17, 2023. 3

[22] Aaron Hertzmann. Can computers create art? In *Arts*, volume 7, page 18. MDPI, 2018. 5

[23] Baptiste Caramiaux and Sarah Fdili Alaoui. "explorers of unknown planets": Practices and politics of artificial intelligence in visual arts. *Proc. ACM Hum.-Comput. Interact.*, 6(CSCW2), nov 2022. doi: 10.1145/3555578. URL `https://doi.org/10.1145/3555578`. 21, 124

[24] Hyung-Kwon Ko, Gwanmo Park, Hyeon Jeon, Jaemin Jo, Juho Kim, and Jinwook Seo. Large-scale text-to-image generation models for visual artists' creative works. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23, page 919–933, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701061. doi: 10.1145/3581641.3584078. URL `https://doi.org/10.1145/3581641.3584078`. 18, 20, 124

[25] Nicholas Davis, Chih-PIn Hsiao, Kunwar Yashraj Singh, Lisa Li, and Brian Magerko. Empirically studying participatory sense-making in abstract drawing with a co-creative cognitive agent. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, IUI '16, page 196–207, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341370. doi: 10.1145/2856767.2856795. URL `https://doi.org/10.1145/2856767.2856795`. 19

[26] Changhoon Oh, Jungwoo Song, Jinhan Choi, Seonghyeon Kim, Sungwoo Lee, and Bongwon Suh. I lead, you help but only with enough details: Understanding user experience of co-creation with artificial intelligence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356206. doi: 10.1145/3173574.3174223. URL `https://doi.org/10.1145/3173574.3174223`. 19

[27] Alex Calderwood, Vivian Qiu, Katy Ilonka Gero, and Lydia B Chilton. How novelists use generative language models: An exploratory user study. In *HAI-GEN+ user2agent@ IUI*, 2020. 19, 124

[28] Elizabeth Clark, Anne Spencer Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith. Creative writing with a machine in the loop: Case studies on slogans and stories. In *23rd International Conference on Intelligent User Interfaces*, IUI '18, page 329–340, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450349451. doi: 10.1145/3172944.3172983. URL `https://doi.org/10.1145/3172944.3172983`. 3, 19

[29] Mark Cartwright, Ayanna Seals, Justin Salamon, Alex Williams, Stefanie Mikloska, Duncan MacConnell, Edith Law, Juan P. Bello, and Oded Nov. Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations. *Proc. ACM Hum.-Comput. Interact.*, 1(CSCW), dec 2017. doi: 10.1145/3134664. URL `https://doi.org/10.1145/3134664`. 3, 20, 80, 83, 90, 93, 129, 136

[30] Sebastian Säger, Benjamin Elizalde, Damian Borth, Christian Schulze, Bhiksha Raj, and Ian Lane. Audiopairbank: towards a large-scale tag-pair-based audio content analysis. *EURASIP Journal on Audio, Speech, and Music Processing*, 2018:1–12, 2018. 3, 64, 77

[31] Kelly Fitz. Sound Modelling and Morphing, 2007. `https://www.cerlsoundgroup.org/Kelly/soundmorphing.html` [Accessed: 14 March 2024]. 3, 4, 15, 57, 59

[32] Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. In *International Conference on Learning Representations*, New Orleans, Louisiana, United States, 2019. ICLR. URL `https://openreview.net/forum?id=ByMVTsR5KQ`. 4, 80

[33] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. In *Proc. 9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, page 125, 2016. 25

[34] Philippe Esling, Axel Chemla-Romeu-Santos, and Adrien Bitton. Generative timbre spaces: regularizing variational auto-encoders with perceptual metrics. In *Proc. Int. Conf. Digital Audio Effects (DAFx-18), Aveiro, Portugal*, pages 369–376, 2018. 40

[35] Philippe Esling, Axel Chemla-Romeu-Santos, and Adrien Bitton. Bridging audio analysis, perception and synthesis with perceptually-regularized variational timbre spaces. In *Proceedings of the 19th International Society for Music Information Retrieval Conference, ISMIR 2018, Paris, France, September 23-27, 2018*, pages 175–181, 2018.

[36] Josh H McDermott and Eero P Simoncelli. Sound texture perception via statistics of the auditory periphery: evidence from sound synthesis. *Neuron*, 71(5):926–940, 2011. 4, 22, 80, 94

[37] Thilo Thiede, William C Treurniet, Roland Bitto, Christian Schmidmer, Thomas Sporer, John G Beerends, and Catherine Colomes. Peaq-the itu standard for objective measurement of perceived audio quality. *Journal of the Audio Engineering Society*, 48(1/2):3–29, 2000. 4, 80

[38] Rainer Huber and Birger Kollmeier. Pemo-q—a new method for objective audio quality assessment using a model of auditory perception. *IEEE Transactions on audio, speech, and language processing*, 14(6):1902–1911, 2006.

[39] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet Audio Distance: A Reference-Free Metric for Evaluating Music Enhancement Algorithms. In *Proc. Interspeech 2019*, pages 2350–2354, Graz, 2019. Interspeech. doi: 10.21437/Interspeech.2019-2219. URL `http://dx.doi.org/10.21437/Interspeech.2019-2219`. 46, 66, 147, 150

[40] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29, Barcelona, Spain, 2016. Curran Associates, Inc. URL `https://proceedings.neurips.cc/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf`. 66

[41] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=r1lUOzWCW`.

[42] Muhammad Ferjad Naeem, Seong Joon Oh, Youngjung Uh, Yunjey Choi, and Jaejun Yoo. Reliable fidelity and diversity metrics for generative models. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 7176–7185, Virtual, 13–18 Jul 2020. PMLR. URL `https://proceedings.mlr.press/v119/naeem20a.html`.

[43] Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, Montreal, Canada, 2018. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2018/file/f7696a9b362ac5a51c3dc8f098b73923-Paper.pdf.

[44] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, Long Beach, USA, 2017. Curran Associates, Inc. URL https://proceedings.neurips.cc/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf. 4, 80

[45] Brecht De Man, Ryan Stables, and Joshua D Reiss. *Intelligent Music Production*. Routledge, 2019. 4, 106

[46] Yijun Zhou, Yuki Koyama, Masataka Goto, and Takeo Igarashi. Interactive exploration-exploitation balancing for generative melody composition. In *26th International Conference on Intelligent User Interfaces*, IUI '21, page 43–47, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380171. doi: 10.1145/3397481.3450663. URL https://doi.org/10.1145/3397481.3450663. 4, 106

[47] Emma Frid, Celso Gomes, and Zeyu Jin. Music creation by example. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–13, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450367080. doi: 10.1145/3313831.3376514. URL https://doi.org/10.1145/3313831.3376514.

[48] Renaud Bougueng Tchemeube, Jeffrey John Ens, and Philippe Pasquier. Calliope: A co-creative interface for multi-track music generation. In *Proceedings of the 14th Conference on Creativity and Cognition*, C&C '22, page 608–611, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393270. doi: 10.1145/3527927.3535200. URL https://doi.org/10.1145/3527927.3535200. 4, 19, 20, 59, 106

[49] Ben Shneiderman. *Human-centered AI*. Oxford University Press, 2022. 5, 7, 17, 126

[50] Jonas Frich, Lindsay MacDonald Vermeulen, Christian Remy, Michael Mose Biskjaer, and Peter Dalsgaard. Mapping the landscape of creativity support tools in hci. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–18, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300619. URL `https://doi.org/10.1145/3290605.3300619`. 5, 19, 106

[51] Linda Candy. Practice based research: A guide. *CCS report*, 1(2):1–19, 2006. 5, 113

[52] Ben Shneiderman. The eight golden rules of interface design[accessed: 30 june 2024], 2016. URL `https://www.cs.umd.edu/~ben/goldenrules.html`. 5, 6, 17, 19

[53] Jonas Löwgren. Articulating the use qualities of digital designs. *Aesthetic Computing*, 2006. 6, 17, 21, 106, 109, 110, 137

[54] Jacob O. Wobbrock and Julie A. Kientz. Research contributions in human-computer interaction. *Interactions*, 23(3):38–44, apr 2016. ISSN 1072-5520. doi: 10.1145/2907069. URL `https://doi.org/10.1145/2907069`. 9

[55] Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. Quantifying the carbon emissions of machine learning. *arXiv preprint arXiv:1910.09700*, 2019. 13

[56] Graham Wallas. *The art of thought*, volume 10. Harcourt, Brace, 1926. 16

[57] Sara Lenzi. The design of data sonification. design processes, protocols and tools grounded in anomaly detection. *Ph.D. Thesis*, 2021. 16, 141

[58] Daniel Hug and Nicolas Misdariis. Towards a conceptual framework to integrate designerly and scientific sound design methods. In *Proceedings of the 6th Audio Mostly Conference: A Conference on Interaction with Sound*, AM '11, page 23–30, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450310819. doi: 10.1145/2095667.2095671. URL `https://doi.org/10.1145/2095667.2095671`. 16

[59] Margaret A Boden. *The creative mind: Myths and mechanisms*. Routledge, 2004. 17, 18

[60] Youngseung Jeon, Seungwan Jin, Patrick C. Shih, and Kyungsik Han. Fashionq: An ai-driven creativity support tool for facilitating ideation in fashion design. In

Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi: 10.1145/3411764.3445093. URL https://doi.org/10.1145/3411764.3445093. 18

[61] Savvas Petridis, Hijung Valentina Shin, and Lydia B Chilton. Symbolfinder: Brainstorming diverse symbols using local semantic networks. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 385–399, 2021. 18

[62] Peter Knees and Kristina Andersen. Searching for audio by sketching mental images of sound: A brave new idea for audio retrieval in creative music production. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, ICMR '16, page 95–102, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450343596. doi: 10.1145/2911996.2912021. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/2911996.2912021. 18, 134

[63] Hugo Scurto, Bavo Van Kerrebroeck, Baptiste Caramiaux, and Frédéric Bevilacqua. Designing deep reinforcement learning for human parameter exploration. *ACM Trans. Comput.-Hum. Interact.*, 28(1), jan 2021. ISSN 1073-0516. doi: 10.1145/3414472. URL https://doi.org/10.1145/3414472. 18

[64] John J Dudley and Per Ola Kristensson. A review of user interface design for interactive machine learning. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 8(2):1–37, 2018. 18, 134

[65] Casey Reas and Ben Fry. *Processing: a programming handbook for visual designers and artists*, volume 6812. Mit Press, 2007. 18, 134

[66] Sebastian Deterding, Jonathan Hook, Rebecca Fiebrink, Marco Gillies, Jeremy Gow, Memo Akten, Gillian Smith, Antonios Liapis, and Kate Compton. Mixed-initiative creative interfaces. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, page 628–635, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450346566. doi: 10.1145/3027063.3027072. URL https://doi.org/10.1145/3027063.3027072. 19, 124

[67] Angie Spoto, Natalia Oleynik, Sebastian Deterding, and Jon Hook. Library of mixed-initiative creative interfaces, 2017. 19

[68] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020. 19

[69] Yuwen Lu, Chengzhi Zhang, Iris Zhang, and Toby Jia-Jun Li. Bridging the gap between ux practitioners' work practices and ai-enabled design support tools. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI EA '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391566. doi: 10.1145/3491101.3519809. URL `https://doi.org/10.1145/3491101.3519809`. 19

[70] Janin Koch, Nicolas Taffin, Andrés Lucero, and Wendy E. Mackay. Semanticcollage: Enriching digital mood board design with semantic labels. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, DIS '20, page 407–418, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369749. doi: 10.1145/3357236.3395494. URL `https://doi.org/10.1145/3357236.3395494`.

[71] Mohammad Amin Mozaffari, Xinyuan Zhang, Jinghui Cheng, and Jin L.C. Guo. Ganspiration: Balancing targeted and serendipitous inspiration in user interface design with style-based generative adversarial network. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391573. doi: 10.1145/3491102.3517511. URL `https://doi.org/10.1145/3491102.3517511`.

[72] Frederic Gmeiner, Humphrey Yang, Lining Yao, Kenneth Holstein, and Nikolas Martelaro. Exploring challenges and opportunities to support designers in learning to co-create with ai-based manufacturing design tools. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394215. doi: 10.1145/3544548.3580999. URL `https://doi.org/10.1145/3544548.3580999`. 19, 20

[73] Akito Van Troyer and Rebecca Kleinberger. From mondrian to modular synth: Rendering NIME using generative adversarial networks. In Marcelo Queiroz and Anna Xambó Sedó, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 272–277, Porto Alegre, Brazil, June 2019. UFRGS. doi: 10.5281/zenodo.3672956. URL `http://www.nime.org/proceedings/2019/nime2019_paper052.pdf`. 19

[74] Corey Ford and Nick Bryan-Kinns. Speculating on reflection and people's music co-creation with ai. *Workshop on Generative AI and HCI at the CHI Conference on Human Factors in Computing Systems 2022*, 2022. 19, 26, 106, 107, 109, 110

[75] Nick Bryan-Kinns, Corey Ford, Alan Chamberlain, Steven David Benford, Helen Kennedy, Zijin Li, Wu Qiong, Gus G. Xia, and Jeba Rezwana. Explainable ai for the arts: Xaixarts. In *Proceedings of the 15th Conference on Creativity and Cognition*, C&C '23, page 1–7, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701801. doi: 10.1145/3591196.3593517. URL `https://doi.org/10.1145/3591196.3593517`. 19, 26, 107, 126, 136

[76] Rebecca Fiebrink and Baptiste Caramiaux. The machine learning algorithm as creative musical tool. *arXiv preprint arXiv:1611.00379*, 2016. 19

[77] Rebecca Fiebrink and Perry R Cook. The wekinator: a system for real-time, interactive machine learning in music. In *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)(Utrecht)*, volume 3, pages 2–1. Citeseer, 2010. 19

[78] Koray Tahiroğlu, Miranda Kastemaa, and Oskar Koli. Al-terity: Non-rigid musical instrument with artificial intelligence applied to real-time audio synthesis. In Romain Michon and Franziska Schroeder, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 337–342, Birmingham, UK, July 2020. Birmingham City University. doi: 10.5281/zenodo.4813402. URL `https://www.nime.org/proceedings/2020/nime2020_paper65.pdf`. 19, 54, 59

[79] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1068–1077, Sydney, NSW, Australia, 2017. JMLR.org. 19, 37, 59, 89, 140

[80] Jesse Engel, Lamtharn (Hanoi) Hantrakul, Chenjie Gu, and Adam Roberts. Ddsp: Differentiable digital signal processing. In *International Conference on Learning Representations*, 2020. 19, 37, 59

[81] Fabio Morreale and Antonella De Angeli. Collaborating with an autonomous agent to generate affective music. *Computers in Entertainment (CIE)*, 14(3):1–21, 2016. 19, 59

[82] Purnima Kamath, Fabio Morreale, Priambudi Lintang Bagaskara, Yize Wei, and Suranga Nanayakkara. Sound designer-generative ai interactions: Towards designing creative support tools for professional sound designers. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA*, CHI '24, New York, NY, USA, 2024.

Association for Computing Machinery. ISBN 979-8-4007-0330-0/24/05. doi: 10.1145/3613904.3642040. 19, 59, 66

[83] Meng-Han Wu and Alexander Quinn. Confusing the crowd: Task instruction quality on amazon mechanical turk. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 5(1):206–215, Sep. 2017. URL `https://ojs.aaai.org/index.php/HCOMP/article/view/13317`. 20, 80, 81, 83, 100

[84] ITU. Recommendation ITU-R BS.1534-2: Method for the subjective assessment of intermediate quality level of audio systems. In *ITU BS Series*, pages 1–36, Int., 2014. Radiocommunication sector of ITU. 20, 81, 84

[85] Ujwal Gadiraju, Jie Yang, and Alessandro Bozzon. Clarity is a worthwhile quality: On the role of task clarity in microtask crowdsourcing. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, HT '17, page 5–14, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450347082. doi: 10.1145/3078714.3078715. URL `https://doi-org.libproxy1.nus.edu.sg/10.1145/3078714.3078715`. 20, 81, 83, 100

[86] V. K. Chaithanya Manam, Dwarakanath Jampani, Mariam Zaim, Meng-Han Wu, and Alexander J. Quinn. Taskmate: A mechanism to improve the quality of instructions in crowdsourcing. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 1121–1130, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366755. doi: 10.1145/3308560.3317081. URL `https://doi-org.libproxy1.nus.edu.sg/10.1145/3308560.3317081`. 83

[87] Gaoping Huang, Meng-Han Wu, and Alexander J. Quinn. Task design for crowdsourcing complex cognitive skills. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380959. doi: 10.1145/3411763.3443447. URL `https://doi-org.libproxy1.nus.edu.sg/10.1145/3411763.3443447`. 20, 83

[88] Cheng-Zhi Anna Huang, Hendrik Vincent Koops, Ed Newton-Rex, Monica Dinculescu, and Carrie J Cai. Ai song contest: Human-ai co-creation in songwriting. *arXiv preprint arXiv:2010.05388*, 2020. 20

[89] Jonas Lowgren and Erik Stolterman. *Thoughtful interaction design: A design perspective on information technology*. Mit Press, 2007. 21, 106, 109, 110, 137

[90] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N. Bennett, Kori Inkpen, Jaime

Teevan, Ruth Kikin-Gil, and Eric Horvitz. Guidelines for human-ai interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–13, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300233. URL https://doi.org/10.1145/3290605.3300233. 21, 137

[91] William W. Gaver, Jacob Beaver, and Steve Benford. Ambiguity as a resource for design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, page 233–240, New York, NY, USA, 2003. Association for Computing Machinery. ISBN 1581136307. doi: 10.1145/642611.642653. URL https://doi.org/10.1145/642611.642653. 21, 106, 109, 110, 137

[92] Nicolas Saint-Arnaud and Kris Popat. *Analysis and Synthesis of Sound Textures*, page 293–308. L. Erlbaum Associates Inc., USA, 1998. ISBN 0805822836. doi: 10.5555/285582.285601. 22, 80

[93] Julius O. Smith. *Spectral Audio Signal Processing*. http://ccrma.stanford.edu/ jos/sasp/, accessed August 19, 2023. online book, 2011 edition. 23, 39, 134

[94] Xavier Serra and Julius Smith. Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14(4):12–24, 1990.

[95] Xavier Serra. *A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition*. Stanford Uni, 1990.

[96] Xavier Serra et al. Musical sound modeling with sinusoids plus noise. *Musical signal processing*, 2:91–122, 1997. 23, 39, 134

[97] Julius O. Smith. *Physical Audio Signal Processing*. http://ccrma.stanford.edu/ jos/pasp/, accessed August 19, 2023. online book, 2010 edition. 23, 39, 134

[98] William W Gaver. What in the world do we hear?: An ecological approach to auditory event perception. *Ecological psychology*, 5(1):1–29, 1993. 37, 39

[99] William W Gaver. How do we hear in the world? explorations in ecological acoustics. *Ecological psychology*, 5(4):285–313, 1993. 23, 37, 39, 134

[100] Ethan Manilow, Prem Seetharman, and Justin Salamon. *Open Source Tools & Data for Music Source Separation*. https://source-separation.github.io/tutorial, October 2020. URL https://source-separation.github.io/tutorial. 23, 24

[101] Denis Smalley. Spectromorphology: explaining sound-shapes. *Organised sound*, 2 (2):107–126, 1997. 23, 109, 126, 136

[102] Lonce Wyse. Audio spectrogram representations for processing with convolutional neural networks. *arXiv preprint arXiv:1706.09559*, 2017. 24

[103] Lonce Wyse. Real-valued parametric conditioning of an RNN for interactive sound synthesis. *arXiv preprint arXiv:1805.10808*, abs/1805.10808, 2018. URL `http://arxiv.org/abs/1805.10808`. 24, 25

[104] Muhammad Huzaifah and Lonce Wyse. *Deep Generative Models for Musical Audio Synthesis*, pages 639–678. Springer International Publishing, Cham, 2021. ISBN 978-3-030-72116-9. doi: 10.1007/978-3-030-72116-9_22. URL `https://doi.org/10.1007/978-3-030-72116-9_22`. 24, 25, 27

[105] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL `https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf`. 24, 25, 26

[106] John Pierce. Sound waves and sine waves. *Music, Cognition, and Computerized Sound: An Introduction to Psychoacoustics', MIT Press, Cambridge MA*, pages 37–56, 1999. 24

[107] JOhn Shepard. Pitch perception and measurement. *Music, Cognition, and Computerized Sound: An Introduction to Psychoacoustics', MIT Press, Cambridge MA*, pages 149–165, 1999. 24

[108] Hugo Caracalla and Axel Roebel. Sound texture synthesis using ri spectrograms. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 416–420. IEEE, 2020. 24

[109] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33:17022–17033, 2020. 24, 61

[110] Chitralekha Gupta, Purnima Kamath, and Lonce Wyse. Signal representations for synthesizing audio textures with generative adversarial networks. In Simone Spagnol Davide Andrea Mauro and Andrea Valle, editors, *Proceedings of the 18th Sound and Music Computing Conference*. Sound and Music Computing Network, Axea

sas/SMC Network, 2021. doi: 10.5281/zenodo.5113511. 24, 28, 46, 66, 67, 148, 155

[111] Zdeněk Průša, Peter Balazs, and Peter Lempel Søndergaard. A noniterative method for reconstruction of phase from stft magnitude. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(5):1154–1164, 2017. doi: 10.1109/TASLP.2017.2678166. 24, 46, 148, 155

[112] Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model. *arXiv preprint arXiv:1612.07837*, 2016. 25

[113] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 25, 32, 58, 61

[114] Prateek Verma and Chris Chafe. A generative model for raw audio using transformer architectures. In *2021 24th International Conference on Digital Audio Effects (DAFx)*, pages 230–237. IEEE, 2021. URL `https://www.dafx.de/paper-archive/2021/proceedings/papers/DAFx20in21_paper_40.pdf`. 25

[115] Chitralekha Gupta, Purnima Kamath, Yize Wei, Zhuoyao Li, Suranga Nanayakkara, and Lonce Wyse. Towards controllable audio texture morphing. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10096328. 25, 37, 59, 67, 76, 135

[116] Lonce Wyse, Purnima Kamath, and Chitralekha Gupta. Sound model factory: An integrated system architecture for generative audio modelling. In Tiago Martins, Nereida Rodríguez-Fernández, and Sérgio M. Rebelo, editors, *International Conference on Computational Intelligence in Music, Sound, Art and Design (Part of EvoStar). Artificial Intelligence in Music, Sound, Art and Design*, pages 308–322, Cham, 2022. Springer International Publishing. ISBN 978-3-031-03789-4. 25, 28, 37, 59, 87, 89, 135, 142

[117] Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. Diffsound: Discrete diffusion model for text-to-sound generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1720–1733, 2023. doi: 10.1109/TASLP.2023.3268730. 25, 31, 32

[118] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*, 2020. 25

[119] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020. 25

[120] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. `http://www.deeplearningbook.org`. 26

[121] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 2180–2188, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819. doi: 10.5555/3157096.3157340. 27

[122] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 27

[123] Javier Nistal, Stefan Lattner, and Gaël Richard. Drumgan: Synthesis of drum sounds with timbral feature conditioning using generative adversarial networks. In *International Society for Music Information Retrieval Conference*, 2020. 28, 37, 40

[124] Javier Nistal, Stefan Lattner, and Gaël Richard. Darkgan: Exploiting knowledge distillation for comprehensible audio synthesis with gans. In *International Society for Music Information Retrieval Conference*, 2021. 28, 37

[125] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 28, 29, 36, 41, 146

[126] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, Los Alamitos, CA, USA, jun 2020. IEEE Computer Society. doi: 10.1109/CVPR42600.2020.00813. URL `https://doi.ieeecomputersociety.org/10.1109/CVPR42600.2020.00813`. 28, 29, 36, 38, 40, 41, 140, 146, 155

[127] Keunwoo Choi, Jaekwon Im, Laurie Heller, Brian McFee, Keisuke Imoto, Yuki Okamoto, Mathieu Lagrange, and Shinosuke Takamichi. Foley sound synthesis at the dcase 2023 challenge. *arXiv preprint arXiv:2304.12521*, 2023. 29, 108, 146, 149

[128] Keunwoo Choi, Jaekwon Im, Laurie Heller, Brian McFee, Keisuke Imoto, Yuki Okamoto, Mathieu Lagrange, and Shinosuke Takamichi. Foley sound synthesis at the dcase 2023 challenge. *In arXiv e-prints: 2304.12521*, 2023. 29, 149

[129] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3121–3138, 2023. doi: 10.1109/TPAMI.2022.3181070. 30

[130] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf`. 31, 58, 60

[131] Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. Text-to-audio generation using instruction guided latent diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM '23, page 3590–3598, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701085. doi: 10.1145/3581783.3612348. URL `https://doi-org.libproxy1.nus.edu.sg/10.1145/3581783.3612348`. 31, 58, 60, 65, 135

[132] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. AudioLDM: Text-to-audio generation with latent diffusion models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 21450–21474. PMLR, 23–29 Jul 2023. URL `https://proceedings.mlr.press/v202/liu23f.html`. 32, 60, 65, 66, 106, 128, 135, 139, 140

[133] Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *arXiv preprint arXiv:2308.05734*, 2023. 58

[134] Zachary Novack, Julian McAuley, Taylor Berg-Kirkpatrick, and Nicholas J Bryan. Ditto: Diffusion inference-time t-optimization for music generation. *arXiv preprint arXiv:2401.12179*, 2024. 31, 60

[135] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 31, 32, 60

[136] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 31

[137] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 31, 60

[138] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL `https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf`. 36, 87, 146

[139] Peter Grosche, Meinard Müller, and Joan Serrà. Audio content-based music retrieval. In Meinard Müller, Masataka Goto, and Markus Schedl, editors, *Multimodal Music Processing*, volume 3 of *Dagstuhl Follow-Ups*, pages 157–174. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2012. 36

[140] Jonathan T Foote. Content-based retrieval of music and audio. In *Multimedia storage and archiving systems II*, volume 3229, pages 138–147. SPIE, 1997.

[141] Avery Wang. The shazam music recognition service. *Communications of the ACM*, 49(8):44–48, 2006.

[142] William P. Birmingham. MUSART: music retrieval via aural queries. In *ISMIR 2001, 2nd International Symposium on Music Information Retrieval, Indiana University, Bloomington, Indiana, USA, October 15-17, 2001, Proceedings*, 2001. 36

[143] Asif Ghias, Jonathan Logan, David Chamberlin, and Brian C Smith. Query by humming: Musical information retrieval in an audio database. In *Proceedings of the third ACM international conference on Multimedia*, pages 231–236, 1995. 36

[144] Mark Cartwright and Bryan Pardo. Synthassist: An audio synthesizer programmed with vocal imitation. In *Proceedings of the 22nd ACM International Conference on Multimedia*, MM '14, page 741–742, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450330633. doi: 10.1145/2647868.2654880.

[145] Bongjun Kim and Bryan Pardo. Improving content-based audio retrieval by vocal imitation feedback. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4100–4104. IEEE, 2019.

[146] Yichi Zhang, Bryan Pardo, and Zhiyao Duan. Siamese style convolutional neural networks for sound search by vocal imitation. *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 27(2):429–441, 2018. 36

[147] Chae Young Lee, Anoop Toffy, Gue Jun Jung, and Woo-Jin Han. Conditional wavegan. *arXiv preprint arXiv:1809.10636*, 2018. 37

[148] Music Technology Group. Audio commons audio extractor, 2018. URL `https://www.audiocommons.org/2018/07/15/audio-commons-audio-extractor.html`. 37

[149] Music Technology Group. Essentia: Open-source c++ library for audio analysis and audio-based music information retrieval, 2022. URL `https://essentia.upf.edu/`. 37

[150] Yunyi Liu, Craig Jin, and David Gunawan. Ddsp-sfx: Acoustically-guided sound effects generation with differentiable digital signal processing. *arXiv preprint arXiv:2309.08060*, 2023. 37

[151] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020. 37, 66

[152] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780, 2017. 37, 65, 66, 68, 138

[153] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020. 38, 54

[154] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1532–1540, Los Alamitos, CA, USA, jun 2021. IEEE Computer Society. doi: 10.1109/CVPR46437.2021.00158. URL `https://doi.ieeecomputersociety.org/10.1109/CVPR46437.2021.00158`. 38, 47, 50, 55, 107, 109, 126, 156

[155] Koray Tahiroglu, Miranda Kastemaa, and Oskar Koli. Ganspacesynth: A hybrid generative adversarial network architecture for organising the latent space using a dimensionality reduction for real-time audio synthesis. In *Proceedings of the 2nd Joint Conference on AI Music Creativity*, 2021. doi: 10.5281/zenodo.5137902. 38, 54

[156] Kazi Nazmul Haque, Rajib Rana, Jiajun Liu, John H. L. Hansen, Nicholas Cummins, Carlos Busso, and Björn W. Schuller. Guided generative adversarial neural network for representation learning and audio generation using fewer labelled audio data. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29: 2575–2590, 2021. doi: 10.1109/TASLP.2021.3098764. 38

[157] Kazi Nazmul Haque, Rajib Rana, and Björn W. Schuller. High-fidelity audio generation and representation learning with guided adversarial autoencoder. *IEEE Access*, 8:223509–223528, 2020. doi: 10.1109/ACCESS.2020.3040797. 38

[158] Rishubh Parihar, Ankit Dhiman, Tejan Karmali, and Venkatesh R. Everything is there in latent space: Attribute editing and attribute style manipulation by stylegan latent space exploration. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 1828–1836, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450392037. doi: 10.1145/3503161.3547972. 38

[159] Lucy Chai, Jonas Wulff, and Phillip Isola. Using latent space regression to analyze and leverage compositionality in gans. In *International Conference on Learning Representations*, Virtual Only, 2021. ICLR. 38, 41, 42

[160] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4080–4090, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. 38, 126, 135

[161] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 41, 46

[162] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 131–135, 2017. doi: 10.1109/ICASSP.2017.7952132. 41

[163] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *Advances in Neural Information Processing Systems*, 35:28708–28720, 2022. 41

[164] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Masked spectrogram modeling using masked autoencoders for learning general-purpose audio representation. In *HEAR: Holistic Evaluation of Audio Representations*, pages 1–24. PMLR, 2022. 41

[165] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 42, 50

[166] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2405–2413, Los Alamitos, CA, USA, jun 2016. IEEE Computer Society. doi: 10.1109/CVPR.2016.264. 45, 108

[167] Chitralekha Gupta, Yize Wei, Zequn Gong, Purnima Kamath, Zhuoyao Li, and Lonce Wyse. Parameter sensitivity of deep-feature based evaluation metrics for audio textures. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR*, pages 462–468, 2022. 45, 76, 94

[168] Andrés Marafioti, Nathanaël Perraudin, Nicki Holighaus, and Piotr Majdak. Adversarial generation of time-frequency features with application in audio synthesis. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4352–4362. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/marafioti19a.html`. 46, 148, 155

[169] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3 (29):861, 2018. doi: 10.21105/joss.00861. URL `https://doi.org/10.21105/joss.00861`. 46

[170] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 46

[171] Kamalesh Palanisamy, Dipika Singhania, and Angela Yao. Rethinking CNN models for audio classification. *CoRR*, abs/2007.11154, 2020. 46

[172] Mark Cartwright, Bryan Pardo, Gautham J. Mysore, and Matt Hoffman. Fast and easy crowdsourced perceptual audio evaluation. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 619–623, Shanghai, China, 2016. IEEE. doi: 10.1109/ICASSP.2016.7471749. 52, 84, 89

[173] Purnima Kamath, Zhuoyao Li, Chitralekha Gupta, Kokil Jaidka, Suranga Nanayakkara, and Lonce Wyse. Evaluating descriptive quality of ai-generated audio using image-schemas. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, IUI '23, page 621–632, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701061. doi: 10.1145/3581641.3584083. URL `https://doi.org/10.1145/3581641.3584083`. 53

[174] Jaewoong Choi, Junho Lee, Changyeon Yoon, Jung Ho Park, Geonho Hwang, and Myungjoo Kang. Do not escape from the manifold: Discovering the local coordinates on the latent space of GANs. In *International Conference on Learning Representations*, 2022. 55

[175] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=_CDixzkzeyb`. 58, 60, 63

[176] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023.

[177] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1921–1930, 2023. 58, 60

[178] Marcelo Caetano. *Morphing isolated quasi-harmonic acoustic musical instrument sounds guided by perceptually motivated features*. PhD thesis, Paris 6, 2011. 59

[179] Marcelo Caetano. Morphing musical instrument sounds with the sinusoidal model in the sound morphing toolbox. In *International Symposium on Computer Music Multidisciplinary Research*, pages 481–503. Springer, 2019. 67

[180] Kelly Fitz, Lippold Haken, Susanne Lefvert, Corbin Champion, and Mike O'Donnell. Cell-utes and flutter-tongued cats: Sound morphing using loris and the reassigned bandwidth-enhanced model. *Computer Music Journal*, 27(3):44–65, 2003. ISSN 01489267, 15315169. URL `http://www.jstor.org/stable/3681801`.

[181] Malcolm Slaney, Michele Covell, and Bud Lassiter. Automatic audio morphing. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 2, pages 1001–1004. IEEE, 1996. 59

[182] Savvas Kazazis, Philippe Depalle, and Stephen McAdams. Sound morphing by audio descriptors and parameter interpolation. In *Proceedings of the 19th International Conference on Digital Audio Effects (DAFx-16). Brno, Czech Republic*, 2016. 59

[183] William A Sethares and James A Bucklew. Kernel techniques for generalized audio crossfades. *Cogent Mathematics*, 2(1):1102116, 2015. 59

[184] Purnima Kamath, Chitralekha Gupta, Lonce Wyse, and Suranga Nanayakkara. Example-based framework for perceptually guided audio texture generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2555–2565, 2024. doi: 10.1109/TASLP.2024.3393741. 59, 107, 109, 126, 154, 155

[185] Curtis Hawthorne, Ian Simon, Adam Roberts, Neil Zeghidour, Josh Gardner, Ethan Manilow, and Jesse Engel. Multi-instrument music synthesis with spectrogram diffusion. *arXiv preprint arXiv:2206.05408*, 2022. 60

[186] Seth* Forsgren and Hayk* Martiros. Riffusion - Stable diffusion for real-time music generation. 2022. URL `https://riffusion.com/about`.

[187] Ke Chen, Yusong Wu, Haohe Liu, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. *arXiv preprint arXiv:2308.01546*, 2023. 60

[188] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 60

[189] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 60

[190] Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019. 60

[191] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 65

[192] Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. *arXiv preprint arXiv:2402.04825*, 2024. 65

[193] The PyTorch Foundation. Pytorch 2.0, 2023. `https://pytorch.org/get-started/pytorch-2.0` [Accessed: 30 March 2024]. 65

[194] Snowflake Inc. Streamlit.io, 2024. `https://streamlit.io/` [Accessed: 30 March 2024]. 65

[195] Marcelo Caetano and Naotoshi Osaka. A formal evaluation framework for sound morphing. In *ICMC*, 2012. 65, 72, 76

[196] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE, 2017. 66

[197] Javier Nistal, Stefan Lattner, and Gaël Richard. Comparing representations for audio synthesis using generative adversarial networks. In *2020 28th European Signal Processing Conference (EUSIPCO)*, pages 161–165. IEEE, 2021. 66

[198] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 66

[199] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, 2019. ISSN 1077-3142. doi: https://doi.org/10.1016/j.

cviu.2018.10.009. URL https://www.sciencedirect.com/science/article/pii/S1077314218304272. 80

[200] Shane Barratt and Rishi Sharma. A note on the inception score. *ArXiv*, abs/1801.01973, 2018. 80

[201] Richard McWalter and Josh H McDermott. Adaptive and selective time averaging of auditory scenes. *Current Biology*, 28(9):1405–1418, 2018. 80, 90

[202] Mark Johnson. *The philosophical significance of image schemas*, pages 15–34. 12 2005. ISBN 978-3-11-018311-5. doi: 10.1515/9783110197532.1.15. 81, 82, 100, 137, 140

[203] George Lakoff and Mark Johnson. *Metaphors we live by*. University of Chicago press, 2008. 82, 136

[204] Katie Wilkie, Simon Holland, and Paul Mulholland. What can the language of musicians tell us about music interaction design? *Computer Music Journal*, 34(4): 34–48, 2010. 82, 85, 140

[205] Zahra Nouri, Ujwal Gadiraju, Gregor Engels, and Henning Wachsmuth. What is unclear? computational assessment of task clarity in crowdsourcing. In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media*, HT '21, page 165–175, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385510. doi: 10.1145/3465336.3475109. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/3465336.3475109. 83

[206] Mark Cartwright, Graham Dove, Ana Elisa Méndez Méndez, Juan P. Bello, and Oded Nov. Crowdsourcing multi-label audio annotation tasks with citizen scientists. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–11, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300522. URL https://doi.org/10.1145/3290605.3300522. 83, 102

[207] Ana Elisa Méndez Méndez, Mark Cartwright, and Juan Pablo Bello. Machine-crowd-expert model for increasing user engagement and annotation quality. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI EA '19, page 1–6, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359719. doi: 10.1145/3290607.3313054. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/3290607.3313054. 83

[208] Michael Schoeffler, Sarah Bartoschek, Fabian-Robert Stöter, Marlene Roess, Susanne Westphal, Bernd Edler, and Jürgen Herre. webmushra—a comprehensive framework for web-based listening tests. *Journal of Open Research Software*, 6(1), 02 2018. doi: 10.5334/jors.187. 84

[209] Mark Cartwright, Bryan Pardo, and Gautham J. Mysore. Crowdsourced pairwise-comparison for source separation evaluation. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 606–610, Calgary, AB, Canada, 2018. IEEE. doi: 10.1109/ICASSP.2018.8462153. 84

[210] Jin Ha Lee. Crowdsourcing music similarity judgments using mechanical turk. In *International Society for Music Information Retrieval*, pages 183–188, Utrecht, Netherlands, 2010. ISMIR. 84

[211] Jieun Oh and Ge Wang. Evaluating crowdsourcing through amazon mechanical turk as a technique for conducting music perception experiments. In *Proceedings of the 12th International Conference on Music Perception and Cognition*, pages 1–6, Greece, 2012. School of Music Studies, Aristotle University of Thessaloniki.

[212] Julián Urbano, Jorge Morato, Mónica Marrero, and Diego Martín. Crowdsourcing preference judgments for evaluation of music similarity tasks. *ACM SIGIR Workshop on Crowdsourcing for Search Evaluation*, pages 9–16, 01 2010.

[213] Ioannis Petros Samiotis, Sihang Qiu, Christoph Lofi, Jie Yang, Ujwal Gadiraju, and Alessandro Bozzon. Exploring the music perception skills of crowd workers. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 9, pages 108–119, Virtual, 2021. Association for the Advancement of Artificial Intelligence (AAAI). 84

[214] Michael I Mandel, Douglas Eck, and Yoshua Bengio. Learning tags that vary within a song. In *International Society for Music Information Retrieval*, pages 399–404, Utrecht, Netherlands, 2010. ISMIR. 84

[215] Jacquelin A Speck, Erik M Schmidt, Brandon G Morton, and Youngmoo E Kim. A comparative study of collaborative vs. traditional musical mood annotation. In *ISMIR*, volume 104, pages 549–554, Miami, USA, 2011. Citeseer, ISMIR. 84

[216] Lonce Wyse, Norikazu Mitani, and Suranga Nanayakkara. The Effect of Visualizing Audio Targets in a Musical Listening and Performance Task. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 304–307, Oslo, Norway, June 2011. Zenodo. doi: 10.5281/zenodo.1178191. URL `https://doi.org/10.5281/zenodo.1178191`. 84

[217] Nicholas Jillings, Brecht De Man, David Moffat, and Joshua Reiss. Web audio evaluation tool: A framework for subjective assessment of audio. In *Web Audio Conference*, Atlanta, USA, 04 2016. Georgia Tech. 84

[218] Joshua R De Leeuw. jspsych: A javascript library for creating behavioral experiments in a web browser. *Behavior research methods*, 47(1):1–12, 2015. 84

[219] Peter W Donhauser and Denise Klein. Audio-tokens: a toolbox for rating, sorting and comparing audio samples in the browser. *Behavior research methods*, 2022. 84

[220] Mark Johnson. *The body in the mind: The bodily basis of meaning, imagination, and reason*. University of Chicago press, 2013. 85

[221] Candace Brower. A cognitive theory of musical meaning. *Journal of music theory*, 44(2):323–379, 2000. 85

[222] Yin-Jyun Luo, Kat Agres, and Dorien Herremans. Learning disentangled representations of timbre and pitch for musical instrument sounds using gaussian mixture variational autoencoders. *International Society of Music Information Retrieval (ISMIR)*, 2019. 85

[223] Brian O'Reilly. Brian o'reilly's electroacoustic compositions and noise music, Aug 2008. URL `https://vimeo.com/dendriform`. 89, 140

[224] Joseph M Antognini, Matt Hoffman, and Ron J Weiss. Audio texture synthesis with random neural networks: Improving diversity and quality. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3587–3591. IEEE, 2019. 94

[225] Lonce Wyse and Prashanth Thattai Ravikumar. Syntex: parametric audio texture datasets for conditional training of instrumental interfaces. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, The University of Auckland, New Zealand, jun 2022. doi: 10.21428/92fbeb44.0fe70450. URL `https://doi.org/10.21428/92fbeb44.0fe70450`. 96, 138, 140

[226] William W. Gaver. What in the world do we hear?: An ecological approach to auditory event perception. *Ecological Psychology*, 5(1):1–29, 1993. doi: 10.1207/s15326969eco0501_1. URL `https://doi.org/10.1207/s15326969eco0501_1`. 101

[227] Ruta Desai, Fraser Anderson, Justin Matejka, Stelian Coros, James McCann, George Fitzmaurice, and Tovi Grossman. Geppetto: Enabling semantic design of expressive robot behaviors. In *Proceedings of the 2019 CHI Conference on*

Human Factors in Computing Systems, CHI '19, page 1–14, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300599. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/3290605.3300599. 102

[228] John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. Talebrush: Sketching stories with generative pretrained language models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391573. doi: 10.1145/3491102.3501819. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/3491102.3501819. 102

[229] Robin Brewer, Meredith Ringel Morris, and Anne Marie Piper. "why would anybody do this?": Understanding older adults' motivations and challenges in crowd work. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 2246–2257, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858198. URL https://doi.org/10.1145/2858036.2858198. 102

[230] David Martin, Benjamin V. Hanrahan, Jacki O'Neill, and Neha Gupta. Being a turker. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '14, page 224–235, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450325400. doi: 10.1145/2531602.2531663. URL https://doi-org.libproxy1.nus.edu.sg/10.1145/2531602.2531663.

[231] Panagiotis G Ipeirotis. Demographics of mechanical turk. *CeDER Working Papers*, 10(1), 2010. URL http://hdl.handle.net/2451/29585. 102

[232] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=jtiQ26sCJi. 106

[233] Robert Jack, Jacob Harrison, and Andrew McPherson. Digital musical instruments as research products. In Romain Michon and Franziska Schroeder, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 446–451, Birmingham, UK, July 2020. Birmingham City University. doi: 10.5281/zenodo.4813465. URL https://www.nime.org/proceedings/2020/nime2020_paper86.pdf. 106

[234] Virginia Braun and Victoria Clarke. Reflecting on reflexive thematic analysis. *Qualitative Research in Sport, Exercise and Health*, 11(4):589–597, 2019. doi: 10.1080/2159676X.2019.1628806. URL https://doi.org/10.1080/2159676X.2019.1628806. 106, 114

[235] Justin D. Weisz, Michael J. Muller, Jessica He, and Stephanie Houde. Toward general design principles for generative ai applications 130-144. In *IUI Workshops*, 2023. URL https://api.semanticscholar.org/CorpusID:255825625. 107, 109, 124

[236] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf. 108

[237] Linda Candy, Shigeki Amitani, and Zafer Bilda. Practice-led strategies for interactive art research. *CoDesign*, 2(4):209–223, 2006. doi: 10.1080/15710880601007994. URL https://doi.org/10.1080/15710880601007994. 113

[238] Virginia Braun and Victoria Clarke. One size fits all? what counts as quality practice in (reflexive) thematic analysis? *Qualitative Research in Psychology*, 18 (3):328–352, 2021. doi: 10.1080/14780887.2020.1769238. URL https://doi.org/10.1080/14780887.2020.1769238. 114

[239] Virginia Braun and Victoria Clarke. To saturate or not to saturate? questioning data saturation as a useful concept for thematic analysis and sample-size rationales. *Qualitative Research in Sport, Exercise and Health*, 13(2):201–216, 2021. doi: 10.1080/2159676X.2019.1704846. URL https://doi.org/10.1080/2159676X.2019.1704846. 114

[240] Leigh Landy. *Understanding the art of sound organization*. Mit Press, 2007. 118

[241] Garima Sharma, Karthikeyan Umapathy, and Sridhar Krishnan. Trends in audio texture analysis, synthesis, and applications. *Journal of the Audio Engineering Society*, 70(3):108–127, 2022. doi: 10.17743/jaes.2021.0060. URL http://www.aes.org/e-lib/browse.cfm?elib=21554. 123

[242] Michael Muller, Lydia B Chilton, Anna Kantosalo, Charles Patrick Martin, and Greg Walsh. Genaichi: Generative ai and hci. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI EA '22, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450391566. doi: 10.1145/3491101.3503719. URL `https://doi.org/10.1145/3491101.3503719`. 124

[243] Michael Muller, Lydia B Chilton, Anna Kantosalo, Q. Vera Liao, Mary Lou Maher, Charles Patrick Martin, and Greg Walsh. Genaichi 2023: Generative ai and hci at chi 2023. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI EA '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450394222. doi: 10.1145/3544549.3573794. URL `https://doi.org/10.1145/3544549.3573794`. 124

[244] Li-Yuan Chiou, Peng-Kai Hung, Rung-Huei Liang, and Chun-Teng Wang. Designing with ai: An exploration of co-ideation with image generators. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, DIS '23, page 1941–1954, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450398930. doi: 10.1145/3563657.3596001. URL `https://doi.org/10.1145/3563657.3596001`. 124

[245] John Joon Young Chung, Shiqing He, and Eytan Adar. The intersection of users, roles, interactions, and technologies in creativity support tools. In *Proceedings of the 2021 ACM Designing Interactive Systems Conference*, DIS '21, page 1817–1833, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384766. doi: 10.1145/3461778.3462050. URL `https://doi.org/10.1145/3461778.3462050`. 124

[246] Mihaly Csikszentmihalyi. *Toward a Psychology of Optimal Experience*, pages 209–226. Springer Netherlands, Dordrecht, 2014. ISBN 978-94-017-9088-8. doi: 10.1007/978-94-017-9088-8_14. URL `https://doi.org/10.1007/978-94-017-9088-8_14`. 124

[247] Tuck Wah Leong, Frank Vetere, and Steve Howard. Randomness as a resource for design. In *Proceedings of the 6th Conference on Designing Interactive Systems*, DIS '06, page 132–139, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933670. doi: 10.1145/1142405.1142428. URL `https://doi.org/10.1145/1142405.1142428`. 125

[248] Jhim Kiel M. Verame, Enrico Costanza, Joel Fischer, Andy Crabtree, Sarvapali D. Ramchurn, Tom Rodden, and Nicholas R. Jennings. Learning from the

veg box: Designing unpredictability in agency delegation. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–13, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356206. doi: 10.1145/3173574.3174021. URL https://doi.org/10.1145/3173574.3174021. 125

[249] Hugging Face. Hugging face, 2023. https://huggingface.co/ [Accessed: 15 November 2023]. 126

[250] Model Zoo. Model zoo, 2023. https://modelzoo.co/ [Accessed: 15 November 2023].

[251] Pytorch Hub. Pytorch hub, 2023. https://pytorch.org/hub/ [Accessed: 15 November 2023]. 126

[252] Kate Crawford. *The atlas of AI: Power, politics, and the planetary costs of artificial intelligence*. Yale University Press, 2021. 126, 135

[253] Nick Bryan-Kinns, Berker Banar, Corey Ford, Courtney N. Reed, Yixiao Zhang, Simon Colton, and Jack Armitage. Exploring XAI for the arts: Explaining latent space in generative music. In *eXplainable AI approaches for debugging and diagnosis.*, 2021. URL https://openreview.net/forum?id=GLhY_0xMLZr. 126, 136

[254] Tomas Lawton, Kazjon Grace, and Francisco J Ibarrola. When is a tool a tool? user perceptions of system agency in human–ai co-creative drawing. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*, DIS '23, page 1978–1996, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450398930. doi: 10.1145/3563657.3595977. URL https://doi.org/10.1145/3563657.3595977. 127

[255] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audiolm: A language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2523–2533, 2023. doi: 10.1109/TASLP.2023.3288409. 128

[256] Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. Audiogen: Textually guided audio generation. In *The Eleventh International Conference on Learning Representations*, 2023. 139

[257] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL `https://openreview.net/forum?id=ivCd8z8zR2`. Featured Certification, Reproducibility Certification. 142

[258] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021. 142

[259] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved RVQGAN. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=qjnl1QUnFA`. 142

[260] Robert Tubb and Simon Dixon. An evaluation of multidimensional controllers for sound design tasks. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 47–56, 2015. 142

[261] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 145, 146

[262] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017. 145, 146

[263] Ashis Pati and Alexander Lerch. Is disentanglement enough? on latent representations for controllable music generation. *CoRR*, abs/2108.01450, 2021. URL `https://arxiv.org/abs/2108.01450`. 151

[264] Christian J. Steinmetz and Joshua Reiss. pyloudnorm: A simple yet flexible loudness meter in python. In *Audio Engineering Society Convention 150*, May 2021. URL `http://www.aes.org/e-lib/browse.cfm?elib=21076`. 155