Creative Text-to-Audio Generation via Synthesizer Programming

Nikhil Singh*, **Manuel Cherep***, **Jessica Shand**
Massachusetts Institute of Technology
{nsingh1,mcherep,shand}@mit.edu

ctag.media.mit.edu

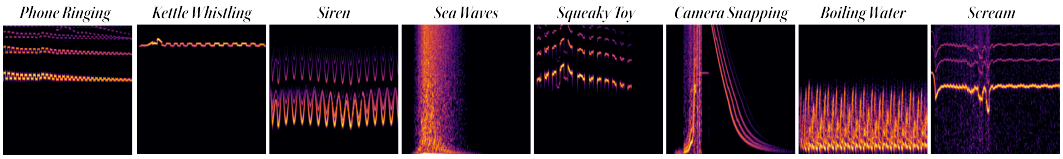

Figure 1: *CTAG* leverages a virtual modular synthesizer to generate sounds which capture the semantics of user-provided text prompts in a sketch-like way, rather than being acoustically literal. Spectrograms of auditory outputs corresponding to eight text prompts showcase the range of sounds this approach can yield, accompanied by a fully interpretable and controllable parameter space.

Abstract

Sound designers have long harnessed the power of abstraction to distill and highlight the semantic essence of real-world auditory phenomena, akin to how simple sketches can vividly convey visual concepts. However, current neural audio synthesis methods lean heavily towards capturing acoustic realism. We introduce an open-source novel method centered on meaningful abstraction. Our approach takes a text prompt and iteratively refines the parameters of a virtual modular synthesizer to produce sounds with high semantic alignment, as predicted by a pretrained audio-language model. Our results underscore the distinctiveness of our method compared with both real recordings and state-of-the-art generative models.

1 Introduction

> *"Of course, bubbles don't make sound, but this is the magic of sound design...you can create the concept of a sound and it seems real."* – Suzanne Ciani

In creative sound design, realism isn't everything. Suzanne Ciani's iconic 1970s *Coca Cola pop and pour* sound effect, which symbolizes the refreshing experience of opening a soda, was crafted with a Buchla synthesizer rather than recorded from a bottle. Her work illustrates the immense power of abstraction in auditory representation, where the essence of a concept can be expressed without mimicking real-world acoustic details, while achieving greater impact.

We recognize a gap in such meaningful abstraction within current neural audio synthesis paradigms, which prioritize acoustic fidelity over high-level semantics. Despite impressive advances, this limits expressive capabilities. We propose a novel method integrating a virtual modular synthesizer with a pretrained audio-language model, allowing for generating intuitively plausible sounds without literal representation. Our interpretable, controllable approach differs significantly from state-of-the-art models, offering benefits for creative work. We show examples in Figure 1.

*These authors contributed equally.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

2 Related Work

Sound Synthesis Analog synthesizers are popular, yet largely inaccessible due to size and cost [1]. Mass-produced digital synthesizers have historically enabled widespread exploration of sound synthesis [2], in both hardware and software. Recent neural approaches like WaveNet and WaveGAN have used neural networks to synthesize coherent audio content. *Oscillator* models like DDSP extend this by combining deep learning with classic signal processing elements. Our approach is motivated by such models, but recognizes that modular synthesizers remain integral to sound design today by offering an interpretable, controllable parameter space.

Language-Sound Correspondence Recent models have connected language and sound by learning multimodal embeddings. CLIP [3] pioneered this approach with contrastive learning on image-text pairs. CLAP [4] and LAION-CLAP [5] extended this approach to audio-text pairs. Contemporary text-to-audio approaches directly generate audio from text, often by using models like CLAP in their learning objectives. However, these models focus on realistic audio renditions. In contrast, we aim to generate abstract yet high-quality sounds using a controllable modular synthesizer guided by CLAP. This allows interpretability and fine-grained control compared to end-to-end neural models. We specifically compare to AudioLDM [6] and AudioGen [7].

Abstract Synthesis Visual sketching offers an intuitive analogy to abstract sound synthesis, using minimalist representations like line drawings to evocatively convey meaning while emphasizing essence over realism. This abstraction problem has seen more progress in images via models like CLIPasso [8], CLIPTexture [9], and ES-CLIP [10]. In audio, prior work like the Sound Sketchpad [11] and SkAT-VG [12] explores sketching interfaces for sound composition. Here, we focus on synthesizing novel abstract sounds from scratch using text descriptions and a modular synthesizer. We also generate, rather than recompose, sounds.

Interpretable and Controllable Synthesis Interpretability and controllability are essential for human-AI co-creation, yet black-box neural models lack these. Some work uses program synthesis [13] or timbre regularization [14] to improve interpretability. Our approach instead offers an interpretable, controllable modular synthesizer programmed with descriptions. This avoids complex neural models while enabling users to examine, understand, and refine sounds.

The Synthesizer Programming Problem Though synthesizers form the bedrock of much modern music, synthesizer programming remains challenging due to the disconnect between parameters and audio output [15]. Prior techniques like inverse synthesis [16, 17, 18] require target audio, limiting creative applications. We instead perform text-to-parameter inference, allowing users to imagine new sounds using language. This bridges the semantic gap while encouraging generative exploration beyond reconstructing existing audio. This approach also enables intuitive sound creation without specialized training.

3 Methods

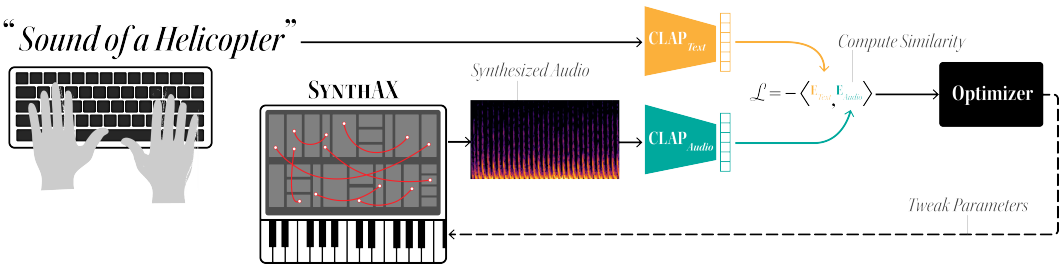

Figure 2: In our approach, we use the LAION-CLAP model [5] to compute the similarity between a user-provided text prompt and SYNTHAX's output. Then, we use the DES algorithm [19] to iteratively adjust parameter settings.

Our methodology hinges on three pillars (See Figure 2): a synthesizer, implemented via SYNTHAX [20], a gradient-free optimization method, implemented via the Evosax [21] evolutionary optimization library, and an objective function based on the LAION-CLAP [5] model, which we use to estimate semantic alignment between the synthesized audio and its corresponding text prompt.

Synthesizer We use a simple synthesizer available in SYNTHAX, a fast modular synthesizer written in JAX [22]. We use the *Voice* synthesizer architecture, adapted from *torchsynth* [23], which has already been used for programmatic synthesis [17]. All parameters are initialized uniformly, $p_i \sim U(0, 1)$. Audio is generated in batches at 48kHz, with 480Hz control rate, and producing 3 seconds of audio.

Optimization Early experiments showed gradients with this synthesizer to be highly unstable. Therefore, we decided to focus on non-gradient methods as have been used in other recent works on abstract synthesis to achieve strong results [10]. We experimented with the Evosax library [21], and selected the Discovered Evolution Strategy (DES) [19] algorithm, finding that it achieved better results faster than the majority of other tested algorithms.

We tuned the hyperparameters of DES via Bayesian optimization using the Adaptive Experimentation (AX) platform [24]. We ran the sweep for 50 trials on a subset of 20 common sounds taken from a dataset of 165 [25]. The sweeping relied on the average score of all 20 prompts for 300 iterations each. We swept the following parameters: $N \in [10, 100]$, $\beta \in [10.0, 30.0]$, $\alpha_\sigma \in [0.0, 1.0]$, $\alpha_m \in [0.0, 1.0]$ and $\sigma_0 \in [0.0, 1.0]$. The result was $\{N = 42, \beta = 11.47, \alpha_\sigma = 0.14, \alpha_m = 0.75, \sigma_0 = 0.18\}$.

Objective Function We use LAION-CLAP [5], with an HTSAT-based audio encoder [26], a RoBERTa-based text encoder [27], and the *audioset-best* checkpoint for general audio. The encoders process the audio data X_i^a and text data X_i^t in batches of size $\mathcal{B}$ where $\mathcal{B}$ corresponds to the optimizer's population size and (X_i^a, X_i^t) is one particular pair of synthesized audio with input text prompt. We extract the audio embeddings E_i^a and the text embeddings E_i^t with the encoders and use them to calculate the similarity score between a batch of audio data and a specific prompt.

A batch of audio $\mathcal{S}(\mathcal{P}_\mathcal{B}) = X_\mathcal{B}^a$ is produced by the synthesizer $\mathcal{S}$ from a batch of parameters $\mathcal{P}$. Then $\min_{\mathcal{P}_\mathcal{B}} -E_\mathcal{B}^{\mathcal{S}(\mathcal{P}_\mathcal{B})} E_i^{t\,T}$ formulates the optimization problem to optimize the similarity score between each audio in the batch and a given text prompt using their corresponding embeddings.

4 Results

4.1 Quantifying Abstractness

Quantitatively evaluating the abstract quality of synthesized sounds is challenging; no ready and validated metrics exist. Since achieving abstract sound synthesis with semantic correspondence is a novel contribution of our work, we propose experiments to touch on related aspects.

	AudioSet-50			**ESC-50**		
	AudioGen	*AudioLDM*	*CTAG*	*AudioGen*	*AudioLDM*	*CTAG*
Acc (Top-1)	0.54	0.25	0.21	0.44	0.24	0.12
Acc (Top-5)	0.80	0.56	0.46	0.63	0.41	0.25

Table 1: Top-1 and Top-5 classification accuracies (%) for two pre-trained classifiers corresponding to a subset of AudioSet (AudioSet-50) and ESC-50. We evaluated both models on results collected using *AudioGen*, *AudioLDM*, and our method, *CTAG*.

We use pre-trained classifiers to quantify the distribution shift to more abstract sounds. We evaluate on two datasets: ESC-50 [28], using a CNN14 classifier, and a 50-class subset of AudioSet, using an AST [29] model. Results are shown in Table 1. Our method shows lower accuracy than *AudioLDM* and *AudioGen*, indicating a shift away from realism, though still well above chance levels. On AudioSet-50 our results are close to AudioLDM, which we observe as having high realism but low prompt-consistency. This helps quantify the abstractness of our sounds, by illustrating their deviation from the distribution of natural sounds.

4.2 Synthesis Quality and Variation

Evaluating quality is challenging without reference audio clips, as in our case. We use spectral descriptors to quantify qualitative differences from other models (Table 2). Our sounds have higher spectral complexity, flux, high-frequency content (HFC), rolloff, and centroid. This suggests perceptual differences from other methods, like more high-frequency content due to our higher sample rate. The measures are not validated quality metrics, but help quantify observed qualitative differences.

Our sounds also achieve higher MP3 compression ratios than other methods. Variable bit rate compression uses lower ratios for more perceptually complex audio. The higher ratios for our sounds suggest they are perceptually "simpler" than sounds from other methods. This provides another quantitative signal that our sounds differ perceptually.

	AudioSet-50			**ESC-50**		
	AudioGen	*AudioLDM*	*CTAG*	*AudioGen*	*AudioLDM*	*CTAG*
Complexity	17.01	15.46	20.86	9.23	10.82	20.09
Flux	0.09	0.09	0.21	0.05	0.07	0.23
HFC	54.64	112.01	684.12	28.37	63.17	551.38
Rolloff	2,468.64	1,716.20	7,966.01	2,294.85	1,679.43	7,066.51
Centroid	1,633.61	1,157.99	4,672.28	1,510.73	1,176.92	4,139.26
Compression Ratio	6.62	7.28	9.71	6.60	7.72	9.73

Table 2: Comparison of spectral descriptors and audio compression ratio, across ESC-50 and AudioSet. Results are grouped by the evaluation of *AudioGen*, *AudioLDM*, and our method, *CTAG*.

4.3 Human Ratings

	AudioGen	*AudioLDM*	*CTAG*
Accuracy	0.70	0.57	0.45
Confidence	3.29	2.64	2.59
Artistic Interpretation	2.78	3.40	3.94

Table 3: User study results for sounds from *AudioGen*, *AudioLDM*, and our method, *CTAG*. We report average accuracy and confidence on label identification, and average rating of the artistic interpretiveness (1-5, realistic portrayal to artistic interpretation) of the sound.

We conducted a user study with 23 participants who rated 60 sounds each, with 20 sounds each from our method, *AudioLDM*, and *AudioGen*. Results are shown in Table 3. The randomly selected sounds covered 10 semantic categories: *Boiling Water*, *Camera Snapping Photos*, *Computer Startup Sound*, *Duck Quack*, *Hammering*, *Heart Beat*, *Scream*, *Squeaky Toy*, *Telephone Ringing*, and *Truck Beeping While Backing Up*. All prompts were taken from [25].

Participants identified the category of each sound and rated their confidence. Our sounds were identified with significantly lower accuracy compared to the other methods, as shown by post-hoc contrasts after a mixed-effects logistic regression model ($p < 0.001$). Participants also rated each sound's artistic interpretiveness versus realistic portrayal on a 5-point scale. Our sounds were rated as significantly more artistically interpretive, modeled with a linear mixed-effects model and tested with post-hoc contrasts ($p < 0.0001$).

5 Conclusion

In this work, we proposed a method for text-to-audio generation that offers a fresh perspective on neural audio synthesis by emphasizing the meaningful abstraction of auditory phenomena, contrary to prevalent methods that prioritize acoustic realism. Our results position this approach as a distinctive tool in the field of audio synthesis, capable of stimulating new directions in audio generation research that consider the level of abstraction as an important design parameter.

Acknowledgements

The authors acknowledge the MIT SuperCloud [30] and Lincoln Laboratory Supercomputing Center for providing resources that have contributed to the research results reported within this paper. This research was conducted with the partial support of a US-Spain Fulbright grant. We extend our heartfelt thanks to all participants in the user study.

References

[1] Trevor Pinch and Frank Trocco. *Analog days: The invention and impact of the Moog synthesizer*. Harvard University Press, 2004.

[2] Paul Théberge. *Any sound you can imagine: Making music/consuming technology*. Wesleyan University Press, 1997.

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[4] 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.

[5] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[6] 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. *arXiv preprint arXiv:2301.12503*, 2023.

[7] Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. Audiogen: Textually guided audio generation. *arXiv preprint arXiv:2209.15352*, 2022.

[8] Yael Vinker, Ehsan Pajouheshgar, Jessica Y Bo, Roman Christian Bachmann, Amit Haim Bermano, Daniel Cohen-Or, Amir Zamir, and Ariel Shamir. Clipasso: Semantically-aware object sketching. *ACM Transactions on Graphics (TOG)*, 41(4):1–11, 2022.

[9] Yiren Song. Cliptexture: Text-driven texture synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5468–5476, 2022.

[10] Yingtao Tian and David Ha. Modern evolution strategies for creativity: Fitting concrete images and abstract concepts. In *International conference on computational intelligence in music, sound, art and design (part of evostar)*, pages 275–291. Springer, 2022.

[11] Nikhil Singh. The sound sketchpad: Expressively combining large and diverse audio collections. In *26th International Conference on Intelligent User Interfaces*, pages 297–301, 2021.

[12] Davide Rocchesso, Guillaume Lemaitre, Patrick Susini, Sten Ternström, and Patrick Boussard. Sketching sound with voice and gesture. *interactions*, 22(1):38–41, 2015.

[13] Halley Young, Maxwell Du, and Osbert Bastani. Neurosymbolic deep generative models for sequence data with relational constraints. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 37254–37266. Curran Associates, Inc., 2022.

[14] Philippe Esling, Axel Chemla-Romeu-Santos, and Adrien Bitton. Bridging audio analysis, perception and synthesis with perceptually-regularized variational timbre spaces. In *International Society for Music Information Retrieval Conference*, 2018.

[15] Jordie Shier. The synthesizer programming problem: improving the usability of sound synthesizers, 2021.

[16] Matthew John Yee-King, Leon Fedden, and Mark d'Inverno. Automatic programming of vst sound synthesizers using deep networks and other techniques. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2(2):150–159, 2018.

[17] Masato Hagiwara, Maddie Cusimano, and Jen-Yu Liu. Modeling animal vocalizations through synthesizers. *arXiv preprint arXiv:2210.10857*, 2022.

[18] Philippe Esling, Naotake Masuda, and Axel Chemla-Romeu-Santos. Flowsynth: Simplifying complex audio generation through explorable latent spaces with normalizing flows. In *International Joint Conference on Artificial Intelligence*, 2020.

[19] Robert Tjarko Lange, Tom Schaul, Yutian Chen, Tom Zahavy, Valenti Dallibard, Chris Lu, Satinder Singh, and Sebastian Flennerhag. Discovering evolution strategies via meta-black-box optimization. *arXiv preprint arXiv:2211.11260*, 2022.

[20] Manuel Cherep and Nikhil Singh. SynthAX: A fast modular synthesizer in JAX. In *Audio Engineering Society Convention 155*. Audio Engineering Society, 2023.

[21] Robert Tjarko Lange. evosax: Jax-based evolution strategies. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation*, pages 659–662, 2023.

[22] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, et al. JAX: composable transformations of python+ numpy programs, 2018.

[23] Joseph Turian, Jordie Shier, George Tzanetakis, Kirk McNally, and Max Henry. One billion audio sounds from gpu-enabled modular synthesis. In *2021 24th International Conference on Digital Audio Effects (DAFx)*, pages 222–229. IEEE, 2021.

[24] Eytan Bakshy, Lili Dworkin, Brian Karrer, Konstantin Kashin, Benjamin Letham, Ashwin Murthy, and Shaun Singh. Ae: A domain-agnostic platform for adaptive experimentation. In *Conference on neural information processing systems*, pages 1–8, 2018.

[25] Sam Norman-Haignere, Nancy G Kanwisher, and Josh H McDermott. Distinct cortical pathways for music and speech revealed by hypothesis-free voxel decomposition. *neuron*, 88(6):1281–1296, 2015.

[26] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 646–650. IEEE, 2022.

[27] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[28] Karol J. Piczak. ESC: Dataset for environmental sound classification. *Proceedings of the 23rd ACM international conference on Multimedia*, 2015.

[29] Yuan Gong, Yu-An Chung, and James R. Glass. AST: Audio spectrogram transformer. *ArXiv*, abs/2104.01778, 2021.

[30] Albert Reuther, Jeremy Kepner, Chansup Byun, Siddharth Samsi, William Arcand, David Bestor, Bill Bergeron, Vijay Gadepally, Michael Houle, Matthew Hubbell, Michael Jones, Anna Klein, Lauren Milechin, Julia Mullen, Andrew Prout, Antonio Rosa, Charles Yee, and Peter Michaleas. Interactive supercomputing on 40,000 cores for machine learning and data analysis. In *2018 IEEE High Performance extreme Computing Conference (HPEC)*, pages 1–6. IEEE, 2018.