Temporal Modeling of Room Impulse Response Generation via Multi-Scale Autoregressive Learning

Sheng Lyu, Yuemin Yu, Chenshu Wu

School of Computing and Data Science, The University of Hong Kong

{shenglyu, yuyuemin}@connect.hku.hk, chenshu@cs.hku.hk

Abstract

The rise of AIGC has revolutionized multimedia processing, including audio applications. Room Impulse Response (RIR), which models sound propagation in acoustic environments, plays a critical role in various downstream tasks such as speech synthesis. Existing RIR generation methods, whether based on ray tracing or neural representations, fail to fully exploit the temporal dynamics inherent in RIR. In this work, we propose a novel method for temporal modeling of RIR through autoregressive learning. Our approach captures the sequential evolution of sound propagation by introducing a multi-scale generation mechanism that adaptively scales across varying temporal resolutions. Extensive evaluations demonstrate that our approach achieves respective T_{60} error rates of 4.1% and 5.3% on two real-world datasets, outperforming existing RIR generation methods. We believe our work opens up new directions for future research.

Index Terms: human-computer interaction, room acoustics, speech processing, room impulse response

1. Introduction

The rapid advancement of Artificial Intelligence Generated Content (AIGC) has transformed digital media production, enabling highly realistic multimodal content. Among these modalities, audio plays a vital role, as it conveys spatial and contextual cues in applications like virtual reality and speech recognition [1, 2]. At the heart of achieving such auditory realism lies the accurate modeling of acoustic environments [3, 4], which hinges on capturing how sound waves interact with physical spaces. These interactions are mathematically encapsulated in Room Impulse Responses (RIRs), which characterize the reverberation, attenuation, and diffraction of sound from a source to a listener. RIRs serve as the acoustic "fingerprint" of a space, enabling virtual sounds to mimic real-world propagation behavior [5]. Therefore, accurate and robust generation of RIRs arouses heated explorations.

However, traditional RIR estimation methods face significant limitations. Wave-based or ray-based approaches, such as finite-difference time-domain simulations [6, 7], often demand computationally expensive calculations and precise knowledge of room geometry and material properties. While data-driven methods leveraging machine learning have shown more promise [8, 9, 10, 11, 12, 13], the intrinsic temporal features of RIRs, critical for modeling physical reflections and reverberations over time, have been long neglected. Some recent methods [14, 15, 16, 17] leverage neural representations, yet their modeling highly depends on the room geometry. The lack of temporal modeling renders these solutions with limited performance.

Inspired by the recent success of large language models (LLMs) [18], which is largely attributed to autoregressive frameworks that predict the next token based on prior context, we adapt this sequential modeling paradigm to RIR generation. RIRs are essentially time-domain signals that describe the temporal propagation of sound as it interacts with surfaces in a room. This temporal progression, from the direct sound to early reflections and finally to late reverberations, creates a natural sequential structure that aligns with autoregressive frameworks. Each part of the RIR depends on the previous parts, as reflections and reverberations result from the propagation of earlier sound waves. This sequential dependency mirrors the structure of language data, motivating our adaptation of autoregressive principles to RIR modeling.

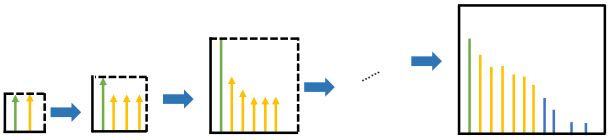

Figure 1: *The multi-resolution generation process of RIR*

While promising, the potential of temporal models for RIR generation via autoregressive learning remains underexplored. In this paper, we propose a novel generation scheme for RIR estimation through autoregressive learning. However, applying autoregressive models to generate RIRs is challenging [19, 20]. First, the sampling rate of RIRs is usually high; token-wise autoregressive generation incurs significant computational overhead. Second, the traditional next-token prediction scheme struggles to capture long-term dependencies, resulting in error accumulation that degrades overall RIR quality. These limitations necessitate a hierarchical approach to balance computational efficiency and temporal fidelity.

To this end, we design a multi-scale learning scheme that hierarchically models RIRs from coarse to fine resolutions. By discretizing RIRs into latent tokens at increasing scales, we autoregressively predict subsequent resolutions while preserving temporal coherence, as shown in Fig. 1. Additionally, we introduce a dedicated vector-quantized autoencoder to tokenize raw RIRs, enabling autoregressive generation across scales.

Extensive experiments demonstrate that our approach achieves an improvement of 28.1% in Reverberation Time (T_{60}) and 92.1% in Early Decay Time (EDT) over FastRIR [10]. Our model performs comparably to neural implicit and ray-tracing-based approaches while reducing inference time by 59.0%. Evaluations on speech synthesis datasets further show that our method introduces no additional word error rate compared to ground-truth RIRs.

Contributions: To the best of our knowledge, this paper

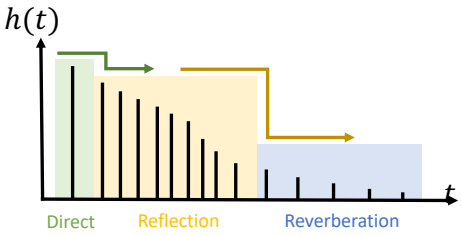

Figure 2: *The sequential structure of RIR*

presents the first-of-its-kind temporal modeling framework for RIR generation via multi-resolution autoregressive learning. We propose a novel token ordering scheme to capture both coarse- and fine-grained RIR features. Additionally, we design a vector-quantized autoencoder to tokenize raw RIRs and generate them through autoregressive modeling. Comprehensive experiments validate the efficacy of our approach.

2. Multi-scale Temporal Modeling

2.1. Autoregressive Learning and RIR

The success of large-scale language models like GPT has demonstrated the immense potential of autoregressive modeling in capturing sequential dependencies [21, 22, 20, 23]. At its core, autoregressive learning decomposes a joint probability distribution into a product of conditional probabilities, where each subsequent element in a sequence is predicted based on its predecessors. Autoregressive modeling excels at capturing dependencies within sequential data, enabling coherent and contextually relevant generation through step-by-step inference. While this approach has achieved significant success in natural language processing, its applicability extending to domains involving sequential data with inherent temporal structures, such as RIRs, is untouched. For RIRs, the temporal evolution of sound propagation aligns naturally with autoregressive principles. Mathematically, as shown in Fig. 2, an RIR is represented as a time-domain signal $h(t)$,

$$h(t) = \sum_{n=0}^{N} \alpha_n \delta(t - \tau_n) + r(t), \tag{1}$$

where N is the number of reflections, α_n is the amplitude of the n-th reflection and τ_n is the time delay. We use $\delta(\cdot)$ to represent the impulse response while $r(t)$ denotes the late reverberation component. RIRs capture how sound evolves from the direct source emission to interactions with surfaces and eventual decay. This process is inherently causal and sequential: the direct sound precedes early reflections [24, 13], which subsequently generate late reverberations. In other words, sound propagation adheres to strict causality, *i.e.*, future states depend on past events. This causal chain mirrors the step-by-step generation process of autoregressive models. By conditioning each prediction on previously generated samples, it can precisely replicate the temporal dynamics of RIRs with high fidelity.

2.2. Multi-Scale Temporal Modeling

A typical autoregressive framework involves three phases: quantization, autoencoding, and autoregressive generation. Quantization determines how the RIR is discretized into units and the sequence in which these tokens are predicted. The critical challenge lies in temporal modeling, *i.e.*, decomposing the RIR into discrete units while preserving the causal structure of sound propagation. Central to this process is the *order* of tokenization. Traditional text-based autoregressive models adopt next-token prediction, tokenizing data word-by-word or character-by-character. While effective for language, this approach is ill-suited for RIRs due to fundamental differences in data structure. RIRs are sampled at high rates (*e.g.*, 48 kHz) [19], producing sequences spanning tens of thousands of samples. Tokenizing at the sample level would incur prohibitive computational costs and struggle to model long-term dependencies like decay envelopes or reflection timing.

To address these limitations, we propose a multi-scale tokenization scheme that partitions the RIR into tokens at varying temporal granularities. Instead of sample-order tokenization, we represent the RIR hierarchically as $h^{(k)}(t)$, where k denotes the scale index. Inspired by the impulse energy α_k and delay τ_k in Eq. (1), each scale $h^{(k)}(t)$ captures progressively broader temporal and amplitude ranges, expanding from early/high-amplitude components (direct sound) to late/low-amplitude features (reverberation). At scale k, the representation $h^{(k)}(t)$ is derived by applying temporal and amplitude scaling operators $\mathcal{T}_k$ and $\mathcal{A}_k$ to the original RIR $h(t)$:

$$h^{(k)}(t) = \mathcal{T}_k(h(t)) \odot \mathcal{A}_k(h(t)), \tag{2}$$

where $\odot$ denotes element-wise multiplication. The temporal operator $\mathcal{T}_k$ restricts $h(t)$ to a window $[t_{\text{start}}^{(k)}, t_{\text{end}}^{(k)}]$:

$$\mathcal{T}_k(h(t)) = \begin{cases} h(t), & t \in [t_{\text{start}}^{(k)}, t_{\text{end}}^{(k)}], \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

with linearly expanding boundaries:

$$t_{\text{start}}^{(k)} = 0, \quad t_{\text{end}}^{(k)} = \gamma_k T_{\max}. \tag{4}$$

Here, γ_k increases linearly with k ($\gamma_k > \gamma_{k-1}$). The amplitude operator $\mathcal{A}_k$ scales $h(t)$ to emphasize specific ranges:

$$\mathcal{A}_k(h(t)) = \sigma_k(A) \cdot h(t), \quad \sigma_k(A) = \eta_k A_{\max}, \tag{5}$$

where η_k decreases linearly with k ($\eta_k < \eta_{k-1}$).

This hierarchical tokenization captures coarse-to-fine RIR features. Coarse scales focus on prominent early components (direct sound, major reflections), while finer scales resolve late reverberations and subtle amplitude variations. Tokens are generated progressively, with finer resolutions conditioned on coarser ones:

$$P(H^{(k)}) = \prod_{m=1}^{M} P\left(h_m^{(k)} \middle| H^{(1)}, H^{(2)}, \ldots, H^{(k-1)}\right), \tag{6}$$

where $H^{(k)}$ denotes tokens at resolution k and $h_m^{(k)}$ is the m-th token. This conditional dependency preserves the causal sound propagation structure, mirroring the natural energy evolution in rooms: direct sound establishes the envelope, followed by reflections that shape reverberation tails.

3. RIR Generation

In this section, we will present our RIR generation pipeline design. At a high level, our framework involves: (1) training a multi-resolution Vector Quantization (VQ) autoencoder to compress RIR signals into a discrete latent space, and (2) training an autoregressive model to learn the temporal dynamics of RIRs conditioned on room geometry.

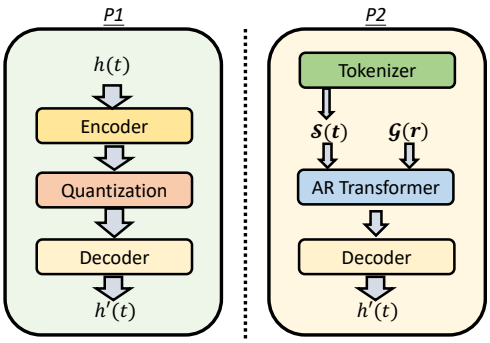

Figure 3: *Dual-Phase Training Procedure. At Phase 1 (P1), the network learns the codebooks through a VQ autoencoder using the multi-resolution scale scheme. At Phase 2 (P2), the network accepts the geometry condition vector $\mathcal{G}(r)$ and the token $\mathcal{S}(t)$ to predict the corresponding token keys.*

3.1. Multi-Resolution Vector Quantization Autoencoder

Consider the RIR $h(t)$, the VQ tokenizer wants to learn the the discrete latent $\mathcal{S}(t)$ though the encoder and a quantizer, *i.e.*,

$$\begin{aligned} \mathcal{E}(t) &= \mathbf{E}\left(h(t)\right) \\ \mathcal{E}(t) &\xrightarrow{\text{Scale}} \mathcal{E}^{(k)}(t) \\ \mathcal{S}(t) &= \mathbf{Q}(\mathcal{E}^{(k)}(t)) \end{aligned} \tag{7}$$

where $\mathbf{E}(\cdot)$ represents the encoder and $\mathbf{Q}(\cdot)$ is the vector quantizer. During this process, the encoder compresses the input RIR into a continuous latent representation. The continuous latent representation is later mapped to the nearest embedding vector in a discrete codebook $\mathcal{C}$ via the vector quantizer, resulting in a discrete latent code. Note that the sequence that we feed the $\mathbf{E}(t)$ into the quantizer follows Eq. (2), where we scale the continuous latent along the temporal and amplitude dimensions. At the training stage, the decoder $\mathbf{D}(t)$ takes the quantized latent representation and reconstructs the original input data $h'(t)$.

The goal is to minimize the reconstruction error between the input and output, while minimizing the distance between continuous latent representation and the embedding vector. To train the RIR-aware quantized autoencoder, we incorporate a combination of different losses, *i.e.*,

$$\mathcal{L}_q = \lambda_c \cdot \mathcal{L}_c + \lambda_t \cdot \mathcal{L}_t + \lambda_f \cdot \mathcal{L}_f + \lambda_g \cdot \mathcal{L}_g \tag{8}$$

where $\mathcal{L}_c$ is the commitment loss, which penalizes deviations between the encoder's output $\mathcal{E}(t)$ and the selected embedding vector $\mathcal{S}(t)$, ensuring that the encoder commits to the discrete latent space. $\mathcal{L}_t$ and $\mathcal{L}_f$ compare the signals with the ground truth in the temporal and frequency domain, respectively. $\mathcal{L}_f$ involves the MSE loss between the amplitude and phase in the spectrum. At last, the generator loss $\mathcal{L}_g$ is included to measure how well the generator is able to fool the discriminator into classifying the generated faked RIR as real.

3.2. Autoregressive Prediction

Once the multi-resolution VQ autoencoder has been trained, it can be used as the foundation for autoregressive prediction. Specifically, the autoregressive model will map the discrete latent codes and learn to predict the next latent code in a ordered sequence, enabling the generation of new RIR samples.

Notably, RIR compresses the geometry information in the temporal responses. Normally, the geometry information can include the room size $c_{\text{room}}(\boldsymbol{r})$, the speaker location $c_{\text{Tx}}(\boldsymbol{r})$ and the microphone location $c_{\text{Rx}}(\boldsymbol{r})$, where $\boldsymbol{r}$ is a 3D Cartesian coordinate. Additionally, to feature the propagation in the room, we incorporate Reverberation Time (T_{60}) as one condition [10]. We concatenate them together to form the room geometry and embed it into the geometry condition vector $\mathcal{G}(\boldsymbol{r})$. To this end, the autoregressive model predicts the next latent code s_t given the previous codes $s_{<t}$ and the geometry condition vector $\mathcal{G}(\boldsymbol{r})$,

$$P(s_t|s_{<t}, \mathcal{G}(\boldsymbol{r})) = \mathbf{R}(s_{<t}, \mathcal{G}(\boldsymbol{r})). \tag{9}$$

We use the cross-entropy loss to train the autoregressive model,

$$L_R = -\sum_t \log P(s_t|s_{<t}, \mathcal{G}(\boldsymbol{r})). \tag{10}$$

At the inference stage, the RIR is reconstructed conditioned on the room geometry condition $\mathcal{G}(\boldsymbol{r})$. We first sample the latent codes from the learnt distribution,

$$s_t \sim P(s_t|s_{<t}, \mathcal{G}(\boldsymbol{r})). \tag{11}$$

We append s_t continuously until the desired length is reached. These discrete latent codes will then be used to look up the corresponding embedding vectors and reconstruct the RIR:

$$h'(t) = \mathbf{D}(\mathcal{C}(s)). \tag{12}$$

4. Evaluation

4.1. Implementation Details

Backbone: We use SEANet [26] as the backbone of autoencoder. It converts temporal RIR into latent expressions using series of convolutional layers. The latent representation is then quantized by the residual vector quantizer [19]. When training the VQ autoencoder, we also follow the design of adversarial training, where the generator is trained to fool the discriminator into classifying the generated RIR as real. We adopt common decoder-only transformer similar to previous works [20] for the implementation of autoregressive model.

Training Details: For the tokenizer, we use Adam optimizer and a learning rate of 1e-5, with a cosine learning rate scheduler. The scaling is done by interpolation in practice. We set the weight of each loss component as equivalent, and we train the model for 50 epochs with a batch size of 16, with the first two epochs as warm-up using a learning rate proportion of 0.005 and 0.01. For the autoregressive model, we use Adam optimizer and a linear learning rate scheduler with a learning rate of 1e-4. We train the model for 20 epochs. The training is conducted on a single NVIDIA A100 GPU.

Dataset: We train and evaluate our model's performance on real-world datasets by adopting two commonly used room impulse response datasets: MeshRIR [27] and Real Acoustic Field (RAF) [25]. MeshRIR collects RIRs in a cuboidal room. We use S1-M3969 dataset split featuring a fixed single speaker for evaluation and the RIRs are resampled to a 24 KHz sampling rate. We use the empty office settings from RAF. We use 90% of the data to train and the rest 10% for testing.

Metrics: We use comprehensive metrics to evaluate our work. These include T_{60} percentage error, Early Decay Time (EDT) error as well as the phase and amplitude error. Schroeder's reverse integration method is utilized to derive the energy decay of the RIRs [7, 28]. The EDT is calculated as the time taken for

Table 1: *Overall Performances on different datasets: MeshRIR and RAF*

Method	**MeshRIR [10]**				**RAF [25]**			
Metrics	Phase	Amp	EDT (ms)	$T_{60}(\%)$	Phase	Amp	EDT (ms)	T_{60} (%)
FastRIR [10]	1.61	1.00	628.4	32.13	1.62	5.03	520.0	48.7
NAF [14]	1.61	0.75	39.0	4.21	**1.27**	2.24	36.14	6.5
AVR [16]	**1.28**	**0.15**	**10.99**	**3.90**	1.62	**0.33**	24.52	6.19
Ours	**1.62**	**0.34**	**49.62**	**4.05**	**1.62**	**0.78**	**24.28**	**5.34**

the energy to decay by 10 dB. The phase and amplitude errors are calculated as the mean absolute error between the generated and ground truth RIRs in the frequency domain.

Baselines: The primary focus of this paper is the generation of RIRs. To evaluate our approach, we compare it with three existing methods: FastRIR [10], NAF [14], and AVR [16]. Among these, FastRIR employs a purely generative approach, while NAF and AVR incorporate neural implicit modeling which require intensive computing and modeling.

4.2. Overall Results

The results are available in Table 1. Our method demonstrates significant improvements in RIR generation quality. On the MeshRIR and RAF datasets, we achieve T_{60} errors of 4.05% and 5.0%, respectively, outperforming the generative method FastRIR by factors of 8x and 9x. Furthermore, our approach achieves an EDT error of 24.28 ms on the RAF dataset, a substantial improvement over FastRIR, which renders an EDT of 520.0 ms. In terms of amplitude error, our method reduces the error by 1.9% and 5.7% compared to FastRIR. These results highlight the effectiveness of our proposed temporal modeling for enhancing the accuracy and quality of RIR generation.

We also benchmark our method against neural-based approaches, *i.e.*, NAF and AVR, which leverage neural representations to model room geometry comprehensively. While these methods excel in capturing detailed room characteristics, our approach achieves comparable performance without requiring intricate geometric modeling. Notably, our method achieves a marginal T_{60} error of just 0.15% compared to AVR and even outperforms NAF in T_{60} accuracy. On the RAF dataset, our model surpasses AVR and NAF by 16% and 22%, respectively, in terms of T_{60} error. Additionally, our EDT error of 24.28 ms outperforms both NAF and AVR, further underscoring the strength of our approach. These findings demonstrate the advantages of our proposed generation model. Compared to complex neural implicit modeling techniques, our method achieves competitive performance while significantly reducing the need for detailed room geometry, thereby alleviating the burden of labor-intensive data preparation.

4.3. Inference Time

We conduct a comparative analysis of inference time between our model and existing approaches. Despite our model having the largest parameter size, it exhibits the lowest inference overhead among the four methods considered. Specifically, our approach reduces inference time by 59.0% compared to AVR and 49.3% compared to NAF. This highlights the superior scalability and efficiency of our model for RIR generation tasks. The significant reduction in computational overhead can be attributed to the adoption of autoregressive modeling, which streamlines the generation process while maintaining high performance. These results underscore the practical advantages of our method in delivering both accuracy and efficiency.

Table 2: *Model Size and Inference Time*

Model	Number of Params	Inference Overhead (ms)
FastRIR	115.27M	159.88
AVR	57.24M	135.27
NAF	2.1M	109.2
Ours	306.97M	55.4

4.4. Spatial Speech Synthesis and ASR

The RIR is commonly employed to generate spatial speech [29, 10, 30]. In our approach, reverberant speech is synthesized by convolving clean speech from the LibriSpeech dataset [31] with randomly sampled RIRs. To evaluate the quality of the generated speech, we decode the simulated reverberant speech using two prominent Automatic Speech Recognition (ASR) systems: Google Speech API and Meta Wit Speech API. The performance is assessed in terms of Word Error Rate (WER). When using Google Speech API, the reverberant speech generated with the original RIR results in a WER of 12.38%. Notably, when employing our generated RIRs, the WER remains unchanged, indicating no degradation in performance. Similarly, with the Meta Wit Speech API, the WER for both the original and generated RIRs is consistently 18.97%. These results demonstrate that our generated RIRs effectively replicate the acoustic effects of real environments while avoiding any additional increase in speech recognition errors. This highlights the fidelity of our RIR generation process in mimicking real-world geometric and acoustic conditions.

5. Conclusion and Future Directions

In this paper, we introduce a novel temporal modeling framework for Room Impulse Response (RIR) generation. Our approach leverages autoregressive learning to capture the sequential nature of sound propagation, paired with a multi-scale generation mechanism that dynamically adapts to varying temporal resolutions. This enables precise modeling of both transient acoustic events and long-term reverberation patterns. Experimental results demonstrate that our method outperforms existing RIR generation techniques by 28.1% in terms of T_{60}.

In the future, we plan to enhance the framework by integrating room geometry embeddings with energy-decaying acoustic features, which could further refine spatial accuracy. Additionally, we will combine neural ray tracing with autoregressive learning to improve robustness across diverse environments.

6. ACKNOWLEDGMENT

This work is supported by NSFC Grant No. 62222216, Hong Kong RGC GRF Grant No. 17212224, RGC ECS Grant No. 27204522, and RGC HLCA Grant No. HLCA/E-712/22.

7. References

[1] A. Ratnarajah, I. Ananthabhotla, V. K. Ithapu, P. Hoffmann, D. Manocha, and P. Calamia, "Towards improved room impulse response estimation for speech recognition," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[2] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[3] S. Lyu and C. Wu, "Ase: Practical acoustic speed estimation beyond doppler via sound diffusion field," *arXiv preprint arXiv:2412.20142*, 2024.

[4] Y. Shao, S.-X. Zhang, and D. Yu, "Rir-sf: Room impulse response based spatial feature for target speech recognition in multi-channel multi-speaker scenarios," in *Proc. Interspeech 2024*, 2024, pp. 4988–4992.

[5] Y. Khokhlov, T. Prisyach, A. Mitrofanov, D. Dutov, I. Agafonov, T. Timofeeva, A. Romanenko, and M. Korenevsky, "Classification of room impulse responses and its application for channel verification and diarization," in *Proc. Interspeech 2024*, 2024, pp. 3250–3254.

[6] S. T. Neely and J. B. Allen, "Invertibility of a room impulse response," *The Journal of the Acoustical Society of America*, vol. 66, no. 1, pp. 165–169, 1979.

[7] M. R. Schroeder, "New method of measuring reverberation time," *The Journal of the Acoustical Society of America*, vol. 37, no. 3, pp. 409–412, Mar. 1965. [Online]. Available: https://doi.org/10.1121/1.1909343

[8] A. Ratnarajah, Z. Tang, and D. Manocha, "Ts-rir: Translated synthetic room impulse responses for speech augmentation," in *2021 IEEE automatic speech recognition and understanding workshop (ASRU)*. IEEE, 2021, pp. 259–266.

[9] A. Ratnarajah, S. Ghosh, S. Kumar, P. Chiniya, and D. Manocha, "Av-rir: Audio-visual room impulse response estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 27 164–27 175.

[10] A. Ratnarajah, S.-X. Zhang, M. Yu, Z. Tang, D. Manocha, and D. Yu, "Fast-rir: Fast neural diffuse room impulse response generator," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 571–575.

[11] M. Wang, S. Clarke, J.-H. Wang, R. Gao, and J. Wu, "Soundcam: a dataset for finding humans using room acoustics," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[12] S. Lee, H.-S. Choi, and K. Lee, "Yet another generative model for room impulse response estimation," in *2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2023, pp. 1–5.

[13] A. Ratnarajah, Z. Tang, and D. Manocha, "Ir-gan: Room impulse response generator for far-field speech recognition," *arXiv preprint arXiv:2010.13219*, 2020.

[14] A. Luo, Y. Du, M. Tarr, J. Tenenbaum, A. Torralba, and C. Gan, "Learning neural acoustic fields," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35. Curran Associates, Inc., 2022, pp. 3165–3177.

[15] M. L. Wang, R. Sawata, S. Clarke, R. Gao, S. Wu, and J. Wu, "Hearing anything anywhere," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 11 790–11 799.

[16] Z. Lan, C. Zheng, Z. Zheng, and M. Zhao, "Acoustic volume rendering for neural impulse response fields," *arXiv preprint arXiv:2411.06307*, 2024.

[17] A. Ratnarajah, Z. Tang, R. Aralikatti, and D. Manocha, "Mesh2ir: Neural acoustic impulse response generator for complex 3d scenes," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 924–933.

[18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[19] K. Qiu, X. Li, H. Chen, J. Sun, J. Wang, Z. Lin, M. Savvides, and B. Raj, "Efficient autoregressive audio modeling via next-scale prediction," 2024. [Online]. Available: https://arxiv.org/abs/2408.09027

[20] K. Tian, Y. Jiang, Z. Yuan, B. PENG, and L. Wang, "Visual autoregressive modeling: Scalable image generation via next-scale prediction," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. [Online]. Available: https://openreview.net/forum?id=gojL67CfS8

[21] J. Xiong, G. Liu, L. Huang, C. Wu, T. Wu, Y. Mu, Y. Yao, H. Shen, Z. Wan, J. Huang *et al.*, "Autoregressive models in vision: A survey," *arXiv preprint arXiv:2411.05902*, 2024.

[22] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 873–12 883.

[23] J. Lu, C. Clark, S. Lee, Z. Zhang, S. Khosla, R. Marten, D. Hoiem, and A. Kembhavi, "Unified-io 2: Scaling autoregressive multimodal models with vision language audio and action," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 26 439–26 455.

[24] I. Szöke, M. Skácel, L. Mošner, J. Paliesek, and J. Černockỳ, "Building and evaluation of a real room impulse response dataset," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, pp. 863–876, 2019.

[25] Z. Chen, I. D. Gebru, C. Richardt, A. Kumar, W. Laney, A. Owens, and A. Richard, "Real acoustic fields: An audio-visual room acoustics dataset and benchmark," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 886–21 896.

[26] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," 2022. [Online]. Available: https://arxiv.org/abs/2210.13438

[27] S. Koyama, T. Nishida, K. Kimura, T. Abe, N. Ueno, and J. Brunnström, "Meshrir: A dataset of room impulse responses on meshed grid points for evaluating sound field analysis and synthesis methods," in *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2021, pp. 1–5.

[28] M. R. Schroeder, "Complementarity of sound buildup and decay," *The Journal of the Acoustical Society of America*, vol. 40, no. 3, pp. 549–551, Sep. 1966. [Online]. Available: https://pubs.aip.org/jasa/article/40/3/549/746670/Complementarity-of-Sound-Buildup-and-Decay

[29] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 5220–5224.

[30] Z. Tang, L. Chen, B. Wu, D. Yu, and D. Manocha, "Improving reverberant speech training using diffuse acoustic simulation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6969–6973.

[31] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.