Few-shot Acoustic Synthesis with Multimodal Flow Matching

Supplementary Material

In the supplementary material, we first present details of our VAE (Sec. A), FLAC (Sec. B), and AGREE (Sec. C). We then provide additional information on our evaluation setup, including datasets, metrics, and baselines (Sec. D). Further results are reported in Sec. E, such as performances on the seen set of AcousticRooms, on HAA with a reversed setup, and complementary evaluation metrics. Additional qualitative examples, including a video, are provided in Sec. F. We give details about the perceptual evaluation setup in Sec. G. Finally, we provide the number of parameters and inference speed of models in Sec. H.

A. VAE

A.1. Training objective

We provide details on each loss used to train the VAE. In the following, let $\mathbf{x}$ and $\hat{\mathbf{x}}$ denote the ground truth and predicted waveforms and $\mathbf{X}$, $\widehat{\mathbf{X}}$ the magnitudes of their STFT representations.

We employ a multiresolution STFT loss $\mathcal{L}_{\text{MR}}$ inspired by [15, 19]. This loss compares the spectrograms of the ground-truth and predicted waveforms at m different resolutions using the spectral convergence $\mathcal{L}_{\text{SC}}$, log-magnitude $\mathcal{L}_{\text{SL}}$ and energy-decay losses $\mathcal{L}_{\text{ED}}$:

$$\mathcal{L}_{\text{MR}}(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{i=1}^{m} \mathcal{L}_{\text{SC}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i) + \mathcal{L}_{\text{SL}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i) + \mathcal{L}_{\text{ED}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i), \quad (1)$$

with

$$\mathcal{L}_{\text{SC}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i) = \frac{||\mathbf{X}_i - \widehat{\mathbf{X}}_i||_F}{||\mathbf{X}_i||_F}, \quad (2)$$

$$\mathcal{L}_{\text{SL}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i) = ||\log(\mathbf{X}_i + \eta) - \log(\widehat{\mathbf{X}}_i + \eta)||_1, \quad (3)$$

$$\mathcal{L}_{\text{ED}}(\mathbf{X}_i, \widehat{\mathbf{X}}_i) = ||10\log_{10}\text{E}(\mathbf{X}_i) - 10\log_{10}\text{E}(\widehat{\mathbf{X}}_i)||_1, \quad (4)$$

where

$$\text{E}(\mathbf{X}_i) = \sum_{k=d}^{T}\sum_{f=1}^{F}(\mathbf{X}_i(f,k))^2,\ 1 \le d \le T. \quad (5)$$

To further improve the quality of the generated samples, we employ an adversarial hinge loss $\mathcal{L}_{\text{adv}}$ and feature matching loss $\mathcal{L}_{\text{feat}}$, based on the multi-scale STFT discriminator from Encodec [2]. Multi-scale discriminators are well suited for capturing structures in audio signals [4, 5, 20]. The discriminator consists of multiple identically structured networks operating on multi-scaled complex-valued STFT, with the real and imaginary parts concatenated. Each sub-network is composed of a 2D convolutional layer, followed by 2D convolutions with increasing dilation rates of 1, 2 and 4 in the time dimension and a stride of 2 along the frequency axis. A final 2D convolution with kernel size 3×3 and stride $(1, 1)$ produces the output prediction. We use 5 scales with STFT window lengths $[2048, 1024, 512, 256, 128]$, hop lengths $[512, 256, 128, 64, 32]$ and FFT sizes $[2048, 1024, 512, 256, 128]$.

The losses are expressed as:

$$\mathcal{L}_{\text{adv}}(\mathbf{x}, \hat{\mathbf{x}}) = \sum_{n=1}^{N} \max[0, 1 - D_n(\mathbf{x})] + \max[0, 1 + D_n(\hat{\mathbf{x}})], \quad (6)$$

$$\mathcal{L}_{\text{feat}}(\mathbf{x}, \hat{\mathbf{x}}) = \frac{1}{NL}\sum_{n=1}^{N}\sum_{l=1}^{L}\frac{||D_n^l(\mathbf{x}) - D_n^l(\hat{\mathbf{x}})||_1}{\text{mean}(||D_n^l(\mathbf{x})||_1)}, \quad (7)$$

where D_n^l is the l-th layer of the n-th discriminator D_n.

Finally, we use a KL divergence loss $\mathcal{L}_{\text{KL}}$ to regularize the latent distribution. The KL loss is given by:

$$\mathcal{L}_{\text{KL}}(\mathbf{x}) = D_{\text{KL}}\big(q_E(\mathbf{z}|\mathbf{x}) \,\|\, \mathcal{N}(0, \mathbf{I})\big), \quad (8)$$

where $q_E(\mathbf{z}|\mathbf{x})$ denotes the encoder's approximate posterior, μ_j and σ_j are the mean and standard deviation of the j-th dimension of the latent variable z and d is the latent dimensionality.

A.2. Implementation details

We train the VAE using the AdamW optimizer [7] with a batch size of 64 on a single H100 GPU. The generator is optimized with a learning rate of 1.5×10^{-5}, the discriminator uses 3×10^{-5}. For the loss terms, all components of the multiresolution STFT loss are equally weighted, the KL loss is weighted by 1×10^{-4}, the adversarial loss by 0.1, and the feature matching loss by 5.0. The code is based on the stable audio tools library [1].

B. FLAC

B.1. Implementation details

When training on the synthetic dataset, we apply two forms of data augmentation to improve robustness: (i) a small random time shift of up to 10 samples in the time domain, and

[1] https://github.com/Stability-AI/stable-audio-tools

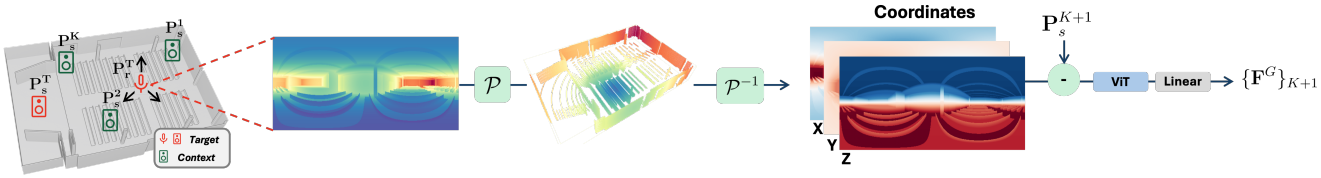

Figure A.1. **Geometry module pipeline:** A panoramic depth map captured at the receiver position is unprojected into a 3D point cloud using the equirectangular projection $\mathcal{P}$, then reprojected so that each pixel encodes its corresponding 3D coordinates. We subtract the source pose associated with the RIR (previously projected in the receiver frame). The resulting representation is processed by a ViT followed by a linear layer to produce geometry-aware features. In HAA, the source and receiver are interchanged.

(ii) the addition of pink noise with a randomly chosen SNR between 40 and 60 dB.

Our DiT model consists of 12 transformer blocks with 8 heads and a hidden width of 256. We train it using a flow matching objective using a learning rate of 5×10^{-5}, AdamW optimizer [7] and a batch size of 64 on a single H100 GPU. We use an Exponential Moving Average (EMA) of the model weights during training and BF16 precision.

B.2. Illustration of the geometry module

We illustrate the geometry module in Fig. A.1.

B.3. Illustration of the DiT variants

In Sec. 5.6, we compare our *AdaLN+CA* DiT architecture (shown in Fig. 3) with two alternative designs: the *In-Context* and the *Cross-Attention* (CA-only) variants. Both alternative architectures are illustrated in Fig. A.2.

B.4. Conditioning: geometry and materials

Panoramic depth maps do not recover occluded surfaces, creating ambiguity that directly motivates our stochastic formulation. While richer geometry (e.g., meshes) could reduce this, it introduces heavier assumptions. FLAC (1-shot) achieves strong performance, and depth can be estimated from RGB using foundation models, making an RGB + 1-RIR setup practical.

Similarly, while material-aware modeling have shown to improve prediction [11, 13], explicit annotations are unavailable in real-world datasets like HAA. Instead, we rely on implicit cues in conditioning RIRs, shown to capture room-material properties [6].

C. AGREE

C.1. Training objective

Given a batch B of geometry embeddings $\mathbf{G} \in \mathbb{R}^{B\times d}$ and acoustic embeddings $\mathbf{A} \in \mathbb{R}^{B\times d}$, our model is trained using a symmetric contrastive objective analogous to CLIP [9]. We first compute pairwise similarity logits

$$\mathbf{L}_G = \lambda\, \mathbf{G}\mathbf{A}^\top, \qquad \mathbf{L}_A = \lambda\, \mathbf{A}\mathbf{G}^\top, \tag{9}$$

where λ is a learnable logit scaling parameter. Each row of $\mathbf{L}_G$ (resp. $\mathbf{L}_A$) contains the similarities between one geometry (resp. acoustic) embedding and all acoustic (resp. geometry) embeddings in the same batch. The ground-truth alignment corresponds to matching indices, $\mathbf{y} = (1, \dots, B)$, and the loss is defined as

$$\mathcal{L}_{\text{contrast}} = \frac{1}{2}\Big(\text{CE}(\mathbf{L}_G, \mathbf{y}) + \text{CE}(\mathbf{L}_A, \mathbf{y})\Big), \tag{10}$$

where CE denotes cross-entropy loss. It encourages aligned geometry-acoustic pairs to have high similarity while pushing apart mismatched pairs.

C.2. Implementation details

AGREE operates on waveforms sampled at 22.05 kHz of length 10,240, matching the training data from AcousticRooms [6] and HAA [18].

For geometry, we use 256×512 panoramic depth maps captured at each receiver location in AcousticRooms (and at each source location in HAA). Following FLAC, depth maps are unprojected via equirectangular projection to obtain 3D points, and projected back in the image space so each pixel contains 3D coordinates. Then, we subtract the source pose expressed in the receiver frame (or the receiver pose for HAA). This provides the geometric context around the RIR's recording configuration. Fig. A.1 illustrates this process.

We jointly fine-tune DINOv3 ViT-S/16 [14] encoder and our frozen VAE audio encoder pretrained on AcousticRooms. A linear layer maps each encoder's output into a shared 512-dimensional embedding space.

Training uses AdamW [7] with a learning rate of $1e^{-4}$, cosine decay, 10,000 warm-up steps, and a weight decay of 0.1. We employ a batch size of 128 and an embedding dimensionality of 512. The model is trained for 100 epochs. We base our implementations on OpenCLIP [3] [2].

C.3. Impact of the geometry encoder

We compare several geometry encoder setup. Specifically, we compare four ViT variants: the encoder from [6], the

[2] https://github.com/mlfoundations/open_clip

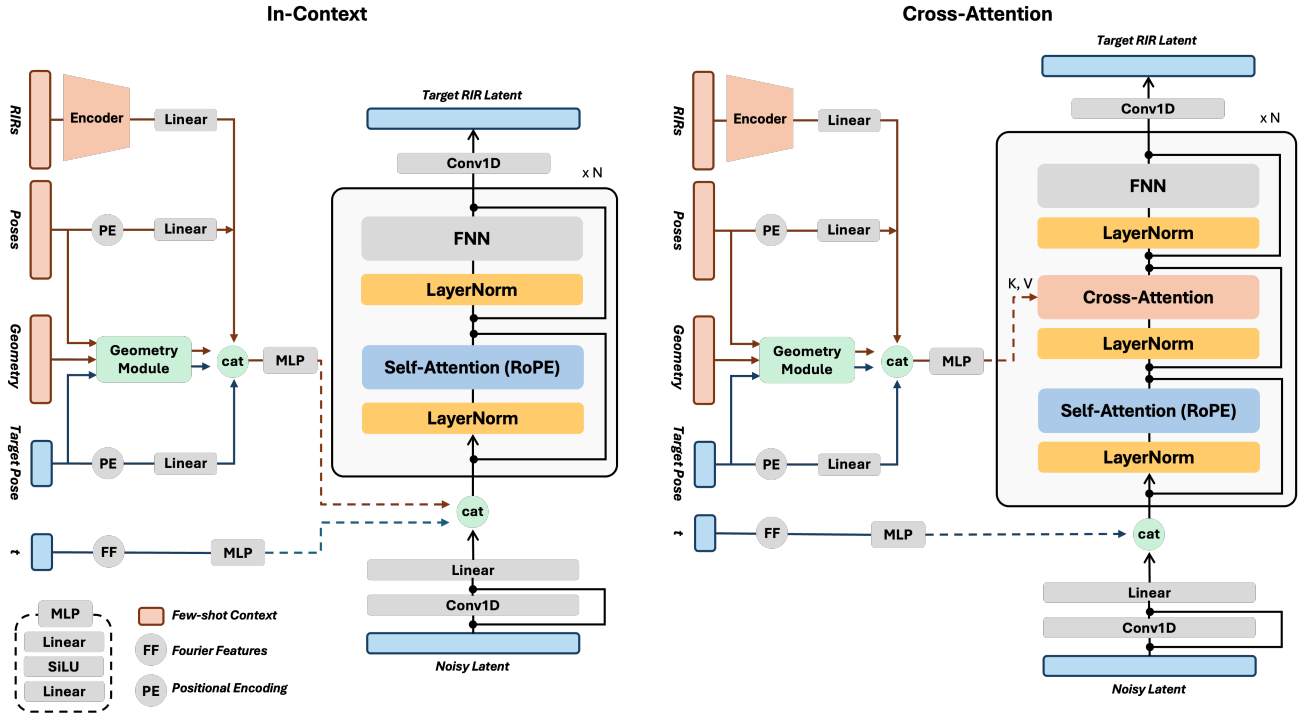

Figure A.2. **DiT architecture variants:** *In-Context* and *Cross-Attention*–only architectures. The *In-Context* variant concatenates all conditioning information with the input before self-attention. In the *Cross-Attention* variant, conditioning is applied solely via cross-attention layers.

ViT-S/16 implementations from OpenCLIP [3] and DINOv3, and the larger DINOv3 ViT-S+/16. For DINOv3, we assess the impact of using the pre-trained weights. Zero-shot retrieval results reported in Tab. A.1.

With no specific weights initialization ($\mathcal{W}_{\text{DINO}} = \text{✗}$), using the ViT-S/16 from DINOv3 achieves the best performance. Using frozen DINO weights is less effective, likely because our geometric representation differs significantly from the RGB images on which DINOv3 was pretrained. The strongest results are obtained by fine-tuning DINOv3 on our task, with ViT-S+/16 providing the best zero-shot retrieval on most metrics.

Since in this work AGREE is primarily used as a shared embedding space for evaluating the scene consistency of generated RIRs, we also train it on the full AcousticRooms dataset. In this setting, the performance gap between ViT-S and ViT-S+ becomes negligible, and we therefore adopt the smaller ViT-S/16 in the main experiments.

C.4. AGREE vs. CRIP

AGREE differs from CRIP [11] in three aspects: (i) *Local alignment*: CRIP uses RGB images unaligned with the RIR sensors, whereas AGREE uses pano depth captured at the receiver location. (ii) *Early reflections*: AGREE encodes local surfaces that govern early reflections (while also capturing global structure), whereas CRIP primarily correlates with late reverberation; ablating comparable geometry in FLAC significantly degrades early-reflection metrics (C50, EDT, see Tab. 1. (iii) *Evaluation role*: CRIP is an auxiliary training signal, whereas AGREE serves as a geometry-aware evaluation framework.

Table A.1. **Zero-shot retrieval on the unseen split of the AcousticRooms dataset using different geometry encoders:** We report both acoustic-to-geometry (A2G) and geometry-to-acoustic (G2A) results for several ViT variants. $\mathcal{W}_{\text{DINO}}$ denotes DINOv3 pre-trained weights. Models marked with † are trained on the full AcousticRooms dataset.

ViT	ϕ_G	$\mathcal{W}_{\text{DINO}}$	A2G R@1↑	A2G R@5↑	A2G R@10↑	G2A R@1↑	G2A R@5↑	G2A R@10↑
[6]	✗	✗	6.26	17.82	25.69	6.63	19.20	27.30
S/16 [3]	✗	✗	26.53	54.96	67.19	27.47	55.78	67.68
S/16 [14]	✗	✗	42.26	72.08	81.11	42.32	71.28	80.29
S/16 [14]	✗	❄	2.00	7.42	12.77	1.94	7.40	13.08
S/16 [14]	✗	✓	**59.78**	83.53	89.35	59.10	85.56	91.04
S+/16 [14]	✗	✓	58.17	**85.06**	**90.72**	**60.01**	**85.85**	**92.00**
S/16 [14]†	✗	✓	85.37	99.70	99.98	84.38	99.53	99.97
S+/16 [14]†	✗	✓	85.26	99.68	99.97	84.28	99.78	99.98

Table A.2. **Performance on seen AcousticRooms scenes:** Results are shown for $K \in \{8, 1, \text{✗}\}$ reference RIRs. For FLAC, we report mean and standard deviation over 5 generations. FLAC outperforms all baselines even in the one-shot setting. ✗ denotes ablations with either geometry (G) or audio conditioning removed.

Method	K	G	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓	R@1 (%) ↑	R@5 (%) ↑	R@10 (%) ↑	FD_G ↓
Random Across Rooms	✗	✗	44.02	6.415	274.43	0.02	0.10	0.21	0.061
Random Same Room	✗	✗	**13.47**	3.582	126.28	1.03	3.12	4.49	**0.006**
FLAC✗	✗	✓	$24.56_{\pm 0.03}$	$\mathbf{2.464}_{\pm 0.003}$	$\mathbf{114.84}_{\pm 0.11}$	$\mathbf{4.97}_{\pm 0.14}$	$\mathbf{15.63}_{\pm 0.17}$	$\mathbf{22.52}_{\pm 0.13}$	0.326
Nearest Neighbor	1	✗	11.91	3.406	120.72	0.00	6.79	9.96	**0.001**
FastRIR	1	✓	15.88	2.152	92.44	0.23	0.89	1.64	0.383
xRIR	1	✓	10.66	1.476	58.67	0.39	1.72	3.15	0.270
FLAC	1	✓	$\mathbf{6.46}_{\pm 0.07}$	$\mathbf{0.692}_{\pm 0.004}$	$\mathbf{28.11}_{\pm 0.08}$	$\mathbf{8.48}_{\pm 0.12}$	$\mathbf{21.54}_{\pm 0.20}$	$\mathbf{28.77}_{\pm 0.13}$	0.296
Linear Interpolation	8	✗	11.30	2.339	86.59	0.87	4.13	7.03	0.393
Nearest Neighbor	8	✗	8.31	1.802	64.77	0.08	**23.08**	**30.71**	**0.002**
FLAC✗	8	✗	$10.47_{\pm 0.01}$	$3.500_{\pm 0.001}$	$125.99_{\pm 0.07}$	$0.10_{\pm 0.01}$	$0.41_{\pm 0.04}$	$0.86_{\pm 0.04}$	0.644
FastRIR	8	✓	15.19	2.063	86.46	0.14	0.79	1.61	0.381
xRIR	8	✓	6.99	0.916	34.15	0.48	2.38	4.02	0.328
FLAC	8	✓	$\mathbf{5.32}_{\pm 0.01}$	$\mathbf{0.643}_{\pm 0.001}$	$\mathbf{25.69}_{\pm 0.05}$	$\mathbf{8.52}_{\pm 0.09}$	$21.89_{\pm 0.20}$	$29.28_{\pm 0.16}$	0.299

D. Evaluation details

D.1. Datasets

AcousticRooms. Each RIR in AcousticRooms is sampled at 22.05 kHz and truncated to 9,600 samples (≈0.435 s). We compute the magnitude spectrograms of the contextual RIRs before feeding them to the ResNet18 using an FFT size of 124, a window length of 62, and a hop size of 31. The panoramic depth maps are provided at a resolution of 256×512 and are projected into 3D point clouds using equirectangular projection.

The dataset includes randomized material assignments drawn from 332 materials across 11 categories, ensuring strong diversity in acoustic behavior even among similar geometries. As the simulation meshes are untextured, no RGB data are available.

HAA. RIRs in the HAA dataset are originally sampled at 48 kHz. We downsample them to 22.05 kHz using Librosa's `resample` and truncate them to 9,600 samples to match our setup. Contextual RIRs are transformed using the same FFT pipeline as for AcousticRooms. While the dataset does not provide depth maps, simplified surface annotations are available[3]. We use these to reconstruct a mesh with Open3D [21] and generate panoramic depth maps at each source position via Open3D raycasting. Note that, due to the simplified surface annotations, these depth maps differ substantially from those in AcousticRooms, widening the domain gap between the datasets. The test set comprises 1,282 instances (*Base* rooms). To compute metrics, we first average results within each room and then average across the four rooms, preventing rooms with more data from dominating the results. Following [6], we exclude the mean T60 for the dampened room, as all methods report unusually high values, likely due to its particular acoustic characteristics.

[3] https://github.com/maswang32/hearinganythinganywhere

D.2. Perceptual metrics

Following [6], we compute metrics on waveform of length 8,000 on the AcousticRooms dataset and 9,600 on the HAA dataset. We use T60, C50 and EDT errors obtained as follows, where $\hat{\mathbf{x}}$ is the synthesized RIR waveform:

$$\text{T60}(\hat{\mathbf{x}}, \mathbf{x}) = \frac{|\text{T60}(\hat{\mathbf{x}}) - \text{T60}(\mathbf{x})|}{\text{T60}(\mathbf{x})}, \tag{11}$$

$$\text{C50}(\hat{\mathbf{x}}, \mathbf{x}) = |\text{C50}(\hat{\mathbf{x}}) - \text{C50}(\mathbf{x})|, \tag{12}$$

$$\text{EDT}(\hat{\mathbf{x}}, \mathbf{x}) = |\text{EDT}(\hat{\mathbf{x}}) - \text{EDT}(\mathbf{x})|. \tag{13}$$

For the AcousticRooms dataset, T60 is estimated based on T20, fitting the decay between -5 dB and -25 dB and linearly extrapolating to 60 dB. For HAA, it is based on T30, following [6] implementation.

D.3. Scene-consistency metrics

To evaluate how well generated RIRs preserve geometry-consistent acoustic behavior, we use AGREE, a CLIP-style audio-geometry joint embedding network. Let ϕ_A and ϕ_G denote its acoustic and geometry encoders, mapping audio $\mathbf{x}$ and geometry g into a shared space. We propose metrics that both measure instance-level alignment (recall) and global distributional consistency (Fréchet Distance).

Audio-to-audio retrieval. For each generated RIR $\widehat{\mathbf{x}}_i$, we compute its normalized embedding $\widehat{\mathbf{A}}_i = \phi_A(\hat{\mathbf{x}}_i)$ and compare it to the normalized embeddings $\mathbf{A}_j = \phi_A(\mathbf{x}_j)$ of the ground-truth RIRs. Similarity is measured as:

$$s_{ij} = \widehat{\mathbf{A}}_i^\top \mathbf{A}_j. \tag{14}$$

The audio-to-audio recall at X (R@X) is then defined as:

$$\text{R@X} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left(i \in \text{TopX}_j(s_{ij})\right), \tag{15}$$

where $\text{TopX}_j(s_{ij})$ returns the indices of the X most similar ground-truth embeddings for each generated RIR. This metric measures how often the ground-truth RIR corresponding to a generated sample appears among the top-X most similar ground-truth embeddings.

Acoustic-to-geometry retrieval. We similarly evaluate the alignment between generated audio and its corresponding room geometry. For each generated RIR embedding $\phi_A(\hat{\mathbf{x}}_j)$, we compute cosine similarities with all ground-truth geometry embeddings $\phi_G(\mathbf{g}_i)$. The resulting recall at X (R@X) measures how often the correct geometry is retrieved among the top-X most similar embeddings.

Fréchet Distance in AGREE space. To assess distributional realism, we compute a Fréchet Distance (FD) between the generated and ground-truth RIR embeddings in the AGREE space. Let $\mathcal{N}(\mu_r, \Sigma_r)$ and $\mathcal{N}(\mu_g, \Sigma_g)$ denote multivariate Gaussian approximations of the ground-truth and generated audio embeddings, respectively:

$$\begin{aligned}\mu_r &= \frac{1}{N_r}\sum_i \phi_A(\mathbf{x}_i),\\ \Sigma_r &= \frac{1}{N_r - 1}\sum_i (\phi_A(\mathbf{x}_i) - \mu_r)(\phi_A(\mathbf{x}_i) - \mu_r)^\top.\end{aligned} \tag{16}$$

and similarly for μ_g, Σ_g. The Fréchet Distance is then defined as:

$$\text{FD}_G = \|\mu_r - \mu_g\|_2^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}\right). \tag{17}$$

Lower values indicate a closer match between the distributions of generated and real RIR embeddings, reflecting better geometry-consistent realism.

D.4. Baselines implementation details

Random across rooms. This baseline randomly selects one RIR from the entire dataset. In HAA, only training samples are considered.

Random same room. This baseline randomly selects one RIR from the same room. In HAA, only training samples from the same room are considered.

Linear interpolation. This baseline interpolates the K reference RIRs based on their distance to the target source. For each reference k, the distance r_k to the target source P_s^T is computed, and the weight is set as $w_k = 1/(r_k + \epsilon)$, normalized to sum to 1. The final RIR is the weighted sum of the K references.

Table A.3. **Effect of the geometry and acoustic conditioning encoders (seen):** We evaluate different configurations of the geometry encoder ϕ_G and acoustic encoder ϕ_A on the seen set of the AcousticRooms dataset. For ϕ_G, we compare the ViT from xRIR and DINOv3 ViT-S/16 under various initialization strategies ($\mathcal{W}_{\text{DINO}}$): trained from scratch, frozen, or initialized with DINO weights. For ϕ_A, we experiment with the ResNet-18 used in xRIR and our frozen VAE encoder.

ViT ϕ_G	$\mathcal{W}_{\text{DINO}}$	ϕ_A	K	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓	R@5 (%) ↑	FD_G ↓
[6]	✗	ResNet	1	6.77	0.741	30.15	12.9	0.319
S/16 [14]	✗	ResNet	1	6.83	0.740	30.23	17.65	0.310
S/16 [14]	❄	ResNet	1	7.61	0.956	36.98	6.40	0.365
S/16 [14]	✓	ResNet	1	**6.46**	**0.692**	28.11	**21.54**	**0.296**
S/16 [14]	✓	VAE	1	6.81	**0.692**	**27.97**	20.28	0.309
[6]	✗	ResNet	8	5.42	0.674	27.48	14.51	0.321
S/16 [14]	✗	ResNet	8	5.60	0.685	27.51	19.24	0.311
S/16 [14]	❄	ResNet	8	6.30	0.858	32.42	7.62	0.374
S/16 [14]	✓	ResNet	8	**5.32**	0.643	25.69	**21.89**	**0.299**
S/16 [14]	✓	VAE	8	5.74	**0.619**	**24.42**	21.12	0.308

Table A.4. **Effect of DiT architecture variants (seen):** Performance comparison of different conditioning strategies on the seen set of the AcousticRooms dataset. We report results for In-Context, Cross-Attention (CA), and our hybrid design combining AdaLN for target information and CA for contextual cues.

Method	K	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓	R@5 (%) ↑	FD_G ↓
In-Context	1	72.37	11.299	1318.71	0.05	1.302
CA	1	14.44	1.411	75.13	7.43	0.406
AdaLN+CA	1	**6.46**	**0.692**	**28.11**	**21.54**	**0.296**
In-Context	8	**5.31**	0.651	26.58	21.10	0.307
CA	8	6.01	0.782	30.16	14.40	0.328
AdaLN+CA	8	5.32	**0.643**	**25.69**	**21.89**	**0.299**

Nearest neighbor (KNN). From the K reference RIRs, this baseline returns the RIR whose source position is closest (in Euclidean distance) to the target source position.

Fast-RIR. This baseline uses the authors' original implementation. To align with our setup, we infer the room size from the panoramic depth map captured at the receiver pose and estimate T_{60} using the K contextual RIRs. Specifically, we compute T_{60} for each of the K RIRs and average the results. Following prior work [8], we incorporate an energy decay loss to improve performance.

xRIR. We use the implementation provided by the authors. Following their supplementary material, the Vision Transformer encoder uses 6 multi-head attention layers (8 heads, hidden size 512) with a patch size of 16×32. Poses are encoded using sinusoidal positional embeddings applied to each 3D coordinate with 20 frequency bins. We set $\lambda = 0.01$ to balance the STFT and energy-decay losses. We train xRIR on the AcousticRooms training split, excluding both the seen and unseen test sets, and use the same trained model for evaluating both splits. For HAA, we fine-tune the AcousticRooms-pretrained model on the four HAA rooms at the same time.

Table A.5. **Acoustic-to-geometry retrieval on the AcousticRooms dataset:** Results are shown for $K \in \{8, 1, ✗\}$ reference RIRs. For FLAC, we report mean and standard deviation over 5 generations. ✗ denotes ablations with either geometry (G) or audio conditioning removed. FLAC achieves higher acoustic-to-geometry recall than the baselines, demonstrating higher scene consistency.

Method	K	G	Unseen			Seen		
			R@1 ↑	R@5 ↑	R@10 ↑	R@1 ↑	R@5 ↑	R@10 ↑
Random Across Rooms	✗	✗	0.03	0.13	0.21	0.00	0.05	0.16
Random Same Room	✗	✗	0.22	0.98	1.78	0.88	2.45	3.78
FLAC✗	✗	✓	**5.12**$_{\pm 0.15}$	**16.10**$_{\pm 0.19}$	**24.01**$_{\pm 0.12}$	**5.58**$_{\pm 0.10}$	**16.18**$_{\pm 0.12}$	**23.09**$_{\pm 0.20}$
Nearest Neighbor	1	✗	0.05	2.04	4.09	0.21	5.65	8.41
Fast-RIR	1	✓	0.16	0.79	1.52	0.15	0.82	1.61
xRIR	1	✓	0.19	1.22	2.38	0.16	1.27	2.54
FLAC	1	✓	**5.60**$_{\pm 0.06}$	**16.54**$_{\pm 0.24}$	**24.32**$_{\pm 0.37}$	**6.37**$_{\pm 0.08}$	**17.71**$_{\pm 0.21}$	**25.05**$_{\pm 0.10}$
Linear Interpolation	8	✗	0.28	1.40	2.87	0.64	2.85	5.10
Nearest Neighbor	8	✗	0.16	8.66	15.20	0.68	**19.21**	**25.98**
FLAC✗	8	✗	0.15$_{\pm 0.01}$	0.45$_{\pm 0.03}$	0.84$_{\pm 0.07}$	0.13$_{\pm 0.01}$	0.43$_{\pm 0.03}$	0.77$_{\pm 0.03}$
Fast-RIR	8	✓	0.17	1.09	1.82	0.16	0.77	1.61
xRIR	8	✓	0.43	1.85	3.01	0.37	1.56	2.93
FLAC	8	✓	**5.79**$_{\pm 0.07}$	**16.84**$_{\pm 0.08}$	**24.62**$_{\pm 0.17}$	**6.53**$_{\pm 0.14}$	18.10$_{\pm 0.15}$	25.58$_{\pm 0.15}$

INRAS. Similar to [1], we modify the original bounce point sampling strategy, which only sampled points at a fixed height. Instead, we construct scene meshes from the HAA surface annotations and apply Poisson sampling to obtain 256 3D bounce points, providing a richer representation of the scene geometry. As the training set contains only 12 RIRs per room, we train per scene with a batch size of 12 for 5k epochs. We found that adjusting the multi-resolution STFT loss parameters improves performance, using FFT sizes $\{128, 512, 1024, 2048\}$, hop lengths $\{16, 50, 120, 240\}$, and window lengths $\{80, 240, 600, 1200\}$.

DiffRIR. We use the official implementation and adapt it to 22,050 Hz. Specifically, we reduce the maximum predicted audio length from 96,000 samples at 48 kHz to 44,100 samples at 22.05 kHz. We use the authors' precomputed sound trajectories and rescale the delay values to match the new sample rate.

E. Additional results

E.1. Seen set of the AcousticRooms dataset

Comparison to the baselines. Tab. A.2 reports results on the seen set of the AcousticRooms dataset, which contains novel source-receiver positions within scenes observed during training. With K=8, FLAC reduces errors by 23.9%, 29.8%, and 24.8% on T60, C50, and EDT, respectively, compared to xRIR. It also substantially improves scene-consistency metrics and slightly improves FD_G (-8.8%) over xRIR. Consistent with the unseen-set results, FLAC with K=1 outperforms all other one-shot methods and even surpasses xRIR with K=8.

Ablation of conditioning modalities. Tab. A.2 reports FLAC variants with one conditioning modality removed (✗). The trends mirror those observed on the unseen set. Removing geometry leads to large drops in geometry-related metrics (R@1-10, FD_G) even with K=8 contextual RIRs. Conversely, using geometry without contextual RIRs still leads to satisfactory performance. C50 and EDT are better with geometry-only than with context-only conditioning, consistent with their dependence on early reflections, which are closely tied to local scene geometry. T60 is better with audio-only conditioning, aligning with its dependence on global room characteristics that are difficult to infer from local geometry alone.

Ablation on geometry and acoustic encoders. Tab. A.3 compares FLAC with different geometry and acoustic encoders on AcousticRooms'seen set. Consistent with the unseen-set results, fine-tuning the DINOv3 ViT S/16 achieves the best overall performance. Using our frozen VAE as the acoustic encoder improves C50 and EDT but slightly degrades other metrics, further motivating the use of the simpler and more efficient ResNet-18 for acoustic conditioning.

Influence of the DiT architecture. Consistent with results on the unseen set of the AcousticRooms dataset, the

Table A.6. **Sim-to-real transfer on the Hearing-Anything-Anywhere dataset with inverted receiver pose:** Few-shot methods are compared to Diff-RIR and INRAS, which require per-scene training (†). For FLAC, we report the mean and standard deviation over 5 generations. Inverting the receiver pose improves T60/C50/EDT but reduces geometry-consistency metrics.

Method	K	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓	R@5 (%) ↑	FD_G ↓
Random Across Rooms	✗	17.40	10.283	533.99	2.02	0.418
Random Same Room	✗	8.00	4.805	180.15	1.51	0.253
Nearest Neighbor	1	8.19	5.000	187.55	1.32	**0.268**
xRIR	1	8.63	4.862	183.27	8.43	0.372
FLAC	1	**3.02**$_{\pm 0.04}$	**1.676**$_{\pm 0.023}$	**75.14**$_{\pm 0.98}$	**10.31**$_{\pm 1.10}$	0.899
Linear Interpolation	8	4.12	2.695	88.19	8.24	0.824
Nearest Neighbor	8	2.89	1.923	77.24	**12.64**	**0.243**
xRIR	8	6.53	3.492	149.69	10.93	0.383
FLAC	8	**2.87**$_{\pm 0.04}$	**1.595**$_{\pm 0.017}$	**69.14**$_{\pm 0.56}$	10.31$_{\pm 0.75}$	0.936
INRAS†	12	6.61	3.967	158.07	2.79	1.070
Diff-RIR†	12	3.74	2.067	88.09	18.03	0.539

combination of AdaLN for target-related conditioning and cross-attention for context leads to the best performance (see Tab. A.4).

E.2. Reversed setup in HAA

The HAA setup is reversed compared to AR. In HAA, context references share the same source, whereas in AR they share the same receiver. Consequently, the panoramic depth is captured at the source pose in HAA and at the receiver pose in AR. In the main paper, we fine-tune FLAC on HAA by simply swapping source and receiver poses (same for AGREE). We also evaluated inverting the receiver projection. This modification significantly improves T60/C50/EDT metrics but reduces scene consistency metrics. In this setting, AGREE used for evaluation is fine-tuned with the same modification. Results are reported in Tab. A.6

E.3. Additional metrics

Acoustic-to-geometry retrieval. To complement the audio-to-audio retrieval metrics presented in the main document, we provide acoustic-to-geometry retrieval metrics in Tab. A.5 for both the seen and unseen sets of the AcousticRooms dataset. FLAC produces more geometry-consistent RIRs as demonstrated by its superior recall performances.

MAG and ENV on HAA. We report MAG and ENV metrics on the HAA dataset in Tab. A.7. Following [18], MAG denotes the multiscale log-spectral L1 distance, which compares generated and ground-truth waveforms in the time-frequency domain across multiple resolutions. ENV is the envelope distance, defined as the L1 distance between the log-energy envelopes of the generated and ground-truth waveforms. The energy decay envelope captures the decay characteristics of a RIR, reflecting the reverberant properties of a room. For both metrics, we follow diffRIR implementation. Consistent with other metrics, FLAC outperforms xRIR at $K{=}8$, and surpasses others at $K{=}1$.

Table A.7. **Additional metrics on the Hearing-Anything-Anywhere dataset:** Few-shot methods are compared against Diff-RIR, which requires per-scene training (†) using MAG and ENV error.

Method	K	MAG ↓	ENV ↓
Random Across Rooms	✗	6.97	1.581
Random Same Room	✗	3.22	0.470
Nearest Neighbor	1	3.24	0.475
xRIR	1	3.97	0.577
FLAC	1	**3.16**	**0.393**
Linear Interpolation	8	3.63	0.835
Nearest Neighbor	8	**2.88**	**0.373**
xRIR	8	3.44	0.407
FLAC	8	3.11	0.381
INRAS†	12	3.30	0.654
Diff-RIR†	12	2.53	0.352

E.4. Timestep sampling strategy

In Tab. A.8, we compare the effect of different training timestep sampling strategies on performance: (i) *LogSNR*, our baseline, which emphasizes higher t values; (ii) *Uniform*, sampling $t \sim \mathcal{U}(0,1)$; (iii) *Ones*, fixing $t = 1$ (fully noisy); and (iv) *Logit-Normal*, where $t = \sigma(\alpha)$ with $\alpha \sim \mathcal{N}(0,1)$, concentrating t in the mid-range.

Table A.8. **Comparison of timestep sampling strategies during training on the generation performance:** *LogSNR* emphasizes high t values, *Logit-Normal* concentrates on intermediate values, *Ones* fixes $t = 1$ (*i.e.*, full noise), and *Uniform* samples t uniformly. *LogSNR* provides the best overall trade-off between seen and unseen performance.

Timestep Sampler	Unseen			Seen		
	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓	T60 (%) ↓	C50 (dB) ↓	EDT (ms) ↓
Uniform	8.97	**0.943**	38.18	**5.22**	**0.636**	25.83
Ones	**8.17**	1.044	37.48	5.29	**0.636**	**25.60**
Logit-Normal	9.77	1.068	42.65	5.80	0.681	27.98
LogSNR	8.60	0.970	**37.13**	5.32	0.643	25.69

F. Qualitative results

t-SNE. Fig. A.3 provides a t-SNE visualization of multiple generated samples across unseen rooms. Samples with the same conditioning cluster tightly, while those from different rooms or conditionings are clearly separated, further demonstrating that the model captures both consistency and diversity in its generations. We observe that acoustically similar scenes lies next to the other, *e.g.*, Restaurants-Cafes, Auditoriums-Listening Rooms, and Apartments (which include bathrooms)-standalone Bathrooms.

Octave-band analysis. We provide more visualization of octave-band analysis in Fig. A.5, Fig. A.6.

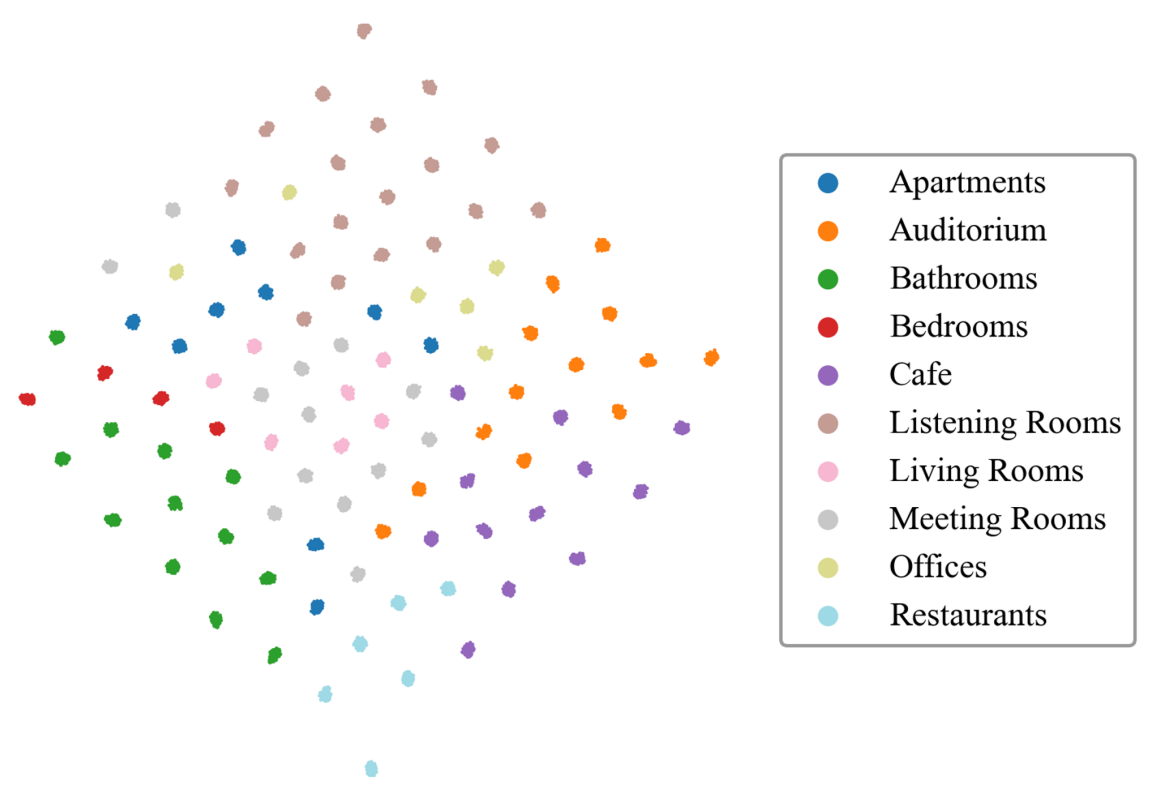

Figure A.3. **t-SNE visualization of generated RIRs across unseen rooms:** Each color represents a different room category. Samples cluster tightly within the same conditioning and remains well separated across conditionings and rooms, confirming consistent yet diverse RIR generation.

Waveforms. We present in Fig. A.7 and Fig. A.8 generated waveforms with xRIR and FLAC against the ground truth for different scenes of the AcousticRooms and HAA datasets. The rank metric corresponds to the audio-to-audio retrieval rank of each prediction against all the ground truth RIRs of the evaluation set.

Video. We provide in the project page several examples of audio generated with FLAC using $K{=}8$ and $K{=}1$ reference RIRs in unseen scenes from both simulated and real environments. The anechoic speech samples used for auralization come from the EARS dataset [12], and the music samples are taken from AVAD-VR [17]. For audio rendering along trajectories in real scenes, we convolve our predicted single-channel RIR at each point of the trajectory with a head-related impulse response derived from a predefined head-related transfer function. This produces binaural RIRs, which are then convolved with the source audio to obtain the final binaural rendering. The rank metric shown in the video corresponds to the audio-to-audio retrieval rank of each prediction against all ground-truth RIRs in the full unseen set, *i.e.*, the position of the correct ground-truth RIR in the similarity matrix.

G. Perceptual evaluation

We conducted a listening study with 46 participants on 14 unseen AR scenes. Participants were presented with the ground-truth (GT), audio generated by FLAC (1-shot) and xRIR (8-shot), and were asked to select which audio sounded closer to the GT. FLAC was preferred in 93.01% of cases. The order of questions and the assignment of methods to "algorithm A" and "algorithm B" were randomized. Participants were first shown a top view of the scene including the microphone and source positions. They then listened to the GT audio ("true"), obtained by convolving the ground-truth RIR with an anechoic signal. Next, they listened to the two generated samples ("algorithm A" and "algorithm B"), which were produced by convolving the same anechoic signal with RIRs generated by FLAC or xRIR. Fig. A.4 shows an example of the user interface used.

For the audio content, we used anechoic speech from the EARS dataset [12]. Different voices were used across questions, and some questions included music excerpts from AVAD-VR [17] to evaluate additional use cases. To ensure reliable listening conditions, participants were required to complete the survey on a laptop or desktop computer while

wearing headphones. They were also asked to report their background noise environment: 34 participants reported a quiet environment and 12 reported some background noise (possibilities were "quiet", "some background noise", "very noisy").

A total of 49 participants initially completed the survey. To screen for potential hearing issues or non-compliance with headphone use, we asked participants whether they had any known hearing impairments and whether they were wearing headphones. Additionally, the first question was a control trial in which participants had to choose the closer audio to the GT audio between the same GT audio and an audio recorded in a completely different scene. Three participants failed this control question and were excluded from the analysis, leaving 46 valid participants. Errors on this control question were correlated with participants reporting hearing impairments.

We also collected demographic information. Among the 46 participants, 13 identified as female and 33 as male. The age distribution was: 3 under 18, 6 between 18-24, 17 between 25-34, 5 between 35-44, 9 between 45-54, 4 between 55-64, and 2 aged 65 or older.

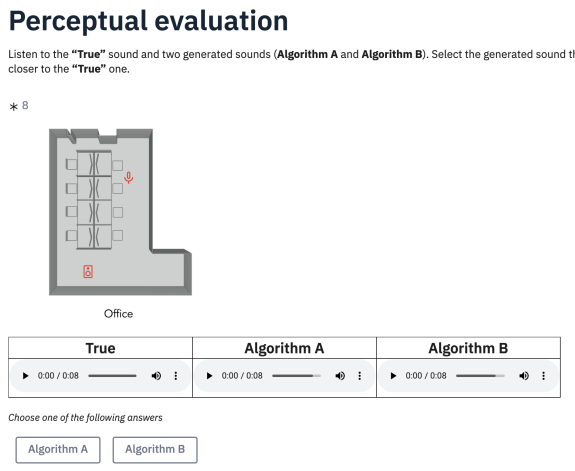

Figure A.4. **User study interface.** We created 2 synthetic audio using FLAC one-shot and xRIR one-shot and ask people which one matches closely with the ground truth.

H. Number of parameters and inference speed

We report in Tab. A.9 the trainable and total inference parameter counts of our models, xRIR, Fast-RIR, and INRAS, along with inference times measured on a single RTX 4090 GPU. Note that INRAS must be trained for each scene, this results in 240M parameters for the full Acoustic Rooms dataset. Results are shown for variants with the frozen VAE encoder (replacing the jointly trained ResNet-18), with xRIR's ViT architecture instead of DINOV3 ViT-S/16, with a DiT depth of 4 instead of 12 (FLAC-S), and for the VAE used to obtain the latent $\mathbf{z}_0$. Our models achieve real-time performance, as inference is faster than the duration of the generated audio. We also compare the performance of FLAC-S against FLAC and xRIR in Tab. A.10. FLAC-S (39M), obtained by reducing the DiT depth from 12 to 4, has a similar number of parameters to xRIR (32M). Despite this reduction, FLAC-S maintains performance comparable to FLAC, outperforming xRIR. This indicates that the gains of our method are not driven by model size.

Table A.9. **Number of trainable parameters along with inference parameters and speed** for our model, its variants, and state-of-the-art methods

. M denotes million.

Model	Trainable Param. (M)	Inference Param. (M)	Speed (ms)
INRAS [16]	$1\times N_{\text{scene}}$	$1\times N_{\text{scene}}$	1.9
Fast-RIR [10]	116	116	0.6
xRIR [6]	32.1	32.1	38.5
VAE	14.6	14.2	8.9
FLAC VAE	38.6	45.7	13.5
FLAC ViT [6]	48.5	55.6	13.7
FLAC-S	37.2	44.3	7.0
FLAC	50.3	57.4	13.5

Table A.10. **Comparison between FLAC-S, FLAC, and xRIR on the unseen AcousticRooms set**: FLAC-S reduces the parameter count by 13M compared to FLAC. Despite having a similar number of parameters to xRIR, it achieves performance comparable to FLAC, outperforming xRIR.

Method	**K**	**T60 (%)** ↓	**C50 (dB)** ↓	**EDT (ms)** ↓	**R@5 (%)** ↑	**FD$_G$** ↓
xRIR	1	14.47	1.961	74.45	1.36	0.263
FLAC-S	1	**9.71**$_{\pm 0.04}$	**1.010**$_{\pm 0.002}$	**39.58**$_{\pm 0.18}$	**17.80**$_{\pm 0.11}$	**0.264**
FLAC	1	9.95$_{\pm 0.05}$	1.046$_{\pm 0.002}$	40.04$_{\pm 0.22}$	18.92$_{\pm 0.10}$	0.303
xRIR	8	9.98	1.354	49.40	2.00	0.307
FLAC-S	8	8.69$_{\pm 0.02}$	**0.942**$_{\pm 0.001}$	37.14$_{\pm 0.08}$	17.88$_{\pm 0.17}$	**0.267**
FLAC	8	**8.60**$_{\pm 0.01}$	0.970$_{\pm 0.002}$	**37.13**$_{\pm 0.02}$	**19.38**$_{\pm 0.15}$	0.305

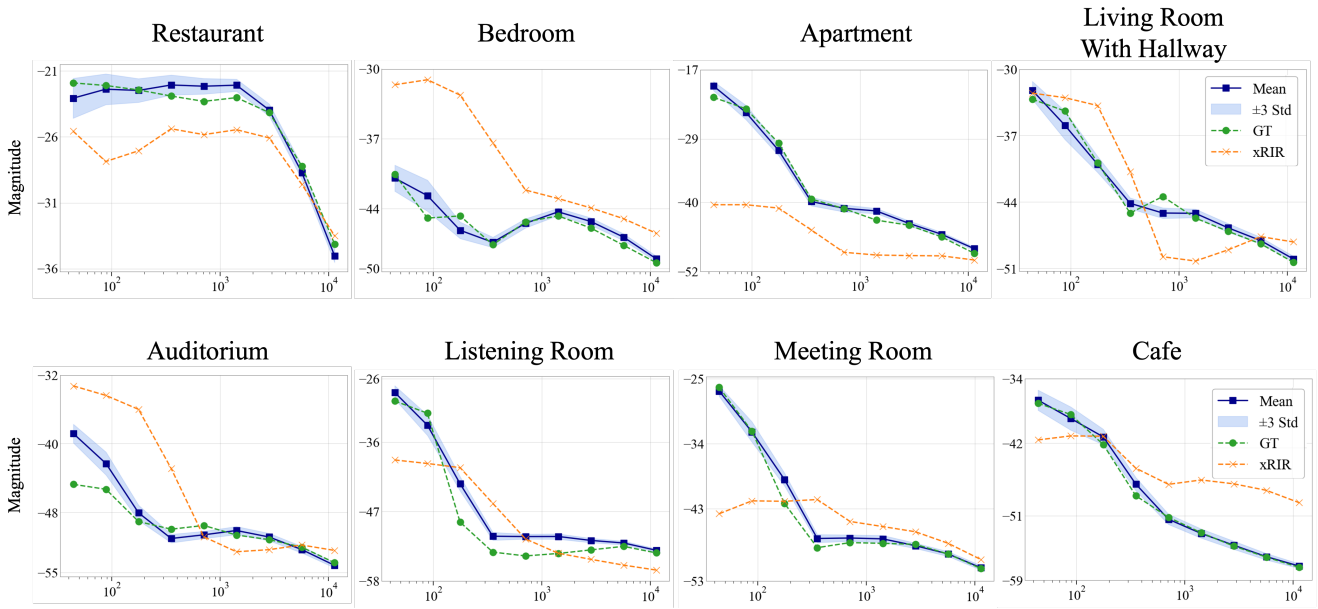

Figure A.5. **Additional octave band analysis on the AcousticRooms dataset:** We generate 100 samples per instance in the unseen set with FLAC under identical conditioning, and plot the mean along with a ± 3 standard deviation interval (covering 99.7% of the distribution). Ground truth and xRIR predictions are shown for comparison.

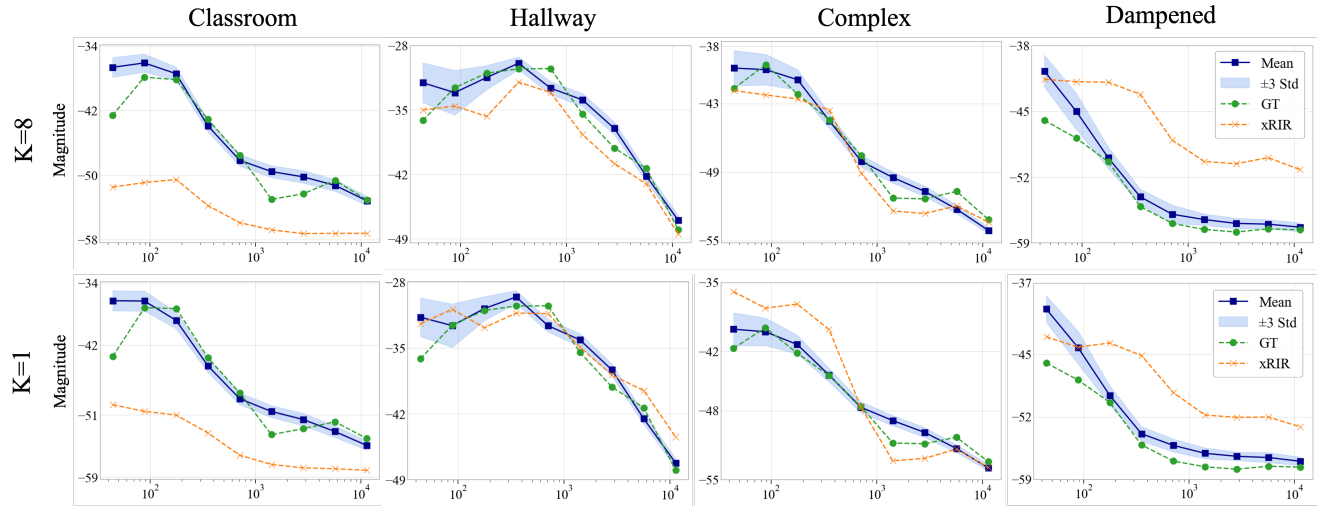

Figure A.6. **Octave band analysis on the HAA dataset.**

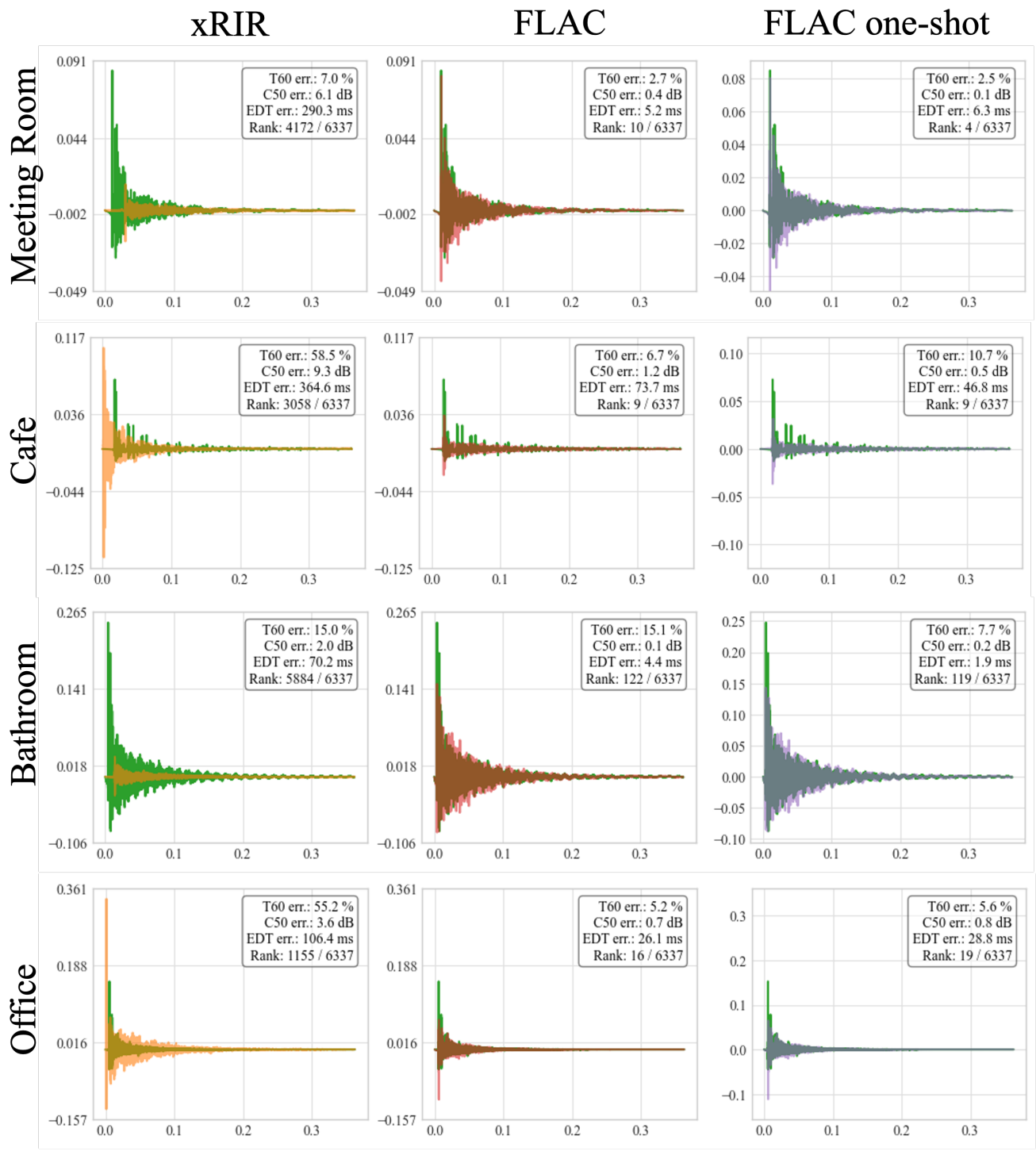

Figure A.7. **Qualitative comparison of predicted RIR waveforms on the unseen set of the AcousticRooms dataset:** We compare xRIR (orange), FLAC (red), and FLAC with $K = 1$ (purple) against the ground truth (green).

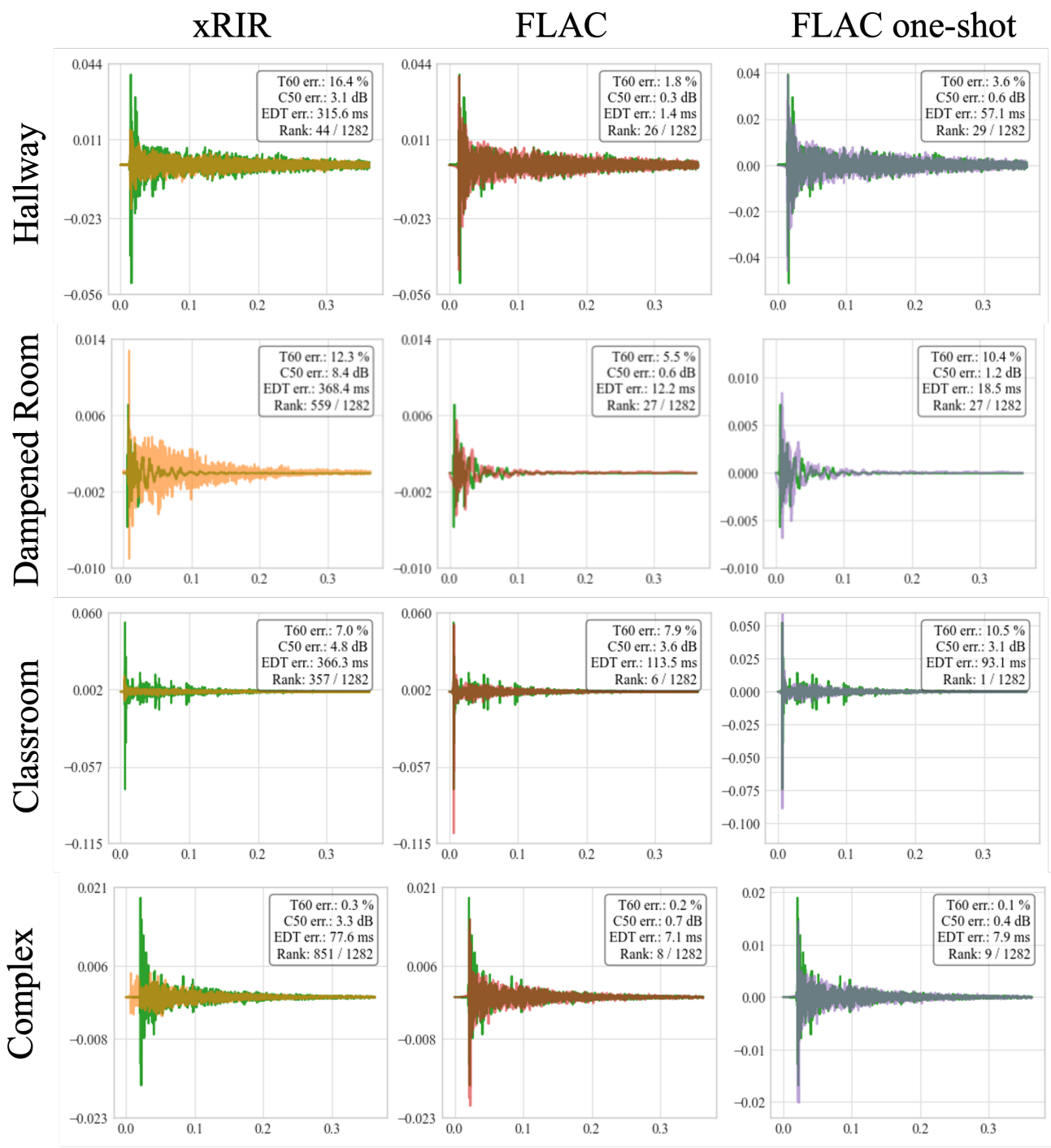

Figure A.8. **Qualitative comparison of predicted RIR waveforms on the HAA dataset:** We compare xRIR (orange), FLAC (red), and FLAC with $K = 1$ (purple) against the ground truth (green).

References

[1] Ziyang Chen, Israel D. Gebru, Christian Richardt, Anurag Kumar, William Laney, Andrew Owens, and Alexander Richard. Real acoustic fields: An audio-visual room acoustics dataset and benchmark. In *CVPR*, 2024. 6

[2] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022. 1

[3] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 2, 3

[4] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In *NeurIPS*, 2020. 1

[5] Kundan Kumar, Rithesh Kumar, Thibault De Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre De Brebisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. In *NeurIPS*, 2019. 1

[6] Xiulong Liu, Anurag Kumar, Paul Calamia, Sebastià V. Amengual Garí, Calvin Murdock, Ishwarya Ananthabhotla, Philip Robinson, Eli Shlizerman, Vamsi Krishna Ithapu, and Ruohan Gao. Hearing anywhere in any environment. In *CVPR*, 2025. 2, 3, 4, 5, 9

[7] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1, 2

[8] Sagnik Majumder, Changan Chen, Ziad Al-Halah, and Kristen Grauman. Few-shot audio-visual learning of environment acoustics. *NeurIPS*, 2022. 5

[9] 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 *ICML*, 2021. 2

[10] Anton Ratnarajah, Shi-Xiong Zhang, Meng Yu, Zhenyu Tang, Dinesh Manocha, and Dong Yu. Fast-rir: Fast neural diffuse room impulse response generator. In *ICASSP*, 2022. 9

[11] Anton Ratnarajah, Sreyan Ghosh, Sonal Kumar, Purva Chiniya, and Dinesh Manocha. Av-rir: Audio-visual room impulse response estimation. In *CVPR*, 2024. 2, 3

[12] Julius Richter, Yi-Chiao Wu, Steven Krenn, Simon Welker, Bunlong Lay, Shinjii Watanabe, Alexander Richard, and Timo Gerkmann. EARS: An anechoic fullband speech dataset benchmarked for speech enhancement and dereverberation. In *Interspeech*, 2024. 8

[13] Mahnoor Fatima Saad and Ziad Al-Halah. How would it sound? material-controlled multimodal acoustic profile generation for indoor scenes. In *ICCV*, 2025. 2

[14] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. 2, 3, 5

[15] Christian J Steinmetz and Joshua D Reiss. auraloss: Audio focused loss functions in pytorch. In *Digital music research network one-day workshop*, 2020. 1

[16] Kun Su, Mingfei Chen, and Eli Shlizerman. Inras: Implicit neural representation for audio scenes. In *NeurIPS*, 2022. 9

[17] David Thery and Brian FG Katz. Anechoic audio and 3d-video content database of small ensemble performances for virtual concerts. In *ICA*, 2019. 8

[18] Mason Long Wang, Ryosuke Sawata, Samuel Clarke, Ruohan Gao, Shangzhe Wu, and Jiajun Wu. Hearing anything anywhere. In *CVPR*, 2024. 2, 7

[19] Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim. Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In *ICASSP*, 2020. 1

[20] Jaeseong You, Dalhyun Kim, Gyuhyeon Nam, Geumbyeol Hwang, and Gyeongsu Chae. Gan vocoder: Multi-resolution discriminator is all you need. *arXiv preprint arXiv:2103.05236*, 2021. 1

[21] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018. 4