

EchoMark: Perceptual Acoustic Environment Transfer with Watermark-Embedded Room Impulse Response

Chenpei Huang¹, Lingfeng Yao¹, Kyu In Lee¹, Lan Zhang², Xun Chen³, Miao Pan¹

¹ University of Houston ² Clemson University ³ Independent Researcher
{chuang30, lyao24, klee48, mpan2}@uh.edu, lan7@clemson.edu, xunchen@outlook.com

Abstract

Acoustic Environment Matching (AEM) is the task of transferring clean audio into a target acoustic environment, enabling engaging applications such as audio dubbing and auditory immersive virtual reality (VR). Recovering similar room impulse response (RIR) directly from reverberant speech offers more accessible and flexible AEM solution. However, this capability also introduces vulnerabilities of arbitrary “relocation” if misused by malicious user, such as facilitating advanced voice spoofing attacks or undermining the authenticity of recorded evidence. To address this issue, we propose EchoMark, the first deep learning-based AEM framework that generates perceptually similar RIRs with embedded watermark. Our design tackle the challenges posed by variable RIR characteristics, such as different durations and energy decays, by operating in the latent domain. By jointly optimizing the model with a perceptual loss for RIR reconstruction and a loss for watermark detection, EchoMark achieves both high-quality environment transfer and reliable watermark recovery. Experiments on diverse datasets validate that EchoMark achieves room acoustic parameter matching performance comparable to FiNS, the state-of-the-art RIR estimator. Furthermore, a high Mean Opinion Score (MOS) of 4.22 out of 5, watermark detection accuracy exceeding 99%, and bit error rates (BER) below 0.3% collectively demonstrate the effectiveness of EchoMark in preserving perceptual quality while ensuring reliable watermark embedding.

Introduction

Audio recordings are a natural medium through which people perceive, understand, and enjoy speech, music, and podcasts. To meet listeners’ expectations, audio technology continually pushes toward higher fidelity and greater immersive experience. Achieving a near-realistic listening experience requires capturing not only the primary speech content but also contextual cues, such as surrounding acoustic environment. Remarkably, even casual listeners can often distinguish between different recording settings. For instance, speech recorded in a large meeting room sounds noticeably different from that captured in a small office. These perceptual differences arise from the physical characteristics of a space, such as size, shape and surface materials, which affect how sound propagates and reflects. As a result, environmental acoustic cues enrich auditory realism and support the authenticity of accompanying visual or spoken content.

Central to these acoustic cues is the room impulse response (RIR), which characterizes how a space shapes sound. Environments with similar RIRs are often perceived as the same space, while mismatched RIRs reveal inconsistencies. Acoustic environment matching (AEM) adapts recorded or synthesized speech to a specific target environment by convolving the clean source signal with an RIR resembling that of the target space. However, directly measuring an RIR in situ can be difficult or impractical in many real-world scenarios, such as for post-production or real-time applications (Ratnarajah et al. 2023). Instead, extracting an RIR from a reverberant recording offers a more flexible and scalable alternative. This enables applications such as film dubbing, where studio-recorded speech is matched to on-screen environments, or immersive voice rendering in games and virtual reality.

Yet as these techniques become more accessible and realistic, they also introduce new vulnerabilities. The ability to alter the acoustic environment of an audio signal without leaving a detectable trace undermines the credibility of RIR-based cues. When synthetic reverberation becomes indistinguishable from natural room acoustics, traditional methods for environment identification and forensic analysis may be rendered ineffective. This opens the door to misuse in scenarios such as voice spoofing, misinformation, or evidence tampering. For example, an attacker may transfer spoofed speech into a familiar environment to deceive a victim into believing the message is authentic. Another potential misuse involves fabricating one’s presence in a certain location by simulating its room acoustic characteristics. In particular, techniques that rely on RIR-based environment identification (Zhao and Malik 2013; Jia et al. 2015; Guo et al. 2021) may fail under adversarial environment transfer. These risks underscore the need for new mechanisms that preserve the immersive experience, i.e., AEM quality, while enabling reliable verification and misuse detection.

In the context of AEM, watermarking the target RIR is particularly desirable, as it is convolved with the entire speech regardless of its content. This allows the service provider to detect potential misuse of their acoustic environment matching systems from RIR, regardless of the source speech content. Moreover, to mitigate legal and ethical concerns regarding unauthorized usage, watermarking offers a proactive layer of accountability. Similar to prior

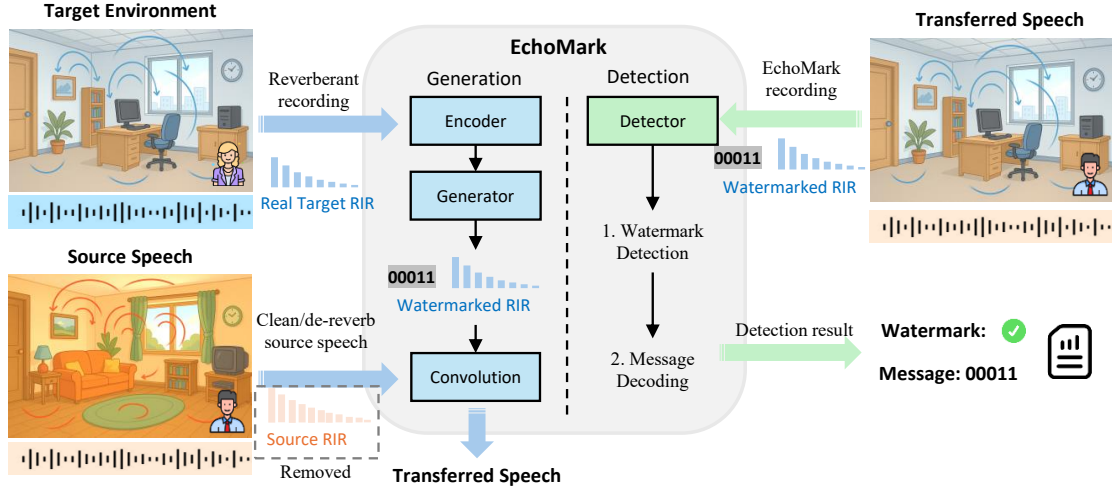


Figure 1: EchoMark system overview in perceptual environment transfer using watermarked RIR.

work (San Roman et al. 2024), the embedded watermark can also encode multiple information bits, such as ownership identifiers or usage metadata.

Nevertheless, we identify three key challenges in extracting and watermarking the RIR from the reverberant speech in target environment: ① **RIR characteristics**. Unlike speech signals, which typically exhibit relatively stable spectral energy distribution over time, RIRs are highly structured and non-stationary. They begin with an impulse-like direct path and early reflections, followed by an exponentially decaying late reverberation. Existing audio watermarking methods are not designed for such signals. ② **RIR variability**. The duration and structure of RIRs vary widely, ranging from a few hundred milliseconds to several seconds depending on the room size and acoustic conditions. This variability poses challenges for designing a robust and generalizable watermarking scheme. ③ **Post-convolution detection**. The watermarked RIR is not directly observable, as it is convolved with source speech during environment transfer. As a result, the watermark must remain detectable from the environment-transferred reverberant audio, regardless of the speech content. Ensuring watermark robustness under diverse speech inputs is essential for reliable detection.

We propose EchoMark, a deep learning framework that generates watermarked RIRs from target reverberant speech for acoustic environment matching. To address the aforementioned challenges, we embed watermark information in the latent space, which explicitly controls the generation of early and late components of the RIR waveform. This design enables the model to learn robust watermark embedding and decoding under variations in both RIR characteristics and source speech content. Specifically, we use a Conformer-based (Gulati et al. 2020) **RIR encoder** to extract RIR-related cues into a fixed-length latent embedding. This embedding captures essential RIR characteristics and remains stable under data augmentation, such as noise injection and different speech contents. Watermark information is hereby

embedded into this latent space to bind with the RIR representation. The **RIR generator** reconstructs the RIR waveform from the watermarked embedding, producing perceptually similar RIRs for environment transfer, i.e., convolution with clean source speech. Finally, a separate **watermark decoder**, using the same architecture as the encoder (different weights), recovers the watermark from the resulting audio. The system is jointly trained with loss functions that balance the two purposes.

Our main contributions are as follows:

- To the best of our knowledge, EchoMark is the first watermarking framework designed for transferring acoustic environments. It enables seamless environment transfer while reducing the risk of misuse for service providers.
- We design a latent-space watermarking mechanism that explicitly controls early and late RIR generation, allowing the watermark to be embedded in perceptually insensitive regions while remaining robust across speech content and SNR variations.
- Experimental results demonstrate that EchoMark achieves high perceptual similarity to target environments, as reflected in room acoustic metrics and listener mean opinion scores. It also achieves high watermark detection accuracy and low bit error rates in message decoding.

Background

Acoustic Environment Matching. Acoustic environment matching is essential for generating perceptually realistic audio, especially in applications such as auditory augmented reality (AAR), film production, and virtual environments. The goal of AEM is to integrate source speech into a target environment such that the resulting audio is perceptually indistinguishable from recordings made in that space. As shown in (Neidhardt, Schneiderwind, and Klein 2022), perceptual similarity is strongly influenced by room acoustic parameters such as reverberation time (RT) and direct-

to-reverberant ratio (DRR), with room impulse responses playing a central role in capturing these characteristics. Recent AEM methods generally fall into two categories: waveform generation conditioned on RIR embeddings, and explicit RIR estimation followed by dereverberation. For example, Su et al. (Su, Jin, and Finkelstein 2020) and Koo et al. (Koo, Paik, and Lee 2021) extract RIR embeddings from the target environment and inject them into neural speech generators, enabling the substitution of source reverberation with that of the target environment.

Blind and Informed RIR Estimation. Recent studies have explored estimating room impulse responses (RIRs) directly from reverberant speech as an alternative to using pre-recorded impulse responses. This enables acoustic environment matching without the need for explicit RIR measurement. Blind RIR estimation methods rely solely on audio and often model dereverberation and RIR recovery jointly (Wager, Choi, and Durand 2020; Bahrman et al. 2024). Traditional signal processing techniques such as weighted prediction error (WPE) (Nakatani et al. 2010) and non-negative matrix factorization (NMF) (Mohammadiha, Smaragdis, and Doclo 2015) have been largely replaced by neural approaches that estimate spectrogram-domain filters. IR-GAN (Ratnarajah, Tang, and Manocha 2020) and Speech2IR (Ratnarajah et al. 2023) use generative networks to synthesize plausible RIRs from reverberant input, while FiNS (Steinmetz, Ithapu, and Calamia 2021a) and DECOR (Lin, Götz, and Schlecht 2025a) incorporate physical constraints for improved realism.

Audio Watermarking. Deep learning has advanced audio watermarking by enabling both watermark detection (0-bit, indicating presence) and message embedding (N-bit, carrying specific information), with improved robustness and perceptual quality. These methods support applications such as copyright protection and content authentication. Liu et al. (Liu et al. 2023) improve distortion resistance using frequency domain models. AudioSeal (San Roman et al. 2024) detects tampering by localizing watermark regions. WavMark (Chen et al. 2023) employs invertible neural networks to jointly model embedding and extraction. While these approaches focus on speech content or speaker identity, none have addressed watermarking the room impulse response. This is particularly important for acoustic environment matching, where the RIR encodes the identity of the space. Watermarking the RIR provides proactive protection against misuse of environment-transferred audio and helps address concerns for AEM service providers.

System Overview

An overview of the proposed EchoMark system is illustrated in Fig. 1. EchoMark takes a reverberant signal from the target environment to generate a watermarked RIR, which is then convolved with clean or dereverberated speech for environment transfer. During detection, the transferred speech produced by the EchoMark model contains an embedded watermark, along with auxiliary message bits that can be reliably extracted.

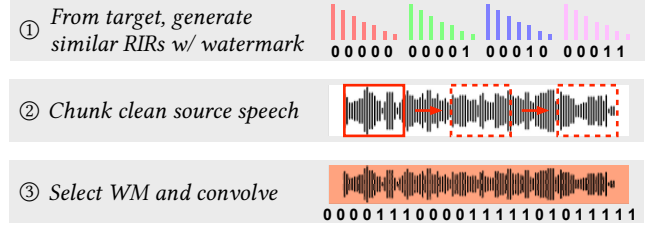


Figure 2: EchoMark’s alternative sequential mode.

Problem Formulation

① Natural Reverberant Speech Generation. Natural reverberant speech can be modeled as the convolution of clean speech with a room impulse response, along with additive noise. This process can be represented both in the time domain and in the short-time Fourier transform (STFT) domain using the convolutive transfer function (CTF) model (Tallmon, Cohen, and Gannot 2009), which is commonly employed in speech and audio signal processing. The formulations are given by:

$$y(n) = \sum_{\tau=0}^{N-1} x(n-\tau)h(\tau) + w(n), \quad (1)$$

$$Y(n', f) = \sum_{\tau=0}^{N-1} X(n'-\tau, f)H(\tau, f) + W(n', f),$$

where x , y , h , and w denote the clean speech, reverberant speech, RIR, and additive noise signals at time index n , respectively. X , Y , H , and W are their corresponding time-frequency representations at STFT frame n' and frequency bin f . The convolution with the RIR causes the energy of the clean speech to extend across successive frames. Note that we will omit the time and frequency index for simplicity in the remainder of this paper.

② Target RIR Generation with Watermark. Building on the formulation in ①, if we can generate an estimated RIR \hat{h} that preserves the room acoustic characteristics of a target RIR h , then convolving \hat{h} with clean source speech will yield perceptually consistent environment-transferred audio. Considering watermark protection, we have two objectives: to generate the RIR waveform ensuring perceptual similarity and watermark fidelity. The generation process is formulated as:

$$\hat{h} = G(E(Y), m) \quad \text{s.t.} \quad P(\hat{h}) \approx P(h), \quad D(Y') = m, \quad (2)$$

where E denotes the RIR encoder, G the RIR generator, and m the embedded watermark. $P(\cdot)$ represents a perceptual evaluation of the acoustic environment, and $D(\cdot)$ is the watermark decoder applied to the generated reverberant signal $Y' = \text{STFT}(x' * \hat{h})$, with $*$ denoting convolution.

③ Watermark Detection on AEM Speech. In this step, the goal is to verify whether a given reverberant speech signal \hat{Y} was generated by the exact AEM model by service provider, i.e., EchoMark. The watermark detector $D(\cdot)$ is

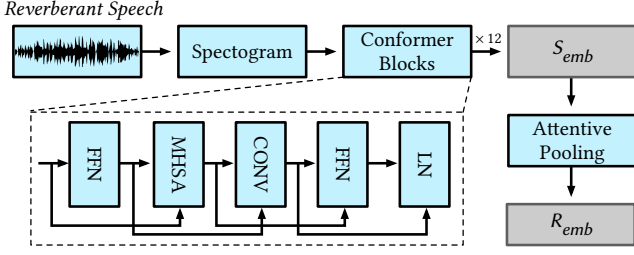


Figure 3: **EchoMark RIR Encoder**. FFN: Feed-Forward Network; MHSA: Multi-Head Self-Attention; CONV: Convolutional Layer; LN: Layer Normalization.

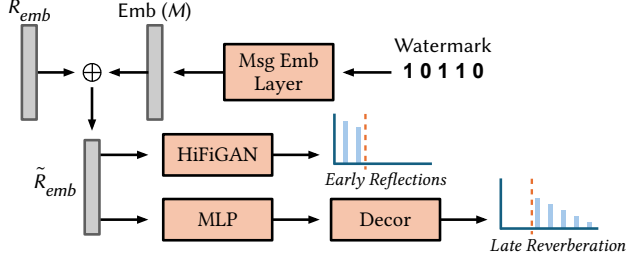


Figure 4: **EchoMark RIR Generator**. Early: HiFiGAN decoder architecture; Late: Decor architecture.

applied to the potential environment-transferred input and produces the following output:

$$D(\tilde{Y}) = \begin{cases} (1, m), & \text{if } \tilde{Y} = \hat{Y}, \\ (0, z), & \text{otherwise,} \end{cases} \quad (3)$$

where the output $(1, m)$ indicates the presence of a valid watermark along with the decoded message m with M bits. If the watermark is not detected, the detector returns $(0, z)$, where z is an arbitrary or meaningless output to be discarded.

Sequential Mode for Long Message. As formulated in Eq. 2, the generator should support arbitrary watermark messages m while maintaining perceptual similarity to the target RIR. That is, for a given acoustic environment h , the generator can produce a family of watermarked RIRs such that $P(\hat{h}(m_1)) \approx P(\hat{h}(m_n)) \approx P(h)$, allowing 2^n different messages to be embedded without altering the perceived environment. When it is required to embed long messages, i.e., more bits, we adopt a simple sequential protocol as alternative mode: the clean source speech is divided into chunks, each convolved with a similar RIR with independent watermark bits, corresponding to a portion of the message. This enables sequential message embedding and decoding within a single long utterance (e.g., 5 bit per second), shown in Fig. 2. Although it is possible to embed more information bits in sequential mode, it requires precise temporal synchronization between generated and test reverberant. In this work, we assume perfect synchronization while mainly focus on the single watermark mode.

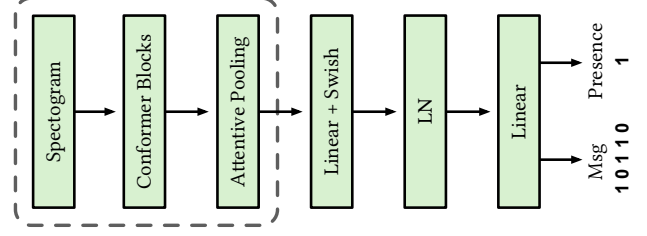


Figure 5: **EchoMark Watermark Detector**. Encoder architecture reused in dashed box.

EchoMark

RIR Encoder. We adopt the Conformer architecture (Gulati et al. 2020) as the backbone of the RIR encoder, which is well-suited for capturing both global and local acoustic patterns. The architecture is shown in Fig. 3. To aggregate temporal information, we apply attentive pooling, following the strategy used in the ECAPA-TDNN model (Desplanques, Thienpondt, and Demuynck 2020) for speaker verification. The encoding process is defined as:

$$\begin{aligned} S_{emb} &= \text{Conformer}(Y), \\ R_{emb} &= \text{AttnPool}(S_{emb}), \end{aligned} \quad (4)$$

where $Y \in \mathbb{R}^{T \times F}$ is the magnitude spectrogram of the input reverberant speech, $S_{emb} \in \mathbb{R}^{T \times d}$ is the encoded sequence, and $R_{emb} \in \mathbb{R}^d$ is the final RIR embedding vector. Note that the produced S_{emb} can also be used for source speech dereverberation, presented as

$$X' = \sigma(\text{ResBlocks}(S_{emb})) \cdot Y,$$

where σ is the sigmoid function. The enhanced spectrogram X' can be converted to waveform using a neural vocoder.

RIR Generator. In Fig. 4, to embed a watermark into the RIR, we first map the binary message $m \in \mathbb{B}^M$ into the latent space using a trainable embedding layer and add it to the original RIR embedding:

$$\tilde{R}_{emb} = R_{emb} + \text{Embedding}(m), \quad (5)$$

where m is an M -bit watermark message. From the empirical results, the maximum number of information bits is set as $M = 5$. Following prior work that separates early reflections and late reverberation (Valimaki et al. 2012; Steinmetz, Ithapu, and Calamia 2021b; Lin, Götz, and Schlecht 2025b), we generate the watermarked RIR \hat{h} as:

$$\begin{aligned} \hat{h} &= \hat{h}_{\text{early}} + \hat{h}_{\text{late}}, \\ \hat{h}_{\text{early}} &= \text{HiFiGAN}(\tilde{R}_{emb}), \\ \hat{h}_{\text{late}} &= \sum_{m,n} A_m^n \exp(-\lambda_n t) w_m(t), \end{aligned} \quad (6)$$

where \hat{h}_{early} (up to 50 ms) is generated by a HiFiGAN-based module (Kong, Kim, and Bae 2020), and \hat{h}_{late} models late reverberation as subband-filtered exponentially decaying noise, similar to DECOR (Lin, Götz, and Schlecht

2025b). Unlike the original DECOR, we condition both components on the embedding \tilde{R}_{emb} , and initialize the decay rate as $\lambda_n = \tau \cdot \text{Sigmoid}(\text{seed})$, where τ is a preset maximum reverberation time (e.g., 3 seconds).

Watermark Detector. To verify the embedded watermark, we use the same architecture as the RIR Encoder with separately trained weights, shown in Fig. 5. After attentive pooling, the output passes through a linear layer followed by a Swish activation, and then another linear projection. The final output has dimension $M + 1$, where the first element indicates watermark presence and the remaining M elements correspond to the message bits. During training, raw outputs are used directly, while inference applies thresholding at zero for binarization.

Model Training

Data Preparation and Augmentation

To construct the training data, we convolve the clean utterances with real RIRs, which provides direct supervision in RIR reconstruction. Both clean speech and RIR waveforms are clipped to 2 seconds per training instance to capture long-range reverberation patterns. The resulting reverberant audio is also truncated to 2 seconds for consistency. To improve robustness, we augment the input reverberant speech by mixing it with isotropic noise at randomly sampled SNR between 0 and 20 dB. This augmentation enhances the model’s generalization to real-world noisy recordings. Finally, the watermarked RIR is convolved with a randomly shuffled batch of clean speech to train watermark detector, accounting for the speech content variation. Additionally, same amount of reverberant speech using real RIRs are also included, providing balanced training samples for watermark presence detection.

Loss Functions

RIR Perceptual Loss. We apply a multi-resolution STFT loss that includes spectral convergence and log-magnitude components:

$$\begin{aligned}\mathcal{L}_{\text{SC}}(\hat{h}, h) &= \frac{\left\| |\text{STFT}(h)| - |\text{STFT}(\hat{h})| \right\|_F}{\left\| \text{STFT}(h) \right\|_F}, \\ \mathcal{L}_{\text{SM}}(\hat{h}, h) &= \frac{1}{N} \left\| \log \left(\frac{|\text{STFT}(h)|}{|\text{STFT}(\hat{h})|} \right) \right\|_1, \\ \mathcal{L}_{\text{STFT}}(\hat{h}, h) &= \sum_{r=1}^R \left[\mathcal{L}_{\text{SC}_r}(\hat{h}, h) + \mathcal{L}_{\text{SM}_r}(\hat{h}, h) \right],\end{aligned}\quad (7)$$

where N is the number of time-frequency bins per resolution, and $\|\cdot\|_F$ and $\|\cdot\|_1$ denote Frobenius and element-wise norms, respectively. To further capture reverberation dynamics, we incorporate an energy decay curve (EDC) loss using the Schroeder integration method (Schroeder 1965), defined

as:

$$\begin{aligned}\text{EDC}_x(t) &= \frac{1}{\text{EDC}_x(0)} \sum_{n=t}^{T-1} x^2(n), \\ \mathcal{L}_{\text{EDC}}(\hat{h}, h) &= \left\| \log(\text{EDC}(h)) - \log(\text{EDC}(\hat{h})) \right\|_1.\end{aligned}\quad (8)$$

The total RIR perceptual loss is given by:

$$\mathcal{L}_{\text{RIR}} = \mathcal{L}_{\text{STFT}} + \mathcal{L}_{\text{EDC}}. \quad (9)$$

Watermarking Loss. To supervise watermark embedding and detection, we define a loss for watermark presence detection and message decoding. The prediction vector has dimension $M + 1$, where the first element indicates watermark presence and the remaining M elements represent the message bits. For all $M + 1$ bits, we adopt a margin-based hinge loss:

$$\mathcal{L}_{\text{wm}} = \frac{1}{M + 1} \sum_{i=0}^M \max(0, 1 - y_i \cdot \hat{y}_i), \quad (10)$$

where $y_i \in \{-1, 1\}$ is the ground-truth label (with binary values mapped to -1 and 1), and \hat{y}_i is the model output before thresholding. We find that hinge loss yields more stable training than binary cross-entropy (BCE), especially when the RIR generation and message decoding objectives compete in the late training stage.

Overall Loss. Finally, to jointly achieve the two optimization objectives, the RIR perceptual loss and the watermarking loss are summed as follows

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{RIR}} + \alpha \mathcal{L}_{\text{wm}}. \quad (11)$$

where α is a weighting factor, set as 1.

Experiment Setup

Dataset and Training. We use LibriSpeech (Panayotov et al. 2015) as the source of clean speech, and construct the target RIR set by merging real RIR datasets from RWCP (Nakamura et al. 2000), the REVERB Challenge (Kinoshita et al. 2017), AIR (Jeub, Schafer, and Vary 2009), and BUT Reverb (Szöke et al. 2019). For training and testing, we use the *train-clean-100* and *dev/test-clean* subsets of LibriSpeech, respectively. The merged RIR dataset is split into training and testing sets with an 80:20 ratio. The input clean speech and reverberant are all set to be 2 seconds in training. Although there is no restriction in testing, short input (as well as chunks in sequential mode) yield faster inference. We set the number of message bits for each watermarked RIR to 5. All models are jointly trained using the AdamW optimizer (Loshchilov and Hutter 2017) with a learning rate of 1×10^{-4} with batch size 40, on a server equipped with 64 CPU cores and two NVIDIA A100 GPUs (80 GB VRAM).

Evaluation Metrics. We compute two standard acoustic parameters using the PyAcoustics toolbox (Mahrt 2016): reverberation time (T_{60}) and direct-to-reverberation ratio (DRR). T_{60} measures the time required for the RIR energy to decay by 60 dB, while DRR measures the ratio between direct and reverberant energy, known as

$$\text{DRR (dB)} = 10 \log_{10} \left(\frac{\sum_{n \in d} h(n)^2}{\sum_{n \in r} h(n)^2} \right), \quad (12)$$

Table 1: Performance on room acoustic metrics and watermark detection/decoding.

Model	T_{60}			DRR			WM Acc. (%) \uparrow
	Bias (s) $\downarrow\downarrow$	RMSE (s) \downarrow	ρ % \uparrow	Bias (dB) $\downarrow\downarrow$	RMSE (dB) \downarrow	ρ % \uparrow	/ BER (%) \downarrow
Noiseless Input							
FiNS	0.041	0.237	64.6	0.430	4.45	72.1	No WM
EchoMark	0.003	0.226	94.9	-2.22	4.03	95.2	100/0.30
Noisy Input (SNR)							
EchoMark (20dB)	0.013	0.228	93.9	-2.21	3.91	94.8	99.6/0.20
EchoMark (10dB)	0.003	0.270	95.7	-2.26	3.93	92.5	99.7/0.30
EchoMark (0dB)	0.021	0.365	87.5	-2.62	4.27	91.1	99.7/0.30

where d denotes the direct path window and r the subsequent reverberant region. For both metrics, we report the bias, root mean squared error (RMSE), and Pearson correlation coefficient ρ with respect to the ground truth. To evaluate watermarking performance, we report detection accuracy and bit error rate (BER) for message decoding, computed over 20 test batches (800 samples and 4000 bits in total).

Results

Overall Performance

We first evaluate the acoustic parameters of the impulse responses generated by EchoMark. Compared to the ground-truth RIRs, smaller bias and RMSE indicate greater perceptual similarity. As a baseline, we compare EchoMark with FiNS (Steinmetz, Ithapu, and Calamia 2021b), a state-of-the-art RIR estimation model that also takes reverberant speech as input. Results are presented in Table 1.

T_{60} . Both FiNS and EchoMark exhibit minimal bias and comparable RMSE. However, EchoMark exhibits a stronger linear correlation with the ground truth, meaning it better preserves the relative variation in estimated reverberation. This improvement is attributed to the explicit modeling of late reverberation in our architecture.

DRR. For DRR, EchoMark tends to produce slightly smaller values, implying generating a stronger reverberant tail relative to the direct path (Eq. 12). Although this result is slightly less accurate than FiNS, the discrepancy is minor as the DRR in the dataset spans a wide range (up to 30 dB), supported by the human listening test.

Watermark. Most notably, EchoMark supports watermark embedding and decoding, achieving 100% watermark detection accuracy with a low BER of 0.30% in the test dataset, for 2 second reverberant input/output embedded with 5-bit watermark message.

Visualization. To visualize the similarity of the target and EchoMark RIR, we show the spectrograms in Fig. 6, whose differences in time-frequency distribution are negligible.

Human Subject Listening Test

We conduct a human listening test comparing 15 EchoMark-generated reverberant utterances with their ground-truth

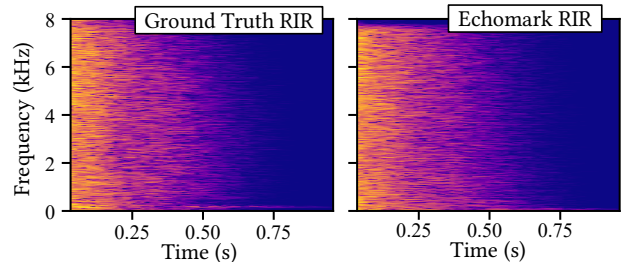


Figure 6: Ground truth (left) and EchoMark RIR (right).

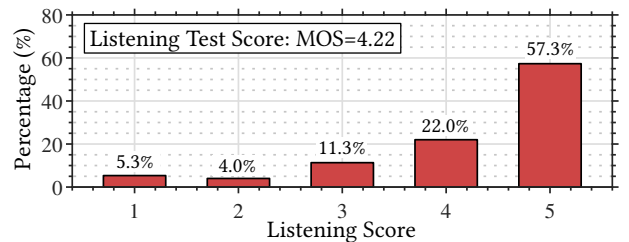


Figure 7: Listening test and mean opinion score.

counterparts. Each pair is played to 20 participants. In the questionnaire, listeners rate how likely the two recordings come from the same acoustic environment on a 5-point Likert scale (1: “very unlikely”, 5: “very likely”). Results are shown in Fig. 7. The mean opinion score (MOS) is 4.22, indicating strong perceptual alignment. Furthermore, 57.3% of the samples received a rating of 5, and 79.3% were rated above the neutral midpoint (score > 3), demonstrating EchoMark’s effectiveness in acoustic environment matching.

Other Impacts

Impact of Noisy Target Reverberant. We test EchoMark with noisy reverberant inputs at SNR levels of 20, 10, and 0 dB, mixed with isotropic noise. The room acoustic RMSEs increase slightly with noise. On the other hand, watermark detection accuracy and BER remain satisfactory across all noise conditions, shown in Table 1.

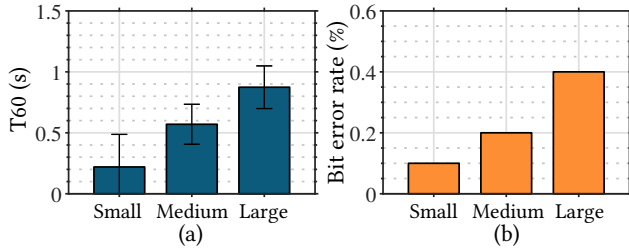


Figure 8: (a) T_{60} and its RMSE w.r.t room types. (b) Corresponding message BER (w/ above 99% detection accuracy)

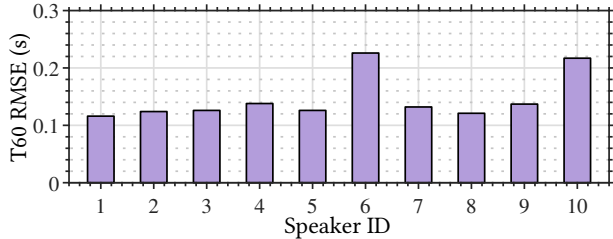


Figure 9: T_{60} RMSE w.r.t. speakers in target reverberant.

Impact of Target Room Type. We examine how different room types affect model performance. In this analysis, we adopt the room type annotations provided in the REVERB dataset (Kinoshita et al. 2017), and report the average T_{60} for each category. As shown in Fig. 8, we have two key observations: ① Compared to medium and large rooms, EchoMark exhibits higher T_{60} RMSE in small rooms, i.e. short RIR cases. This may be due to the limited temporal length available for embedding the watermark information. As a result, the model prioritizes the watermarking accuracy. ② In contrast, larger rooms yield lower T_{60} errors, but at the cost of a slightly increased message BER, from an average of 0.2% to 0.4%. This trade-off illustrates the competing objectives between perceptual RIR reconstruction and watermark embedding. Specifically, longer impulse responses amplify the contribution of the STFT-based loss in Eq. 7, which may overweight its influence relative to the watermark loss during optimization.

Human Speaker Variation in Target Reverberant. We investigate whether variation in human speakers affects the performance of AEM and watermarking. To this end, we select 10 speakers from the LibriSpeech *dev-clean* subset and synthesize 120 reverberant utterances per speaker, each corresponding to a distinct target acoustic environment. The clean source speech used for reverberation is randomly sampled from the dataset. As shown in Fig. 9, the T_{60} evaluation results indicate that EchoMark produces consistent performance across all 10 speakers. Speakers 6 and 10 exhibit slightly higher T_{60} RMSE compared to others, but the difference remains marginal. Importantly, the watermark detection accuracy for all speakers exceeds 99%, with message decoding bit error rates (BER) consistently below 0.3%.

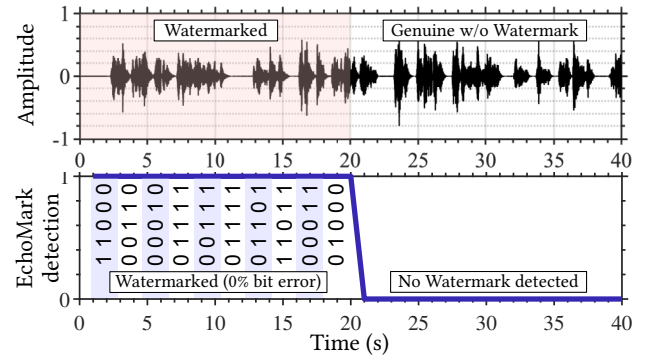


Figure 10: Example of sequential watermark mode.

Example of Sequential Watermark Mode

Finally, we demonstrate an example of sequential watermark embedding using EchoMark. A one-minute reverberant speech sample is first synthesized by convolving clean audio with a real RIR randomly selected from the dataset, simulating a genuine recording in the acoustic environment. For RIR encoding, the first 2-second segment of the reverberant speech is used to extract the target RIR latent embedding. Based on this encoded RIR, we generate 10 watermarked RIRs, each carrying a unique 5-bit sequence to collectively form a 50-bit message. These watermarked RIRs are then convolved with successive 2-second chunks of clean speech. The resulting audio segments replace the corresponding portions of the original reverberant speech, simulating the audio dubbing process. The watermark detector then analyzes the dubbed audio in non-overlapping 2-second windows. As shown in Fig. 10, all watermarked segments in this example are correctly detected, and the embedded messages are recovered without any bit errors. It is worth noting that robust temporal alignment is essential for sequential watermark decoding. Addressing synchronization for RIR watermarking remains an open direction for future work.

Conclusion

We present EchoMark, the first deep learning framework to generate watermarked room impulse responses from reverberant speech for acoustic environment matching. In this design, EchoMark integrates an RIR encoder, generator, and detector, jointly trained for both RIR reconstruction and watermark embedding. Experiments show that EchoMark achieves comparable room acoustic accuracy to the FiNS baseline, which has no watermarking capability. It further achieves over 99% watermark detection accuracy and a message bit error rate below 0.3%. Human listening tests confirm the perceptual similarity between genuine and synthesized audio using the watermarked RIR, with a mean opinion score of 4.22 out of 5. Additional evaluations demonstrate robust performance across different SNR, room types and speakers, and support for sequential watermarking to embed multiple messages. These results highlight the potential of EchoMark to empower creators with perceptually consistent AEM while proactively safeguarding against misuse.

References

- Bahrman, L.; Fontaine, M.; Roux, J. L.; and Richard, G. 2024. Speech dereverberation constrained on room impulse response characteristics. *arXiv preprint arXiv:2407.08657*.
- Chen, G.; Wu, Y.; Liu, S.; Liu, T.; Du, X.; and Wei, F. 2023. Wavmark: Watermarking for audio generation. *arXiv preprint arXiv:2308.12770*.
- Desplanques, B.; Thienpondt, J.; and Demuynck, K. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. *arXiv preprint arXiv:2005.07143*.
- Gulati, A.; Qin, J.; Chiu, C.-C.; Parmar, N.; Zhang, Y.; Yu, J.; Han, W.; Wang, S.; Zhang, Z.; Wu, Y.; et al. 2020. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*.
- Guo, D.; Luo, W.; Gu, C.; Wu, Y.; Song, Q.; Yan, Z.; and Tan, R. 2021. Infrastructure-free smartphone indoor localization using room acoustic responses. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, 343–344.
- Jeub, M.; Schafer, M.; and Vary, P. 2009. A binaural room impulse response database for the evaluation of dereverberation algorithms. In *2009 16th international conference on digital signal processing*, 1–5. IEEE.
- Jia, R.; Jin, M.; Chen, Z.; and Spanos, C. J. 2015. Sound-Loc: Accurate room-level indoor localization using acoustic signatures. In *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, 186–193. IEEE.
- Kinoshita, K.; Delcroix, M.; Gannot, S.; Habets, E. A.; Haeb-Umbach, R.; Kellermann, W.; Leutnant, V.; Maas, R.; Nakatani, T.; Raj, B.; et al. 2017. The REVERB challenge: A benchmark task for reverberation-robust ASR techniques. In *New Era for Robust Speech Recognition: Exploiting Deep Learning*, 345–354. Springer.
- Kong, J.; Kim, J.; and Bae, J. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in neural information processing systems*, 33: 17022–17033.
- Koo, J.; Paik, S.; and Lee, K. 2021. Reverb conversion of mixed vocal tracks using an end-to-end convolutional deep neural network. In *ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 81–85. IEEE.
- Lin, J.; Götz, G.; and Schlecht, S. J. 2025a. Deep room impulse response completion. *EURASIP Journal on Audio, Speech, and Music Processing*, 2025(1): 20.
- Lin, J.; Götz, G.; and Schlecht, S. J. 2025b. Deep room impulse response completion. *EURASIP Journal on Audio, Speech, and Music Processing*, 2025(1): 20.
- Liu, C.; Zhang, J.; Fang, H.; Ma, Z.; Zhang, W.; and Yu, N. 2023. Dear: A deep-learning-based audio re-recording resilient watermarking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 13201–13209.
- Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Mahrt, T. 2016. PyAcoustics. <https://github.com/timmahrt/pyAcoustics>. Accessed: 2025-07-29.
- Mohammadiha, N.; Smaragdis, P.; and Doclo, S. 2015. Joint acoustic and spectral modeling for speech dereverberation using non-negative representations. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 4410–4414. IEEE.
- Nakamura, S.; Hiyane, K.; Asano, F.; Nishiura, T.; and Yamada, T. 2000. Acoustical Sound Database in Real Environments for Sound Scene Understanding and Hands-Free Speech Recognition. In *LREC*.
- Nakatani, T.; Yoshioka, T.; Kinoshita, K.; Miyoshi, M.; and Juang, B.-H. 2010. Speech Dereverberation Based on Variance-Normalized Delayed Linear Prediction. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(7): 1717–1731.
- Neidhardt, A.; Schneiderwind, C.; and Klein, F. 2022. Perceptual matching of room acoustics for auditory augmented reality in small rooms-literature review and theoretical framework. *Trends in Hearing*, 26: 23312165221092919.
- Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 5206–5210. IEEE.
- Ratnarajah, A.; Ananthabhotla, I.; Ithapu, V. K.; Hoffmann, P.; Manocha, D.; and Calamia, P. 2023. Towards improved room impulse response estimation for speech recognition. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.
- Ratnarajah, A.; Tang, Z.; and Manocha, D. 2020. IR-GAN: Room impulse response generator for far-field speech recognition. *arXiv preprint arXiv:2010.13219*.
- San Roman, R.; Fernandez, P.; Elshar, H.; Défossez, A.; Furon, T.; and Tran, T. 2024. Proactive Detection of Voice Cloning with Localized Watermarking. In *International Conference on Machine Learning*, 43180–43196. PMLR.
- Schroeder, M. R. 1965. New Method of Measuring Reverberation Time. *Journal of the Acoustical Society of America*, 37: 409–412.
- Steinmetz, C. J.; Ithapu, V. K.; and Calamia, P. 2021a. Filtered noise shaping for time domain room impulse response estimation from reverberant speech. In *2021 IEEE workshop on applications of signal processing to audio and acoustics (WASPAA)*, 221–225. IEEE.
- Steinmetz, C. J.; Ithapu, V. K.; and Calamia, P. 2021b. Filtered noise shaping for time domain room impulse response estimation from reverberant speech. In *2021 IEEE workshop on applications of signal processing to audio and acoustics (WASPAA)*, 221–225. IEEE.
- Su, J.; Jin, Z.; and Finkelstein, A. 2020. Acoustic matching by embedding impulse responses. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 426–430. IEEE.
- Szöke, I.; Skácel, M.; Mošner, L.; Paliesek, J.; and Černocký, J. 2019. Building and evaluation of a real room

impulse response dataset. *IEEE Journal of Selected Topics in Signal Processing*, 13(4): 863–876.

Talmon, R.; Cohen, I.; and Gannot, S. 2009. Relative transfer function identification using convolutive transfer function approximation. *IEEE Transactions on audio, speech, and language processing*, 17(4): 546–555.

Valimaki, V.; Parker, J. D.; Savioja, L.; Smith, J. O.; and Abel, J. S. 2012. Fifty years of artificial reverberation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(5): 1421–1448.

Wager, S.; Choi, K.; and Durand, S. 2020. Dereverberation using joint estimation of dry speech signal and acoustic system. *arXiv preprint arXiv:2007.12581*.

Zhao, H.; and Malik, H. 2013. Audio recording location identification using acoustic environment signature. *IEEE Transactions on Information Forensics and Security*, 8(11): 1746–1759.

Reproducibility Checklist

Instructions for Authors:

This document outlines key aspects for assessing reproducibility. Please provide your input by editing this .tex file directly.

For each question (that applies), replace the “Type your response here” text with your answer.

Example: If a question appears as

```
\question{Proofs of all novel claims  
are included} {(yes/partial/no)}  
Type your response here
```

you would change it to:

```
\question{Proofs of all novel claims  
are included} {(yes/partial/no)}  
yes
```

Please make sure to:

- Replace **ONLY** the “Type your response here” text and nothing else.
- Use one of the options listed for that question (e.g., **yes**, **no**, **partial**, or **NA**).
- **Not** modify any other part of the `\question` command or any other lines in this document.

You can `\input` this .tex file right before `\end{document}` of your main file or compile it as a stand-alone document. Check the instructions on your conference’s website to see if you will be asked to provide this checklist with your paper or separately.

1. General Paper Structure

- 1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) [yes](#)
- 1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) [yes](#)
- 1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) [yes](#)

2. Theoretical Contributions

- 2.1. Does this paper make theoretical contributions? (yes/no) [no](#)

If yes, please address the following points:

- 2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) [Type your response here](#)
- 2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) [Type your response here](#)
- 2.4. Proofs of all novel claims are included (yes/partial/no) [Type your response here](#)
- 2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) [Type your response here](#)
- 2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) [Type your response here](#)
- 2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) [Type your response here](#)
- 2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) [Type your response here](#)

3. Dataset Usage

- 3.1. Does this paper rely on one or more datasets? (yes/no) [yes](#)

If yes, please address the following points:

- 3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) [yes](#)
- 3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) [NA](#)
- 3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) [NA](#)

- 3.5. All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations (yes/no/NA) [yes](#)
- 3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) [yes](#)
- 3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisfying (yes/partial/no/NA) [NA](#)

4. Computational Experiments

- 4.1. Does this paper include computational experiments? (yes/no) [yes](#)

If yes, please address the following points:

- 4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) [yes](#)
- 4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) [yes](#)
- 4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) [yes](#)
- 4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) [yes](#)
- 4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) [partial](#)
- 4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) [yes](#)
- 4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) [yes](#)
- 4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) [yes](#)
- 4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) [yes](#)
- 4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) [yes](#)
- 4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) [no](#)
- 4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) [NA](#)