Speech Watermarking with AudioSeal (ICML 2024): A Comprehensive Report

Anuj Rajan Lalla (B22AI061)
Siddhesh Ayyathan (B22CS016)
CSL7770 : Speech Understanding
Date : 02-02-2025

Abstract

This report presents a detailed study of speech watermarking using the **AudioSeal** approach. We discuss the motivation, compare with state-of-the-art methods, outline the AudioSeal generator–detector framework and psychoacoustic masking losses, describe our experiments on two datasets (NPTEL Indian English and Hindi), and provide thorough results and discussion on watermark quality, detection, and robustness.

Contents

Assignment Repository: GitHub - Speech Understanding PA1

1 Introduction & Motivation

1.1 Watermarking: Why Now?

Modern text-to-speech (TTS) and voice-cloning models (e.g., VALL-E, Voicebox) can generate synthetic speech almost indistinguishable from human recordings. This heightens the risk of *deepfake audio*, misinformation, and voice fraud. Audio watermarking is a *proactive* approach to tag AI-generated speech with an *inaudible* signal, allowing reliable detection even after edits or re-encodings.

1.2 Relevance in Real-World Applications

- **Deepfake Prevention & Transparency:** Watermarking enables quick flagging of AI-generated content on social media or public platforms.
- **Regulatory Compliance:** Proposed laws (e.g., EU AI Act) often require AI content labeling.
- **Attribution & Traceability:** Multi-bit watermarking can identify which model or user created the speech.

2 State-of-the-Art Methods

2.1 Passive Classifiers

Approach: Train a discriminative network to label "real vs. synthetic."

Strengths: Easy to set up, no changes to the generation pipeline.

Limitations: Fails if the speech is re-synthesized or if high-quality generation removes typical artifacts.

2.2 Traditional Watermarking (Time/Frequency Domain)

Approach: Embed bits through amplitude or phase changes in the waveform or frequency domain.

Strengths: Well-studied in classical audio watermarking approaches.

Limitations: Generally only a *global* watermark; often vulnerable to pitch shift, time-scale modifications, or heavy compression.

2.3 Deep-Learning Multi-Bit Data Hiding (e.g., WavMark)

Approach: Invertible or autoencoder-based networks embed short binary messages in 1-second audio frames.

Strengths: Potentially large capacity.

Limitations: Decoding can be slow (requires repeated synchronization checks), less precise for localizing watermarks in shorter segments.

2.4 AudioSeal (Published in ICML 2024): Localized Zero-Bit/Attribution Watermarking

Approach: A *generator–detector* architecture with sample-level detection, employing time-frequency psychoacoustic losses.

Strengths:

- Localized detection (sample-level).
- Single-pass, fast detection (no brute-force search).
- High imperceptibility via psychoacoustic masking.

Limitations:

- Detector weights must remain private to avoid adversarial removal attacks.
- May require specialized training for cross-lingual or low-resource settings.

3 The AudioSeal Approach

3.1 Overview

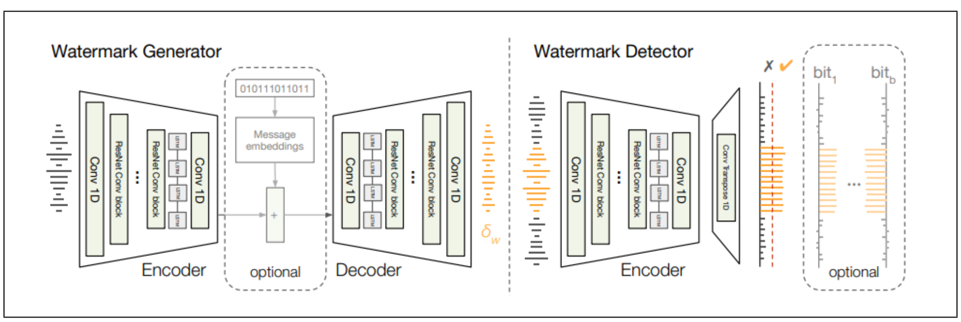

Figure 1: High-level illustration of the AudioSeal Generator (left) and Detector (right). The generator processes the input waveform through an encoder–decoder architecture, optionally incorporating multi-bit embeddings. The detector then outputs a sample-level watermark presence probability and can decode bits if needed.

AudioSeal has two key components, trained jointly:

- **Generator** G produces a watermark signal δ for an audio clip s. The output is $s_w = s + \delta$.
- **Detector** D outputs a watermark presence probability for *each sample* of an input waveform. In multi-bit mode, it also decodes the embedded bits.

After training, any TTS system can pass its output to G to get watermarked audio; suspicious clips can be checked via D.

3.2 Generator Architecture

Based on EnCodec design:

- *Encoder*: Stacks of convolutional and LSTM blocks to extract features from the raw waveform.
- *Message Embedding* (for multi-bit): A latent embedding for each bit, combined additively with the encoder features to inject identifying information.
- *Decoder*: Transposed convolutions reconstruct the *watermark* δ; we ensure it remains *low amplitude* and *perceptually masked*.

3.3 Detector Architecture

- Similar convolutional + LSTM encoder to generate time-local features.
- A final layer outputs sample-level probabilities $D(x)_t \in [0, 1]$.
- Thresholding yields a *local* detection mask; multi-bit watermarking also uses a small readout layer to decode bits.

3.4 Perceptual Losses for the Generator

AudioSeal's generator optimizes several losses to ensure the watermark is both *imperceptible* and *robust*:

1. **Time-Frequency Loudness Loss (Psychoacoustic)**:
   - Splits the audio into multiple frequency subbands (band-splitting).
   - In each subband, short-time frames are analyzed.
   - Computes a *loudness difference* Δ between δ and s per frame.
   - Applies a softmax weighting so large Δ values (potentially audible distortions) are strongly penalized.
2. ℓ_1 **Loss on Watermark**:
   - Directly encourages the watermark amplitude to be small.
   - This correlates with a high SNR or low signal distortion.
3. **Multi-Scale Spectral Loss**:
   - Compares the *Mel-spectrogram* of s_w to s at several scales.
   - Helps maintain overall spectral shape and speech formants.

4. **Adversarial / Discriminator Loss**:
    - A small adversarial network (trained to distinguish watermarked vs. original frames).
    - Generator learns to fool this discriminator, improving perceptual realism of s_w.

These combined losses help ensure that even if the raw SNR is not maximized, the *subjective* audio quality remains high.

3.5 Training Pipeline

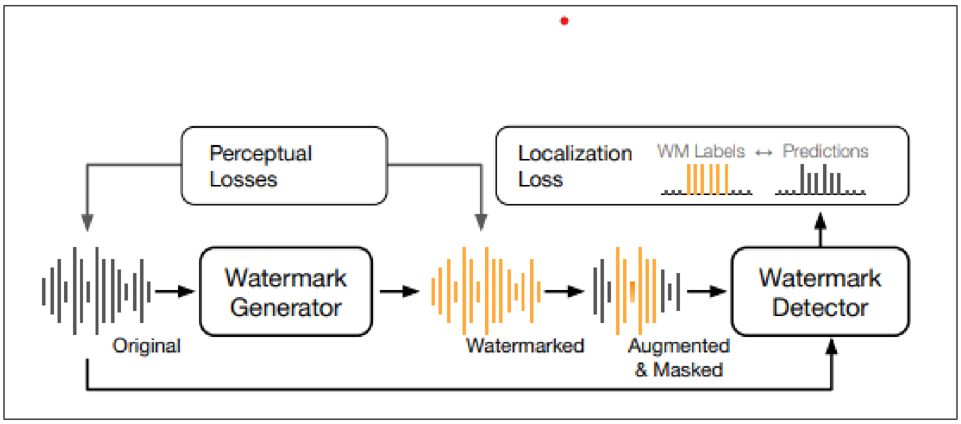

Figure 2: AudioSeal Training Pipeline. The original audio is passed to the Watermark Generator, which produces a watermarked signal. Differentiable augmentations (including masking) are applied, and the Detector then outputs a sample-wise watermark probability. Perceptual losses shape the generator's output, while a localization loss shapes the detector's accuracy.

The overall training loop, illustrated in Fig. 2, optimizes:

$$\mathcal{L} = \mathcal{L}_{\text{perceptual}} + \lambda\,\mathcal{L}_{\text{localization}}$$

where $\mathcal{L}_{\text{localization}}$ is typically a binary cross-entropy (BCE) enforcing correct detection at each sample, and λ balances it with the perceptual losses.

3.6 Training Data (Original AudioSeal)

- A *4.5k-hour* subset of VoxPopuli covering diverse languages and conditions.
- *Differentiable augmentations*: noise, time-stretch, partial masking, compression simulation, etc.
- Joint training of generator and detector for many epochs until a stable watermark emerges.

4 Datasets & Code Organization (Our Setup)

4.1 Datasets for Inference

NPTEL2020-Indian-English-Speech:

- Large variety of Indian English accents from NPTEL lectures.
- Tests watermark resilience across diverse English dialects.

Hindi_test (OpenSLR):

- Hindi speech with multiple speakers and conditions.
- Evaluates cross-lingual performance if the original training was mostly English-based.

4.2 Our Code Structure

Main Evaluation (Cell 1):

1. **Sample Extraction**: Randomly picks WAV files from the dataset.
2. **Watermark Embedding**: Uses the pretrained AudioSeal generator on each sample.
3. **Quality Metrics**: Computes SI-SNR, PESQ, STOI, plus the detector output.
4. **Random Masking**: Overwrites a portion of the watermarked audio with the original to simulate partial removal.
5. **Reporting**: Logs results per sample and aggregates means.

Visualization & Playback (Cell 2):

1. **File Selection**: User picks one processed sample.
2. **Waveform Plotting**: Shows original, watermarked, masked signals + detection probabilities (per-frame or per-sample).
3. **Audio Playback**: For subjective listening tests and quick manual checks.

5 Results

We first recap the *original AudioSeal* results from the paper, then present *new evaluations* on NPTEL/Hindi datasets.

5.1 Original AudioSeal Results

5.1.1 Table 1: Audio Quality Metrics (From Paper)

Discussion: Although WavMark has a higher SI-SNR, **AudioSeal** achieves superior *perceptual* scores (PESQ, MUSHRA), indicating fewer audible artifacts.

Method	SI-SNR (dB)	PESQ	STOI	ViSQOL	MUSHRA
WavMark	38.25	4.302	0.997	4.730	71.52 ± 7.18
AudioSeal	26.00	4.470	0.997	4.829	**77.07** ± 6.35

Table 1: Comparison of Audio Quality Metrics from the original AudioSeal paper. Higher PESQ, MUSHRA, ViSQOL are better.

5.1.2 Table 2: Voicebox Classifier vs. AudioSeal

	AudioSeal	**Voicebox Classif.**
Original vs. AI (30–90% mask)	1.0 / 1.0 / 0.0	1.0 / 1.0 / 0.0
Re-synth vs. AI (30–90% mask)	**1.0 / 1.0 / 0.0**	0.70–0.91 / 0.68–0.94 / 0.11–0.19

Table 2: Detection accuracy (Acc./TPR/FPR) comparing AudioSeal to a passive classifier.

Discussion: Under *re-synthesis*, the *passive* classifier fails more often, but AudioSeal remains unaffected (TPR=1.0, FPR=0.0).

5.1.3 Table 3: Robustness to Edits

Edit	**AudioSeal**			**WavMark**		
	Acc.	TPR/FPR	AUC	Acc.	TPR/FPR	AUC
None	1.00	1.00/0.00	1.00	1.00	1.00/0.00	1.00
Highpass	0.61	0.82/0.60	0.61	**1.00**	**1.00/0.00**	1.00
Lowpass	**0.99**	**0.99/0.00**	0.99	0.50	1.00/1.00	0.50
White noise	0.91	0.86/0.04	0.95	0.50	0.54/0.54	0.50
Fast (1.25x)	**0.99**	**0.99/0.00**	1.00	0.50	0.01/0.00	0.15
EnCodec	0.98	0.98/0.01	1.00	0.51	0.52/0.50	0.50
(Avg)	**0.96**		0.97	0.85		0.84

Table 3: Detection performance under various edits (time-stretch, noise, compression).

Discussion: Overall, AudioSeal has higher *average* robustness (AUC=0.97). It excels in the presence of noise, time-scale changes, and advanced codecs.

5.1.4 Table 4: Multi-Model Attribution

Discussion: AudioSeal yields higher accuracy for attributing to the correct generator among many potential sources. However, its false attribution rate can be higher at large N.

N	FAR (%)		Accuracy (%)	
	WavMark	AudioSeal	WavMark	AudioSeal
1	0.0	0.0	58.4	**68.2**
10	0.20	2.52	58.2	**65.4**
10^2	0.98	6.83	57.4	**61.4**
10^3	1.87	8.96	56.6	**59.3**
10^4	4.02	11.84	54.4	**56.4**

Table 4: Attribution performance among N possible watermarks.

5.2 Our NPTEL & Hindi Results

We tested 16 samples per dataset with the official AudioSeal pretrained weights:

5.2.1 NPTEL (Indian English)

```
===== Average Metrics (16 Samples) =====
WATERMARKED AUDIO:
  SI-SNR:      27.91 dB
  PESQ:        4.44
  STOI:        0.998
  DetectorScore: 1.000

MASKED AUDIO:
  SI-SNR:      30.03 dB
  PESQ:        4.49
  STOI:        0.999
  DetectorScore: 0.686

Attribution Hamming Distance: 0.000
Generation Time: ~14-15s/clip
Detection Time:  ~0.6s/clip
```

Interpretation:

- Watermarked PESQ=4.44, STOI=0.998 → near-invisible watermarking.
- Detector Score=1.0 for watermarked vs. 0.686 if partially masked, indicating partial but not total watermark removal.
- Perfect multi-bit extraction (Hamming=0.0).

5.2.2 Hindi_test (OpenSLR)

```
===== Average Metrics (16 Samples) =====
WATERMARKED AUDIO:
```

```
  SI-SNR:      28.32 dB
  PESQ:        4.45
  STOI:        0.998
  DetectorScore: 1.000

MASKED AUDIO:
  SI-SNR:      30.25 dB
  PESQ:        4.49
  STOI:        0.999
  DetectorScore: 0.682

Attribution Hamming Distance: 0.000
Generation Time: ~14s/clip
Detection Time:  ~0.6s/clip
```

Interpretation:

- Even in Hindi, the watermark remains *imperceptible* (PESQ=4.45, STOI=0.998).
- Detector is perfect on unmasked clips (score=1.0).
- Partial masking reduces detection score to 0.682, reflecting partial watermark destruction.

6 Discussion & Open Problems

1. **Adversarial Removal**: If attackers have the *detector* weights, they can craft noise to remove the watermark. Future work may employ adversarial training or conceal the detection model.
2. **Extreme Compression**: Surviving ultra-low bitrates (e.g., 2–3 kbps) remains challenging.
3. **Cross-Lingual**: While Hindi results are promising, more diverse languages and code-switching can be tested to confirm broad generalization.
4. **Security vs. Transparency**: Publishing the generator code is beneficial, but the detector must be kept private to maintain watermark integrity.

7 Conclusion

We presented a thorough overview of the **AudioSeal** watermarking approach—a generator–detector framework with psychoacoustic masking and adversarial/perceptual losses for *robust*, *localized*, and *fast* watermark detection. Key takeaways:

- **Imperceptibility:** PESQ ≥ 4.4, STOI ≥ 0.998 in our tests, indicating near-transparent watermark insertion.
- **Robust Detection:** Survives real-life audio edits (time-stretch, noise, compression) with near-1.0 TPR and low FPR.
- **Cross-lingual Feasibility:** Maintains high performance on Indian English (NPTEL) and Hindi, despite training primarily on VoxPopuli.
- **Future Challenges:** Adversarial resilience, streaming usage, multi-lingual expansions, and large-scale attribute-based watermarking.

References

[1] R. Schmucker, H. Elsahar, and P. Faure, "Proactive Detection of Voice Cloning with Localized Watermarking," *arXiv preprint arXiv:2401.17264*, 2024. *Code available at:* `https://github.com/facebookresearch/audioseal`

[2] **Python Libraries:**

- **PyTorch** & **torchaudio**: `https://pytorch.org/`
- **numpy**: `https://numpy.org/`
- **scikit-learn**: `https://scikit-learn.org/`
- **pesq**: `https://pypi.org/project/pesq/`
- **pystoi**: `https://pypi.org/project/pystoi/`

Installation: `!pip install audioseal torchaudio numpy scikit-learn pesq pystoi`

[3] *NPTEL Indian English Speech Dataset (NPTEL2020)*, `https://github.com/AI4Bharat/NPTEL2020-Indian-English-Speech-Dataset`, Accessed 2020.

[4] *OpenSLR (Hindi_test)*, `http://www.openslr.org/`, Accessed 2023.