



Latent Secret Spin: Keyed Orthogonal Rotations for Blind Speech Watermarking in Anisotropic Latent Spaces

Emma Coletta, Massimiliano Todisco, Michele Panariello, Antonio Faonio, Nicholas Evans

EURECOM, France

[firstname.lastname]@eurecom.fr

Abstract

We introduce Latent Secret Spin (LSS), a blind speech watermarking method based on geometric operations in codec latent space. Based upon orthogonal rotations to principal components, LSS induces imperceptible but detectable covariance signatures according to a pseudo-random watermarking schedule. The scheme generalises across datasets, preserves perceptual quality and, unlike some learned, neural watermarking schemes, it does not require neural network training, is resistant to common signal manipulations and is flexible to payload size. Analyses show that structured latent-space watermarking is a promising and interpretable alternative to existing approaches.

Index Terms: speech watermarking, blind detection, latent space, neural codecs, interpretability.

1. Introduction

Astonishing advances in generative artificial intelligence have revolutionised multimedia content creation, but have also increased the risk of misuse, including the spread of misinformation, identity fraud, deepfakes and malicious content manipulation. Digital watermarking [1] offers a proactive solution and can be used to embed imperceptible yet traceable markers within digital media, enabling the identification of AI-generated content [1, 2].

In the audio domain, speech watermarking is used to embed imperceptible yet reliably detectable payloads directly into the audio waveform, enabling traceability and integrity verification [3]. By incorporating inaudible markers in the audio, watermarking provides a reliable mechanism for verifying the provenance of a speech recording.

Effective, practical speech watermarking strategies must meet three key criteria [1, 4, 5]:

- **Imperceptibility:** watermarks must not degrade perceptual audio quality or introduce artifacts that are detectable by a human listener.
- **Robustness:** they must remain detectable under a range of signal transformations, such as compression, additive noise and filtering.
- **Security:** they must resist unauthorised removal, spoofing, or forgery under adversarial conditions.

Many practical deployment scenarios also require *blind* watermarking. A watermarking scheme is considered blind if detection can be performed without access to the original, unwatermarked asset [4, 6]. Blind detection is particularly advantageous and often necessary when the original host asset is inaccessible or unavailable. Non-blind approaches are then impractical [2].

We introduce Latent Secret Spin (LSS), a novel principal component-based framework for blind neural speech watermarking. LSS operates directly in the continuous latent space of a neural audio codec, enabling imperceptible watermark embedding while ensuring robust detection under realistic signal transformations. The latent space provides a structured representation in which statistical watermark signatures can be introduced and reliably detected [7, 8, 9]. Recent approaches, such as WavMark [10] and AudioSeal [11], demonstrate the effectiveness of learned embedding-detection pipelines for robust and imperceptible speech watermarking, motivating our use of similar structured latent representations.

LSS projects latent representations into a principal component space to exploit its natural geometric structure. Watermarks are embedded through small, localised orthogonal rotations within targeted principal component planes. The embedding process is controlled by a schedule which specifies a set of embedding parameters deterministically derived from a secret key, including the selection of principal component planes, the order in which rotations are applied, and the corresponding rotation angles. Rotations are distributed across dimensions and time, inducing small but consistent statistical dependencies among principal components that can be exploited for detection. This design enables reliable blind watermark detection which is robust to common signal transformations and unauthorized detection while preserving perceptual audio quality.

LSS is also lightweight in that it operates on the geometry of pre-trained speech representations and does not require the training of any embedding model. To the best of our knowledge, LSS is the first speech watermarking method to induce controlled covariance patterns in a principal component space through geometric transformations, and the first to use covariance patterns explicitly for payload embedding.

The paper is structured as follows. In Section 2 we provide a review of related work in the literature. In Section 3 we introduce the geometric principles underlying the LSS framework, while the algorithm itself is described in Section 4. We outline our experimental setup in Section 5 and report results in Section 6. We discuss the main findings, ongoing research directions, and future work in Sections 7 and 8.

2. Related Work

Early digital watermarking techniques operated directly in the signal domains, e.g. spatial for images, temporal for audio. Alternatively, they embedded payloads in transform domains using standard representations like the Discrete Cosine Transform (DCT), Discrete Fourier Transform (DFT), or Discrete Wavelet Transform (DWT) [1, 2,

4, 5, 12, 13, 14]. Principal Component Analysis (PCA) was used in watermarking to remove linear dependencies and decorrelate data [15, 16]. PCA was typically paired with frequency-domain transformations such as the DCT or DWT to embed payloads into selected frequency components [16, 17, 18, 19]. Within all these frameworks, watermarks were generally embedded by modifying dominant principal components or their associated coefficients [18, 20, 15]. Recent works have also explored PCA for feature extraction in zero-shot watermarking schemes [21, 22]. In the speech domain, PCA has been used to improve watermark robustness and security by separating core speech content from noise and interference, enabling watermark embedding in stable features, such as the formant structure, that are less affected by common signal processing distortions [23, 24].

Recent advances in deep learning have led to neural approaches which encode watermarks into latent representations of speech content, improving both imperceptibility and robustness. End-to-end architectures such as WavMark [10] encode watermarks through waveform-level perturbations learned and applied using an encoder-decoder pipeline, while approaches like AudioSeal [11] emphasize reliable and localised watermark detection through jointly trained generator-detector frameworks. Operating in the latent space of neural audio codecs generally enables high-fidelity watermarks deeply rooted within the underlying core signal features [7, 8, 9]. Nonetheless, we have observed that the additive nature of existing schemes can leave watermarks undetectable in even modest levels of additive noise. We hence seek a different approach.

3. Geometric principles

Latent Secret Spin (LSS) is a blind speech watermarking framework which operates in the continuous latent space of a neural encoder. LSS relies on a simple geometric idea whereby detectable changes in covariance are introduced using small orthogonal rotations in an anisotropic plane defined by principal components. In this section we describe the principle in its simplest form, independently of secret-key scheduling and distributed embedding strategies.

The underlying geometric principle is illustrated in Figure 2. Let x denote an input speech waveform and let

$$F = \mathcal{E}(x) \in \mathbb{R}^{n \times T}$$

be the corresponding sequence of continuous latent features extracted using a neural encoder $\mathcal{E}$, where n denotes the latent dimensionality and T the number of analysis frames. To operate in a space with explicit and analytically tractable covariance structure, latent features are projected into a new space of principal components defined beforehand using PCA and a representative speech corpus. First, the same encoder $\mathcal{E}$ is applied to every utterance in the corpus. We denote by F_c the resulting corpus-level latent feature distribution. Second, we define the global mean and covariance of F_c as

$$\mu = \mathbb{E}[F_c] \in \mathbb{R}^n, \quad \Sigma = \text{Cov}(F_c) \approx U \Lambda U^\top,$$

where the columns of U are the principal components (i.e. the eigenvectors of Σ), where $U^\top U = I$, and where

$$\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n), \quad \lambda_1 \geq \lambda_2 \geq \dots > 0,$$

contains the corresponding eigenvalues sorted in descending order, with U ordered correspondingly. As illustrated

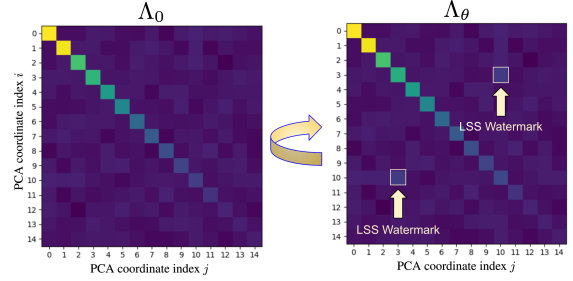


Figure 1: *Illustration of the LSS watermarking geometric principle in PCA space. Λ_0 denotes the covariance matrix before embedding, while Λ_θ denotes the covariance matrix after a small rotation of angle $\theta = 0.18$ rad in the PCA plane $(i, j) = (3, 10)$. The induced off-diagonal covariance terms correspond to the injected watermark. For readability, only a zoomed 15×15 portion of the full PCA space is displayed.*

in Figure 2, latent features are then projected into the space defined by the principal components according to

$$Z = U^\top (F - \mu).$$

The use of principal components provides a coordinate system in which the latent covariance is approximately diagonal, $\text{Cov}(Z) \approx \Lambda$,

implying that latent dimensions are decorrelated. The variance of principal component axes is then, in general, distinct.¹ In this case the principal components space is anisotropic.

A watermark is then embedded by applying small orthogonal rotations to Z in selected principal component planes. Consider a plane spanned by principal components i, j such that $\lambda_i > \lambda_j$. Before rotation, the covariance in plane (i, j) is approximately

$$\Sigma_0 = \text{Cov}([Z_i, Z_j]) \approx \begin{bmatrix} \lambda_i & 0 \\ 0 & \lambda_j \end{bmatrix}.$$

A rotation of angle $\theta > 0$ defined by

$$R(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

is then applied. The covariance is then given by

$$\Sigma_\theta = R(\theta) \Sigma_0 R(\theta)^\top.$$

The rotation introduces an off-diagonal covariance term

$$\begin{aligned} \Delta \text{Cov}_{ij} &= (\lambda_i - \lambda_j) \sin \theta \cos \theta \\ &= \frac{1}{2} (\lambda_i - \lambda_j) \sin(2\theta), \end{aligned} \quad (1)$$

which, for small angles for which $\sin(2\theta) \approx 2\theta$, becomes

$$\Delta \text{Cov}_{ij} \approx (\lambda_i - \lambda_j) \theta. \quad (2)$$

Thus, for $\lambda_i > \lambda_j$ and $|\theta| \ll 1$, $\Delta \text{Cov}_{ij} > 0$ for $\theta > 0$, but $\Delta \text{Cov}_{ij} < 0$ for $\theta < 0$.

This principle, illustrated in Fig. 1, is the fundamental mechanism behind LSS: small rotations in anisotropic planes induce an off-diagonal covariance term, the sign of which carries the sign (direction) of the rotation.

The rotation is applied in principal component space, yielding the watermarked representation Z^* .

¹Unique eigenvalues are not strictly guaranteed. We have not observed repeated eigenvalues in practice.

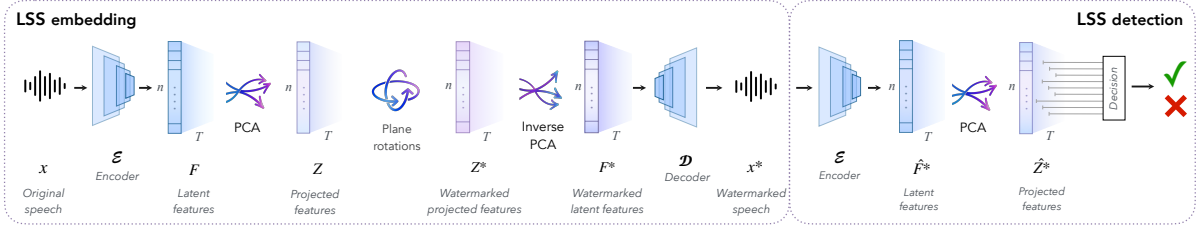


Figure 2: Overview of LSS embedding and detection pipelines. During embedding, the input signal x is encoded into latent features F , projected to principal component space as Z , watermarked to obtain Z^* , mapped back to latent space F^* and then decoded back to utterance x^* , watermarked. Trial signals are encoded and projected into the same space before watermark detection is performed.

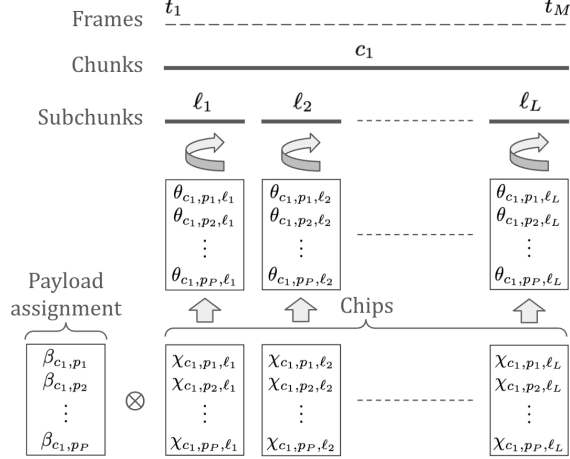


Figure 3: Watermark schedule within one chunk c_1 . The chunk is partitioned into subchunks $\ell_1, \dots, \ell_L$. Each selected plane $p_1, \dots, p_P$ carries a payload sign $\beta_{c_1,p}$, while each subchunk is associated with a plane-dependent chip vector. Their product determines the local signed rotation angles $\theta_{c_1,p,\ell}$.

Watermarked latent features are recovered by inverse projection:

$$F^* = UZ^* + \mu.$$

The decoder $\mathcal{D}$ is then used to reconstruct the watermarked speech waveform according to

$$x^* = \mathcal{D}(F^*).$$

The watermark can then be recovered trivially, even after x^* is subject to various audio manipulations, perhaps applied by an adversary to render the watermark undetectable. The watermarked speech signal x^* can be re-encoded using the same encoder $\mathcal{E}$ and, with knowledge of the same principal components, an estimation $\hat{Z}^*$ can be recovered and tested for the corresponding covariance signature.

In principle, the watermark introduced by the single, deterministic rotation in one principal component plane as described here is weak and insufficient. An adversary could detect or recover such a trivial watermark relatively easily given a sufficient number of watermarked utterances. As we shall now show, we build on this core principle by distributing rotations across multiple planes and temporal segments. With knowledge of these planes and segments, local rotations can then be aggregated into statistically significant, detectable perturbations. In practice, rotations are applied in planes and segments selected according to a secret-key-controlled schedule, which guarantees undetectability without knowledge of the same secret-key.

Algorithm 1 LSS Watermark Embedding

Input: Input signal x , PCA basis (U, μ) , encoder $\mathcal{E}$, decoder $\mathcal{D}$, key K , nonce N , rotation magnitude θ , payload bits $\mathbf{b} = (b_1, \dots, b_B)$

Output: Watermarked signal x^*

```

 $F \leftarrow \mathcal{E}(x)$ 
 $Z \leftarrow U^\top (F - \mu)$ 
Convert payload bits  $\mathbf{b}$  into signed circular list  $\beta = (\beta_1, \dots, \beta_B)$ 
Partition  $Z$  into  $C$  chunks  $[c_1, \dots, c_C]$ 
// Each chunk  $c$  is associated to a list of planes generated pseudo-randomly
for each chunk  $c$  in  $[c_1, \dots, c_C]$  do
     $[p_1, \dots, p_P]_c \leftarrow \text{pseudo\_random}(K, N)$ 
    // Each plane  $p$  encodes a bit from  $\beta$ 
    for each plane  $p$  in  $[p_1, \dots, p_P]_c$  do
         $\beta_{c,p} \leftarrow \text{get\_next\_bit}(\beta)$ 
        Divide  $c$  into  $L$  subchunks  $[\ell_1, \dots, \ell_L]$ 
        // One bit is scattered across subchunks.
        // Each subchunk is associated to a chip  $\chi$  generated pseudo-randomly
        for each subchunk  $\ell$  in  $[\ell_1, \dots, \ell_L]$  do
             $\chi_{c,p,\ell} \leftarrow \text{pseudo\_random}(K, N)$ 
             $\theta_{c,p,\ell}^* \leftarrow \beta_{c,p} \cdot \chi_{c,p,\ell} \cdot \theta$ 
             $Z^* \leftarrow \text{Rotate plane } p \text{ by } \theta_{c,p,\ell}^* \text{ over each } t \text{ in subchunk } \ell$ 
 $F^* \leftarrow UZ^* + \mu$ 
 $x^* \leftarrow \mathcal{D}(F^*)$ 
return  $x^*$ 

```

4. Latent Secret Spin

We now describe the practical implementation of LSS, including watermark embedding, watermark detection, and the keyed pseudo-random schedule.

4.1. Watermark embedding

The embedding procedure is applied to Z across temporal chunks and subchunks, and across a set of geometric planes. The process is described in the following, illustrated in Fig. 3 and summarised in Algorithm 1.

First, the sequence of projected latent features Z is partitioned into C chunks $c_1, \dots, c_C$ each of length M frames $t_1, \dots, t_M$ where t denotes the frame index. For each chunk c , we select according to the pseudo-random schedule a set of principal component planes $p_1, \dots, p_P$ where each plane p corresponds to a pair of principal directions (i_p, j_p) . Each chunk is further subdivided into L subchunks $\ell_1, \dots, \ell_L$.

The watermark is encoded as a B -bit sequence referred

to as the *payload*. In practice, it is represented as a vector $\beta \in \{\pm 1\}^B$, where bit values 1 become +1 and bit values 0 become -1. Signed bits from β are embedded into Z sequentially; namely, each signed payload bit $\beta_{c,p} \in \{\pm 1\}$ is embedded through rotations in a plane p across the chunk c . When all bits have been assigned, β is repeated cyclically.

The embedding of a bit in a plane is distributed over temporal subchunks:

for each selected plane p and each subchunk ℓ , we determine from the schedule a pseudo-random chip

$$\chi_{c,p,\ell} \in \{\pm 1\},$$

which controls the direction of the local rotation in that plane and subchunk (see Fig. 3).

The resulting signed rotation for chunk c , plane p and sub-chunk ℓ is hence given by

$$\theta_{c,p,\ell}^* = \beta_{c,p} \cdot \chi_{c,p,\ell} \cdot \theta$$

where $\theta > 0$ is a fixed rotation angle chosen as hyperparameter. The value of $\theta_{c,p,\ell}^*$ is either $+\theta$ or $-\theta$, where the sign is determined by the signs of the bit and the chip. Hence, within each subchunk ℓ , chip $\chi_{c,p,\ell}$ acts to modulate the signed payload carrier $\beta_{c,p}$. For each chunk c and each plane $p = (i_p, j_p)$, each feature frame $Z(t)$ has its i_p and j_p components rotated according to

$$\begin{bmatrix} Z_{i_p}^*(t) \\ Z_{j_p}^*(t) \end{bmatrix} = R(\theta_{c,p,\ell}^*) \begin{bmatrix} Z_{i_p}(t) \\ Z_{j_p}(t) \end{bmatrix}, \quad t \in \ell.$$

Thus, the watermark is distributed across multiple geometric planes and temporal segments. After the set of rotations has been applied, the modified representation Z^* is projected back to the original latent space F^* and then decoded, yielding the watermarked waveform x^* .

4.2. Watermark detection

A summary of the detection procedure is presented in Algorithm 2. Using the same neural encoder $\mathcal{E}$, the detector first projects a trial waveform x' into latent space F' . Then, according to the same principal components used for initial encoding, an approximation of potentially watermarked features Z' is recovered. The representation is then processed according to the same watermark schedule, with chunks $c_1, \dots, c_C$, subchunks $\ell_1, \dots, \ell_L$, planes $p_1, \dots, p_P$ and chips $\chi_{c,p,\ell}$. Then, for each chunk c , plane p , and each subchunk ℓ , the detector computes the normalised covariance

$$C_{c,p,\ell} = \frac{1}{|\ell| \sqrt{\lambda_{i_p} \lambda_{j_p}}} \sum_{t \in \ell} (Z'_{i_p}(t) - \bar{Z}'_{i_p,\ell}) (Z'_{j_p}(t) - \bar{Z}'_{j_p,\ell}),$$

where p is the plane spanning principal components (i_p, j_p) , $\bar{Z}'_{i_p,\ell}$ and $\bar{Z}'_{j_p,\ell}$ denote the local mean for each component over sub-chunk ℓ , and $|\ell|$ is the number of frames per sub-chunk. Bits $\beta_{c,p}$ and chips $\chi_{c,p,\ell}$ are derived from the schedule in the same way as in the embedding process. They are accumulated, along with each corresponding $C_{c,p,\ell}$, into a detection score

$$S = \sum_c \sum_p \sum_\ell \beta_{c,p} \chi_{c,p,\ell} C_{c,p,\ell} \quad (3)$$

where the bit and the chip compensate for the sign modulation introduced during embedding. In this case, signed covariance terms sum constructively and the score S is positive. In the case of a different schedule produced by an incorrect key, a mismatched payload, or an absent watermark,

Algorithm 2 LSS Watermark Detection

Input: Investigated signal x' , PCA basis (U, μ) , encoder $\mathcal{E}$, key K , nonce N , payload β , decision threshold τ
Output: Decision if β is watermarked into x'

```

 $S \leftarrow 0$  // Watermark score tracker
 $F' \leftarrow \mathcal{E}(x')$ 
 $Z' \leftarrow U^\top (F' - \mu)$ 
// Loop as in the embedding process
Partition  $Z'$  into  $C$  chunks  $[c_1, \dots, c_C]$ 
for each chunk  $c$  in  $[c_1, \dots, c_C]$  do
     $[p_1, \dots, p_P]_c \leftarrow \text{pseudo\_random}(K, N)$ 
    for each plane  $p$  in  $[p_1, \dots, p_P]_c$  do
         $\beta_{c,p} \leftarrow \text{get\_next\_bit}(\beta)$ 
        Divide  $c$  into  $L$  subchunks  $[\ell_1, \dots, \ell_L]$ 
        for each subchunk  $\ell$  in  $[\ell_1, \dots, \ell_L]$  do
             $\chi_{c,p,\ell} \leftarrow \text{pseudo\_random}(K, N)$ 
            // Compute covariance term of plane  $p$  over
            // individual frames of subchunk  $\ell$ 
             $i_p, j_p \leftarrow \text{pair of principal components defining } p$ 

             $C_{c,p,\ell} \leftarrow \frac{\sum_{t \in \ell} (Z'_{i_p}(t) - \bar{Z}'_{i_p,\ell}) (Z'_{j_p}(t) - \bar{Z}'_{j_p,\ell})}{|\ell| \sqrt{\lambda_{i_p} \lambda_{j_p}}}$ 

             $S \leftarrow S + \chi_{c,p,\ell} \cdot \beta_{c,p} \cdot C_{c,p,\ell}$ 
return True if  $S > \tau$ , False otherwise

```

covariance terms are equally likely to have positive or negative sign: they will sum destructively in (3), producing a lower score. Binary detection decisions are then produced by a comparison between S and a decision threshold.

4.3. Keyed pseudo-random schedule generation

LSS watermark schedules are generated pseudo-randomly using a secret key K and a utterance-specific nonce N and are shared by the encoder and decoder. This keyed schedule determines the selection of planes p and the chips $\chi_{c,p,\ell}$ during embedding.

The detector evaluates the covariance terms in the same sequence of planes and chip signs as during embedding, reproduced deterministically from the pseudo-random function using the same K, N pair. Without it, the plane and chip sequence is different: therefore, the sign of $C_{c,p,\ell}$ oscillates randomly between positive and negative, and the contributions in (3) cancel out on average. This results in $S \approx 0$, preventing watermark detection by any authorized party who does not possess both K and N .

5. Experimental Setup

We evaluate the effectiveness of the proposed LSS framework in both in-domain and out-of-domain settings. To assess robustness, we apply various audio manipulations to watermarked speech before detection.

5.1. Datasets

We evaluate the proposed LSS framework using two different speech datasets: VoxPopuli [25] and ASVspoof 5 [26]. For both, we create two speaker-disjoint partitions. The first is used to derive the set of principal components, while the second is used for evaluation. We use the English subset of the 100k-hour unlabelled partition of VoxPopuli. Principal components are estimated using approximately 100k samples, while evaluation is performed using a second set of 10k utterances for embedding and detection experiments.

We use data sourced from the ASVspoof5 evaluation partition from which we select uncompressed bona fide utterances. From $\approx 35\text{k}$ utterances, we set aside 10k samples for evaluation and use the remaining 25k utterances to estimate principle components.

Following [11], all utterances are standardised to a duration of 10 s. Shorter recordings are circularly padded to avoid introducing zeros that would otherwise distort latent covariance statistics, while longer utterances are truncated.

5.2. Configuration

Encoder $\mathcal{E}$ and decoder $\mathcal{D}$ are pre-trained 24 kHz implementations of EnCodec [27]² operating at a target bandwidth of 6.0 kbps. Feature representations have dimensionality $n = 128$ and are produced at a frame rate of 75 Hz.

We use chunk sizes of $M = 32$ frames (≈ 427 ms), subchunk sizes of $L = 8$ frames (≈ 107 ms), and $P = 24$ planes.

We achieve reliable detection and imperceptible watermarking using rotations of $\theta = 0.18$ rad.

Source code³ is available and can be used to reproduce all LSS results reported in this paper. Those of other techniques are reported from [11].

5.3. Audio manipulations

We apply seven manipulations to watermarked utterances x^* before detection, namely sixth-order Butterworth low-pass, highpass, and bandpass filters, MP3 compression at bitrates from 12 to 256 kbps, resampling from 24 kHz to 16 kHz and back, and additive white Gaussian and $1/f$ pink noise at SNRs from 5 to 20 dB.

5.4. Evaluation metrics

Following [11], we report AUC-ROC of watermark detection. We do not compute accuracy as it does not reflect the overall strength of the method, but rather its performance at a specific, use case-dependent decision threshold.

In addition, we estimate imperceptibility using the two-sided objective perceptual evaluation of speech quality metric with a wideband setting (PESQ-WB).

6. Results

In the following we present an analysis of detection performance, robustness and estimated perceptual quality.

6.1. Detection Performance

Detection results are reported in Table 1 for both in-domain and out-of-domain conditions, depending on whether principal components are estimated and evaluated on the same dataset or on different datasets. In all cases, detection performance remains strong when the correct key is used, with consistently high AUC values. The close agreement between in-domain and out-of-domain results indicates that the principal component representation generalises well across datasets and that detection does not rely on dataset-specific characteristics. By contrast, when an incorrect key is used, performance collapses to random guess level ($\text{AUC} \approx 0.5$), confirming that reliable detection depends on knowledge of the correct secret key.

LSS achieves strong detection performance across most distortions, as shown in Table 2. In particular, un-

Table 1: *Detection performance on clean watermarked speech for all dataset scenarios. Results are shown for the correct key only; incorrect-key detection is at random guess level ($\text{AUC} \approx 50.0\%$).*

Scen.	Dom.	Training	Evaluation	AUC (%)
T1	In	VoxPopuli-100k	VoxPopuli-100k	99.6
T2	In	ASVspoof 5	ASVspoof5	99.7
T3	Out	VoxPopuli-100k	ASVspoof5	99.7
T4	Out	ASVspoof 5	VoxPopuli-100k	99.3

Table 2: *Detection performance under non-malicious audio manipulations. Results are reported for LSS and AudioSeal on configuration T1 of Table 1.*

Manipulation	Configuration	LSS AUC (%)	AudioSeal AUC (%)
None	–	99.6	100.0
Lowpass	$f_c = 1$ kHz	96.5	67.8
Lowpass	$f_c = 1.5$ kHz	98.1	100.0
Highpass	$f_c = 1$ kHz	87.3	100.0
Highpass	$f_c = 1.5$ kHz	80.3	100.0
Bandpass	500 Hz – 5 kHz	97.4	100.0
MP3	32 kbps	99.5	100.0
Resample	[24 $\rightarrow$ 16 $\rightarrow$ 24] kHz	99.7	100.0
White Noise	SNR = 5 dB	94.8	99.8
White Noise	SNR = 20 dB	99.3	100.0
Pink Noise	SNR = 5 dB	95.6	99.9
Pink Noise	SNR = 20 dB	99.4	100.0
Average		95.6	97.2

der common channel effects such as MP3 compression, resampling, additive noise at 20 dB, and bandwidth filtering, performance remains close to the clean condition, indicating good robustness to realistic transmission degradations. For contextual reference, AudioSeal [11] achieves a slightly higher average AUC overall (97.2% versus 95.6% for LSS), while LSS remains close on average and outperforms AudioSeal under lowpass filtering at $f_c = 1$ kHz. These manipulations were chosen to cover a range of common post-processing operations, including spectral filtering, compression, resampling, and additive noise.

To further analyse robustness, Fig. 4 reports detection accuracy as a function of manipulation intensity for several distortions. Results are shown for both matched (T2) and mismatched (T3) scenarios defined in Table 1. Across all conditions, the two curves remain closely aligned, indicating that detection does not rely on dataset-specific cues but instead exploits stable statistical properties of the codec latent representation. This behaviour further suggests that LSS generalises well under distribution shifts combined with signal degradation.

6.2. Estimated perceptual quality

Figure 5 shows PESQ distributions for the in-domain scenario T2 and the out-of-domain scenario T3 defined in Table 1. In both cases, distributions for unwatermarked and watermarked utterances largely overlap, indicating only modest quality degradation due to LSS watermarking. Across all configurations, the average drop is below $\Delta\text{PESQ} = 0.2$.

7. Discussion

LSS shows that blind speech watermarking can be achieved effectively in codec latent space by exploiting the anisotropic structure of PCA representations. The method

²We resample 16 kHz data to 24 kHz.

³Code and sample utterances are available at <https://github.com/eurecom-aspl/lss>

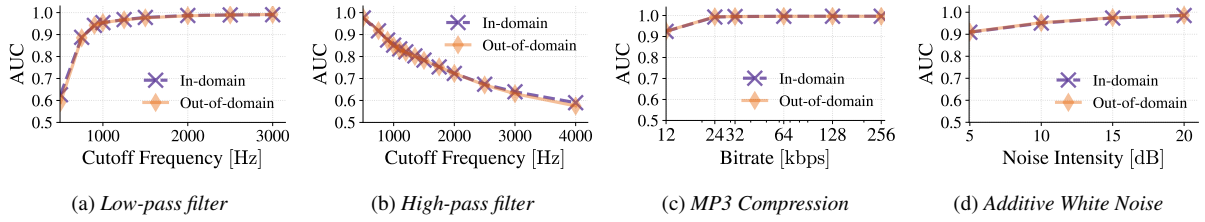


Figure 4: Detection accuracy as a function of manipulation intensity for four types of signal distortions. Purple curves denote the in-domain scenario (T2) and orange curves the out-of-domain setting (T3).

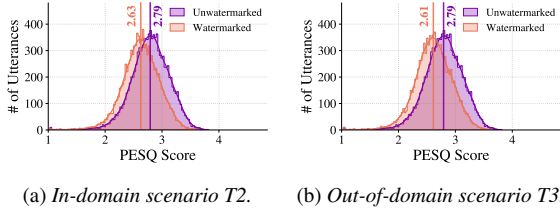


Figure 5: Distribution of PESQ scores for codec-reconstructed speech before (purple) and after (orange) watermark embedding. The annotated values indicate the average PESQ computed over the full evaluation set.

provides reliable keyed detection, generalises well across datasets, and remains robust under common signal manipulations while preserving perceptual quality.

Key strengths of LSS are its simplicity, interpretability, and high degree of flexibility. Unlike fully learned watermarking systems, it does not rely on a dedicated trained embedder-detector pair, but instead operated through small orthogonal rotations in selected PCA planes. The resulting covariance changes are predictable and analytically tractable, making the method easier to reason about in terms of robustness, security and failure modes.

This design also allows the behaviour of LSS to be adjusted through tuning of parameters such as number of planes P or rotation magnitude θ , without retraining. More broadly, this statistical formulation provides a complementary perspective to fully learned watermarking approaches and while not being tied to a specific encoder, suggesting that it could in principle be applied to other suitable latent representation.

Results indicate that LSS remains stable across heterogeneous manipulations, including the residual vector quantization naturally introduced by EnCodec.

In contrast to approaches such as AudioSeal [11], where robustness is learned through exposure to distortions during training, LSS exhibits structural robustness: since watermarking is performed directly in the codec latent space, the watermark is inherently subjected to the same quantisation and compression operations as the underlying representation. Robustness to neural codec compression therefore arises naturally from the embedding space rather than being learned as an external invariance.

Although high-pass filtering remains the most challenging condition, LSS degrades more gracefully than competing methods [11], whose performance varies more strongly across filtering manipulations. This more predictable behaviour is attractive in practical scenarios, where signal post-processing is typically unknown.

This is particularly encouraging because LSS was not extensively optimised: the reported configuration was chosen primarily to validate the geometric principle, suggesting that further improvements may be achieved through

tuning of the embedding schedule and rotation parameters.

LSS also occupies an interesting position with respect to payload design: the embedding depends on payload-specific sign assignments, so that the hidden message structure directly shapes the watermark pattern.

LSS does not impose a fixed payload length a priori. In contrast, in AudioSeal the payload capacity is defined during training and cannot be increased afterward without retraining the model. In addition, unlike AudioSeal, LSS restricts watermark detection to authorised parties holding the secret key: even if the decoder weights are publicly disclosed, detection remains infeasible without access to the key. While in this work we focus on watermark detection, future investigations will explore the possibility of explicit payload recovery.

The presented study has some limitations. Experiments are restricted to bona fide speech and a fixed codec configuration, while robustness is evaluated under common, non-malicious manipulations rather than stronger, adaptive attacks. Since watermarks are distributed at the chunk level, splicing and other, similar temporal manipulations will impact upon detection reliability and require further consideration. In addition, perceptual quality is estimated through only objective metrics rather than subjective listening tests. We leave these limitations and the broader design space of LSS to be examined more thoroughly in future work, particularly under stronger attack models and across alternative latent representations.

8. Conclusions

We introduced LSS, a blind speech watermarking method based on small orthogonal rotations in structured latent spaces. By exploiting the anisotropic geometry of PCA representations, LSS induces a detectable covariance signature that enables reliable keyed detection without requiring a trained embedder or detector. Experiments show that LSS is robust, perceptually transparent, and able to generalise across datasets. Results show that geometric operations in codec latent spaces can provide a stronger and interpretable alternative to fully learned watermarking approaches. LSS therefore offers a promising direction for robust and imperceptible speech watermarking, while paving the way for future work using stronger attack models, richer payload embedding and recovery, and improved schedule design.

9. Acknowledgements

This work was supported by the COMPROMIS project (ANR22-PECY-0011) funded by a French government grant managed by the Agence Nationale de la Recherche under the France 2030 program.

10. References

- [1] L. Cao, "Watermarking for AI Content Detection: A Review on Text, Visual, and Audio Modalities," in *The 1st Workshop on GenAI Watermarking*, 2025.
- [2] M. Charfeddine, E. Mezghani, S. Masmoudi, C. B. Amar, and H. Alhumyani, "Audio watermarking for security and non-security applications," *IEEE Access*, vol. 10, pp. 12 654–12 677, 2022.
- [3] M. Faundez-Zanuy, J. J. Lucena-Molina, and M. Hagsmüller, "Speech watermarking: an approach for the forensic analysis of digital telephonic recordings," *Journal of forensic sciences*, vol. 55, no. 4, pp. 1080–1087, 2010.
- [4] G. Hua, J. Huang, Y. Q. Shi, J. Goh, and V. L. Thing, "Twenty years of digital audio watermarking—a comprehensive review," *Signal processing*, vol. 128, pp. 222–242, 2016.
- [5] W.-N. Lie and L.-C. Chang, "Robust and high-quality time-domain audio watermarking based on low-frequency amplitude modification," *IEEE transactions on multimedia*, vol. 8, no. 1, pp. 46–59, 2006.
- [6] M. Barni and F. Bartolini, *Watermarking systems engineering: enabling digital assets security and other applications*. Crc Press, 2004.
- [7] C. Liu, J. Zhang, H. Fang, Z. Ma, W. Zhang, and N. Yu, "DeAR: a deep-learning-based audio re-recording resilient watermarking," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. [Online]. Available: <https://doi.org/10.1609/aaai.v37i11.26550>
- [8] S. Wen, Q. Zhang, T. Hu, and J. Li, "Robust Audio Watermarking Against Manipulation Attacks Based on Deep Learning," *IEEE Signal Processing Letters*, vol. 32, pp. 126–130, 2025.
- [9] H. Li, Z. Wu, X. Xie, J. Xie, Y. Xu, and H. Peng, "Voice-Mark: Zero-Shot Voice Cloning-Resistant Watermarking Approach Leveraging Speaker-Specific Latents," in *Inter-speech 2025*, 2025, pp. 5108–5112.
- [10] G. Chen, Y. Wu, S. Liu, T. Liu, X. Du, and F. Wei, "Wav-Mark: Watermarking for Audio Generation," 2023.
- [11] R. S. Roman, P. Fernandez, H. Elsahar, A. Défossez, T. Furon, and T. Tran, "Proactive detection of voice cloning with localized watermarking," in *Proceedings of the 41st International Conference on Machine Learning*, ser. ICML'24. JMLR.org, 2024.
- [12] M. Celik, G. Sharma, and A. Tekalp, "Pitch and duration modification for speech watermarking," in *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, vol. 2, 2005, pp. ii/17–ii/20 Vol. 2.
- [13] S. Murata, Y. Yoshitomi, and H. Ishii, "Audio watermarking using Wavelet Transform and Genetic Algorithm for realizing high tolerance to MP3 compression," *Journal of Information Security*, vol. 2, no. 3, p. 99, 2011.
- [14] P. K. Dhar and T. Shimamura, "A DWT-DCT-based audio watermarking method using singular value decomposition and quantization," *Journal of Signal Processing*, vol. 17, no. 3, pp. 69–79, 2013.
- [15] T. D. Hien, Y.-W. Chen, and Z. Nakao, "PCA based digital watermarking," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, 2003, pp. 1427–1434.
- [16] N. Chawla and V. Singh, "A Review of DWT and PCA based Digital Watermarking Schemes," 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:212595206>
- [17] S. Sinha, P. Bardhan, S. Pramanick, A. Jagatramka, D. K. Kole, and A. Chakraborty, "Digital video watermarking using discrete wavelet transform and principal component analysis," *International Journal of Wisdom Based Computing*, vol. 1, no. 2, pp. 7–12, 2011.
- [18] A. Saboori and S. A. H. Hosseini, "A new method for digital watermarking based on combination of DCT and PCA," *2014 22nd Telecommunications Forum Telfor (TELFOR)*, pp. 521–524, 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:9023750>
- [19] M. Tonge, A. Gupta, R. Gandhi, and P. Vishwavidyalaya, "A Survey of Digital Watermarking Techniques," 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:115557554>
- [20] S.-z. Wang, "Watermarking based on principal component analysis," *Journal of Shanghai University (English Edition)*, vol. 4, no. 1, pp. 22–26, 2000.
- [21] S. A. Kahdim and A. M. Abduldaim, "Principal component analysis for zero watermarking technique," *Comput Sci*, vol. 18, no. 1, pp. 85–97, 2023.
- [22] B. Yang, H. Yan, L. Zhang, Q. Yan, Z. Hou, X. Wang, and X. Xu, "Zero Watermarking Algorithm for Hyperspectral Remote Sensing Images Considering Spectral and Spatial Features," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2025.
- [23] S. Wang, W. Yuan, J. Wang, and M. Unoki, "Speech Watermarking Based on Robust Principal Component Analysis and Formant Manipulations," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2082–2086.
- [24] S. Wang, C. Wang, W. Yuan, L. Wang, and J. Wang, "Secure echo-hiding audio watermarking method based on improved PN sequence and robust principal component analysis," *IET Signal Processing*, vol. 14, no. 4, pp. 229–242, 2020. [Online]. Available: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-spr.2019.0376>
- [25] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talmnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 993–1003. [Online]. Available: <https://aclanthology.org/2021.acl-long.80/>
- [26] X. Wang, H. Delgado, H. Tak, J. weon Jung, H. jin Shim, M. Todisco, I. Kukanov, X. Liu, M. Sahidullah, T. Kinnunen, N. Evans, K. A. Lee, J. Yamagishi, M. Jeong, G. Zhu, Y. Zang, Y. Zhang, S. Maiti, F. Lux, N. Müller, W. Zhang, C. Sun, S. Hou, S. Lyu, S. Le Maguer, C. Gong, H. Guo, L. Chen, and V. Singh, "ASVspoof 5: Design, collection and validation of resources for spoofing, deepfake, and adversarial attack detection using crowdsourced speech," *Computer Speech & Language*, vol. 95, p. 101825, 2026. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0885230825000506>
- [27] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High Fidelity Neural Audio Compression," *Transactions on Machine Learning Research*, 2023, featured Certification, Reproducibility Certification. [Online]. Available: <https://openreview.net/forum?id=ivCd8z8zR2>