To appear in:

International Journal of Mathematical Modelling & Computations

Online ISSN: 2228-6233 Print ISSN: 2228-6225

Received: 19- December-2025

Revised: 12- June -2026

Accepted: 13- June-2026

DOI: https://doi.org/10.57647/ijm2c.2027.1701.08

ORIGINAL RESEARCH

Efficient Audio Watermarking via Advanced Signal Processing: EMD–SVD Decomposition and Intelligent Embedding for Secure Multimedia

Yasmin Makki Mohialden[1] Mohammad Mosleh[2*] Jamal N. Hasoon[3] Reihaneh Khorsand[1]

[1]Department of Computer Engineering, Isf.C., Islamic Azad University, Isfahan, Iran
[2]Department of Computer Engineering, Dez.C., Islamic Azad University, Dezful, Iran
[3]Computer Science Department, College of Science,Mustansiriyah University ,Baghdad, Iraq
*Corresponding Author: Mohammad Mosleh (E-Mail: Mohammad.Mosleh@iau.ac.ir)

Abstract. Audio watermarking is an important technique for copyright protection, multimedia authentication, and secure content distribution. However, many existing methods still struggle to achieve a reliable balance among imperceptibility, robustness, and embedding capacity, mainly because watermark insertion is often performed using fixed rules that do not sufficiently reflect the local behaviour of audio signals. This paper proposes an adaptive audio watermarking framework in which each processing stage is designed to support a specific decision in the embedding and extraction pipeline. First, empirical mode decomposition (EMD) is used to decompose each audio frame into intrinsic mode functions (IMFs), providing a signal-adaptive representation of the non-stationary host audio. Then, a 1D convolutional neural network(1D-CNN) extracts representative features from these components. Based on these features, K-Means++ clustering identifies stable and perceptually suitable IMFs with favourable energy and variance characteristics. The watermark is embedded in the SVD domain by modifying the dominant singular values through a 2-bit quantization strategy, which improves payload capacity while preserving audio quality. Finally, an XGBoost classifier learns the selected embedding locations and supports blind watermark extraction. Experiments on four audio genres show that the proposed method achieves an average SNR of 45.1 dB, ODG of −0.28, embedding capacity of 2350 bps, and perfect extraction under no-attack conditions with BER = 0 and NC = 1.0. The method also maintains low BER and high NC under Stirmark and common signal-processing attacks, making it suitable for secure audio distribution and copyright protection.

Keywords: Audio Watermarking, Empirical Mode Decomposition, Singular Value Decomposition, XGBoost Classifier, Copyright Protection.

List of Abbreviations

Abbreviation	Full term
EMD	Empirical Mode Decomposition
SVD	Singular Value Decomposition
IMF	Intrinsic Mode Function
1D-CNN	One-Dimensional Convolutional Neural Network
XGBoost	Extreme Gradient Boosting
SNR	Signal-to-Noise Ratio
BER	Bit Error Rate
NC	Normalized Correlation
ODG	Objective Difference Grade
PEAQ	Perceptual Evaluation of Audio Quality
bps	bits per second

1. Introduction

Watermarking systems, which belong to the significant branches on the information security and ownership, have been advanced on the purpose of safeguarding the data and securing intellectual rights on the property[1, 2]. These systems can be applied to different data like images, audio and video. Of such media, audio watermarking is the media that has drawn the attention of researchers because of the prevalence of audio cues in digital media, as well as the unique issues of audio signal complexity that present difficult challenges. Audio signals are necessarily sensitive to alterations because of their special frequency and dynamic features; thus, it is of great significance to design such a system that allows embedding and extracting the watermark without any obvious alteration in audio quality. Audio watermarking is primarily applicable to secure authenticity, tampering, property rights, which are considered essential issues in the digital environment. The watermarking in audio is mainly applicable in ensuring authenticity, tampering, property rights and these are thought to be critical concerns in the digital world. There are two significant stages of the audio watermarking systems the embedding stage and the extraction stage. The watermark in the embedding section is deliberately placed in some coefficients of the audio signal in a way that it does not leave any significant change in the audio signal that can be perceived by an audio listener. On the other hand, the extraction in the extraction process is performed on the audio signal. The extraction can be performed in two executions; blind and non-blind. In the blind method, it is only necessary to have the signal with the watermark used in watermark recovery and the original reference signal is not needed; in the non-blind method the original reference signal is also needed during the extraction process. In order to measure the performance of the watermarking systems, three criteria are employed in the watermarking systems which include transparency, robustness and capacity. Transparency describes the extent to which the variations within the audio signal are not measurable and is commonly expressed in terms of signal to noise ratio (SNR), and this value should be at least 20 dB. The robustness measures the accuracy of watermark recovery in situations where the watermarks are added with noise and other signal processing attacks such as compression, in other terms, the bit error rate (BER). Also, capacity refers to the amount of the data that can be incorporated in one unit of time that in audio watermarking should not less than 20 bits per second.

One of the main challenges in audio watermarking is to find a good trade-off between transparency, robustness and embedding capacity.

Conventional audio watermarking techniques, such as least significant bit (LSB) [3], echo hiding[4], spread spectrum[5], and phase coding[6], are simple and computationally efficient. However, many of them rely on fixed embedding rules or static modulation parameters, which limits their adaptability to the non-stationary characteristics of audio signals and may weaken the balance among transparency, robustness, and capacity.

Recent learning-based watermarking methods have attempted to improve extraction accuracy by using classifiers such as support vector machine (SVM), K-nearest neighbor(KNN), neural networks, and ensemble models. However, in many cases, learning is mainly applied at the extraction or detection stage, while the embedding process still depends on fixed or weakly adaptive rules. Therefore, an adaptive embedding framework is needed to select suitable watermarking locations according to the intrinsic characteristics of the host audio signal.

This paper describes an adaptive audio watermarking framework that includes signal decomposition, feature-based IMF selection, transform-domain embedding, and intelligent blind extraction. In the proposed method, EMD decomposes the audio signal into IMF components, from which a 1D-CNN-based feature extraction module extracts representative features. KMeans++ clustering is then used to find suitable IMFs for watermark embedding, and SVD-domain 2-bit quantisation boosts embedding capacity. Finally, XGBoost facilitates blind watermark extraction by predicting the desired embedding locations during the extraction stage.

The remainder of this paper is organised as follows. Section 2 discusses related audio watermarking methods, including a comparative analysis and research gap. Section 3 provides a theoretical background for EMD and SVD. Section 4 discusses the proposed embedding and extraction procedures. Section 5 presents the experimental findings, and Section 6 concludes the paper with limitations and future research directions.

2. Related Works

Audio watermarking has been revealed as an essential method of content protection in digital form in recent years. A number of current techniques have been suggested and they have led to enhancements of robustness, imperceptibility and calculational efficiency.

The Empirical Mode Decomposition (EMD) was used at Khaldi and Boudraa (2012) to determine adaptive components to embed and enhance noise and filtering attack robustness[7]. The time-frequency analysis framework proposed by Mandic et al. (2013) is built on the EMD and its variations, the ensemble EMD (EEMD) and multivariate EMD (MEMD)[8]. MEMD enables aligned decomposition of multichannel, nonstationary signals, improving interpretability and synchrony analysis. Practical insights include handling noise, unbalanced channels, and optimizing computational efficiency. Peng et al. (2013) came up with a learning-based audio watermarking scheme that relies on kernel Fisher discriminant analysis (KFDA) to solve the relationships between energies of audio frames to improve resilience when dealing with common signal distortions[9]. Latifpour et al. (2015) designed a K-nearest neighbors (KNN)-based watermarking algorithm based on the wavelet-domain changes in energy with chaotic synchronization, and demonstrated better detection rates and sturdiness[10]. Mohsenfar et al. (2015) combined the QR decomposition with genetic algorithm to make the best embedding locations and still maintain the quality of audio[11]. The hybrid scheme suggested by Mosleh et al. (2016) that combines single value decomposition (SVD) with support vector machines (SVM) is used to retrieve watermark information cleverly amidst noise and distortion[12]. Pourhashemi et al. (2019) used Lucas regular sequences and Fast Fourier Transform (FFT), to optimize energy-level-based embedding to ensure robust watermarking[13]. Abdelwahab et al. (2020) offered a scheme that integrates both SVD and fractional fourier transform (FRT) to improve on the security of embedding at the rotated frequency domain[14]. El-Gazar et al. (2020) implemented the hybrid approach

that uses EMD and SVD, resulting in better robustness and audio quality in speech watermarking[15]. An intelligent and efficient method for extracting watermarks was introduced by Pourhashemi et al. (2020)[16]. The presented method is based on the combination of two powerful techniques: Lifting Wavelet Transform (LWT) and SVM. The fuzzy inference system was designed by Mosleh et al. (2021) in the DCT domain that used energy, zero-crossing rate (ZCR), and music edge attributes to adaptively choose embedding segments[17]. A hybrid extraction system proposed by Pourhashemi et al. (2021) is based on discrete wavelet transform (DWT) and an ensemble of smart classifiers to improve accuracy and strength in detection[18]. Wu et al. (2022) suggested a spectrum-distribution-based scheme modulating the eigenvalue difference between adjacent frequency bands, in which the polarity-based embedding and energy-based synchronization can enable the blind extraction[19]. Alshathri and Hemdan (2023) came up with wavelet-based fusion scheme, Arnold transform, and SVD in order to transmit fused medical images in audio form with a high payload and strong robustness[20]. The calculationally efficient algorithm proposed by Yamni et al. (2023) is based on fractional Charlier transform (FrCT) and dual-tree complex wavelet transform (DTCWT)[21]. A group differential relations-based watermarking scheme was suggested by Lai et al. (2023) to complementary ensemble EMD (CEEMD)-based watermarking, which applies to the average amplitudes of the final IMF. The method is high in imperceptibility and robustness with embedding in low-frequency components, Arnold transform, encryption, and BCH coding[22]. To trade-off watermark resistance and audio quality, Naqash et al(2024). employed iterative filtering with SVD for watermark insertion[23]. Li et al. (2024) proposed IDEAW, a neural watermarking system that has invertible dual-embedding and has an attack-layer simulation to enhance robustness, capacity, and localization efficiency[24]. Table 1 illustrates the comparative summary of related work.

Table 1: Comparative summary of existing audio watermarking methods, highlighting techniques, key features, advantages, and limitations.

Authors (Year)	Domain	Key Features / Method	Advantages	Limitations
Khaldi & Boudraa (2012) [7]	Time(EMD)	Adaptive component selection for embedding	Robust under noise/filter attacks	Mode-mixing; High computational cost
Mandic et al. (2013) [8]	Time-Frequency	Multivariate EMD	Aligned multichannel decomposition; improved synchrony analysis	High complexity
Peng et al. (2013) [9]	Time	KFDA decoder	High robustness to signal distortions	Strong dependence on training data
Latifpour et al. (2015) [10]	Wavelet	KNN+Choas	Improved detection & resilience	Limited scalability to large datasets
Mohsenfar et al. (2015) [11]	Time(QR decomposition)	Optimized embedding positions with GA	Optimal embedding locations	High algorithmic complexity
Mosleh et al. (2016) [12]	Time(SVD)	SVM detector	Robust to noise	Complex model design
Pourhashemi et al. (2019) [13]	FFT	Lucas sequence +Energy-level-based embedding	Energy-based robust mbedding	Requires careful parameter tuning

Abdelwahab et al. (2020) [14]	FRT+SVD	Embedding in rotated frequency domain	Enhanced security	Sensitive to fractional order selection
El-Gazar et al. (2020) [15]	EMD + SVD	Embedding in the first IMF	Robustness while maintaining quality	Moderate computational cost
Pourhashemi et al. (2020)[16]	LWT+SVM	SVM decoder	Improved robustness and transparency	Require carful tunning
Mosleh et al. (2021) [17]	DCT	Fuzzy inference system for segment selection	Balanced imperceptibility and robustness	Fuzzy rules need tuning
Pourhashemi et al. (2023) [18]	DWT	Ensembel decoding	High detection accuracy and robustness	Need large database
Wu et al. (2022) [19]	Frequency spectrum	eigenvalue difference embedding with polarity and energy synchronization	Blind extraction and strong sync	Sensetive to distoration
Alshathri & Hemdan (2023) [20]	Wavelet + SVD	“Wavelet fusion + Arnold scrambling + SVD (medical IoT)	High payload and robustness	High complexity cost
Yamni et al. (2023) [21]	FrCT + DTCWT	Hybrid FrCT + DTCWT embedding	Low processing overhead	Limited experimental scope
Lai et al. (2023) [22]	CEEMD(low-freq IMF)	CEEMD+ Arnold + BCH	High imperceptibility and robustness	Complex process
Naqash et al. (2024) [23]	SVD	Itrative filtering	Improved robustness and transparency	Possible perceptual quality loss
Li et al. (2024) [24]	Neural embedding	IDEAW (dual-embedding)	High capacity and robustness	Heavy traning cost

2.1 A Comparison of Related Watermarking Methods

The reviewed audio watermarking methods demonstrate that different approaches have addressed robustness, imperceptibility, capacity, and extraction accuracy from various angles. Classical methods like LSB, echo hiding, spread spectrum, and phase coding are generally straightforward and computationally efficient, but they frequently rely on fixed embedding rules or modulation parameters. Transform-domain and decomposition-based methods, such as FFT, DWT, DCT, EMD, SVD, and FRT-based schemes, typically improve robustness and transparency in the face of specific attacks; however, many of them continue to rely on manually selected coefficients or predefined embedding locations. This limits their ability to adapt to the non-stationary and content-dependent properties of audio signals.

Learning-based watermarking methods, such as SVM-, KNN-, fuzzy-, neural-network-, and ensemble-based approaches, enhance extraction or detection accuracy through intelligent decision-making. Nonetheless, in many cases, learning is applied primarily at the extraction stage, with the embedding process remaining fixed or only weakly adaptive. More recent neural watermarking methods can improve adaptability and payload capacity, but they may necessitate more training and incur higher computational

costs. As a result, existing approaches continue to struggle with balancing robustness, perceptual transparency, embedding capacity, and computational efficiency.

2.2 Research Gap

The comparative analysis reveals that the main challenge in audio watermarking is to provide a reliable trade-off between the robustness, imperceptibility, embedding capacity and computational efficiency. Increasing the embedding strength or payload capacity will improve capacity and robustness but may also lead to audible distortion and degrade perceptual transparency. Conversely, weak or overly conservative embedding maintains audio quality but often does so at the expense of payload capacity and resistance to signal-processing attacks.

The major gap is that many existing methods do not adapt the embedding process adequately to the intrinsic structure of the host audio signal. Fixed embedding positions and static parameters cannot sufficiently reflect local variations of non-stationary audio. Moreover, the learning-based methods improve the accuracy of extraction, but they usually lack a guidance to select stable and perceptual suitable embedding components. Thus, an adaptive watermarking framework is required to select suitable embedding locations, control the embedding strength, and increase payload capacity without significant degradation of audio quality. This motivates the proposed intelligent EMD–SVD-based watermarking scheme, which aims to improve the balance among robustness, imperceptibility, and embedding capacity through adaptive IMF selection and classifier-assisted blind extraction.

3. Preliminaries

This section provides a rough overview of two basic tools- EMD and SVD- that are applied in the proposed scheme.

3.1 Empirical Mode Decomposition

Empirical mode decomposition (EMD) is a data-driven signal decomposition method that is adaptive and thus ideal in the analysis of non-stationary and nonlinear signals. In contrast to the classical techniques of decomposition that are based upon preset basis functions (e.g. a Fourier or a wavelet), EMD directly identifies intrinsic mode functions (IMFs) using the data itself, and thus offers a better representation of signal properties[8, 25].

The EMD algorithm operates through an iterative process, which includes:

- Identifying local extrema and constructing upper and lower envelopes.
- Computing the mean envelope and extracting an oscillatory mode.
- Verifying IMF conditions (e.g., consistency between extrema and zero crossing).
- Iterative refinement until an IMF is obtained.
- Extracting the residual and repeating the process until the signal is fully decomposed.

Finally, the original signal is represented as the sum of the IMFs and a residual as follows:

$$x(t) = \sum_{i=1}^{n} IMF(i) + r(t) \quad (1)$$

, where IMF(i) represents the i-th intrinsic mode function, and r(t) is the non-ossilatory residual.

3.1. Singular Value Decomposition

Singular Value Decomposition (SVD) is a fundamental mathematical technique extensively utilized in linear algebra, signal processing, and image processing applications. The primary capability of SVD lies in its decomposition of any real $m \times n$ matrix into three distinct matrices, which highlight the inherent structure of the original data[14, 26].

Given a real matrix $A \in \mathbb{R}^{m \times n}$, the singular value decomposition of A is expressed as:

$$A = U\Sigma V^T \quad (2)$$

, where:

- A is the input matrix, U and V are orthogonal matrices, and Σ is the diagonal matrix of singular values.
- U∈Rm×m: orthogonal matrix containing the left singular vectors of A, satisfying $\mathrm{U}^{\mathrm{T}}\mathrm{U} = \mathrm{I}$.
- V∈Rn×n: orthogonal matrix containing the right singular vectors of A, satisfying $\mathrm{V}^{\mathrm{T}}\mathrm{V} = \mathrm{I}$.
- Σ∈Rm×n: a diagonal matrix (with dimensions identical to A) whose non-zero elements are the singular values σ1,σ2,…,σr arranged in descending order:

$$\sigma 1 \geq \sigma 2 \geq \cdots \geq \sigma r > 0 \tag{3}$$

, where σ_1, σ_2, ..., σ_r are the singular values arranged in descending order.

The rank of matrix A is denoted by r, and thus matrix Σ can be represented explicitly as:

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_r \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}_{m \times n} \tag{4}$$

, where r is the rank of matrix A, corresponding to the number of non-zero singular values.

4. Proposed Method

Traditional audio watermarking techniques often struggle to balance imperceptibility, robustness, and embedding capacity particularly when faced with diverse signal processing attacks or perceptual constraints. Transform-domain audio watermarking schemes are considered effective in particular scenarios, but fail to adapt to the properties of audio signals locally. Conversely, time-domain techniques are usually sensitive towards the typical operations including compression and filtering. To overcome these limitations, we propose a hybrid and intelligent audio watermarking framework based on the synergy of soft computing techniques and mathematical transformations, which is able to embed the watermarked data in appropriate locations. In order to increase the efficiency of the extraction process, a classifier is additionally trained to learn the designated locations for watermarking, which increase the retrieval accuracy. This combination effectively increases the unobservability, robustness, and embedding capacity. The core idea is to embed a binary image watermark into stable, perceptually insensitive components of the host audio. The watermark image is first converted into a bitstream. The audio signal is segmented into fixed-size non-overlapping frames, and each frame is decomposed using the EMD tool into several IMFs. Deep features are extracted using a 1D-CNN as feature extraction module. These features are clustered via K-Means++ to identify statistically stable IMFs—those with higher energy and lower variance, suitable for embedding. Once suitable IMFs are identified, the embedding is performed using SVD and a 2-bit quantization mechanism, which increases embedding capacity while maintaining audio quality. Moreover, an XGBoost classifier is trained on the labeled IMF features to learn eligible IMFs for embedding, which are used in the extraction step to determine target IMFs and blindly extract watermarked bits (see Fig. 1).

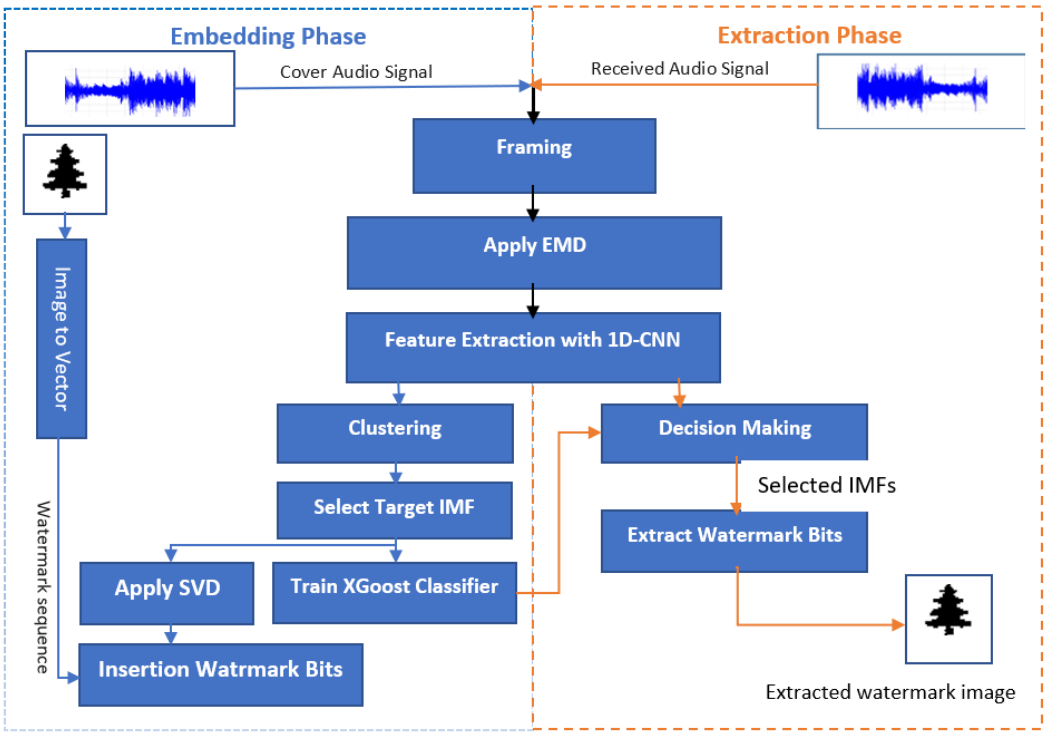

Fig.1: Proposed smart audio watermarking scheme

4.1. Watermark Embedding Process

As discussed in the previous part, the embedding process of the proposed scheme employs an intelligent mechanism to accurately identify suitable locations for watermark insertion, ensuring transparency and robustness. The detailed steps of the embedding procedure are outlined below:

Inputs: Cover audio signal and binary watermark image

Step 1: Convert watermark image to bitstream

The binary watermark image is reshaped into a one-dimentional vector

Step 2: Frame the audio signal

The audio signal is divided into non-overlapping frames

Step 3: Apply empirical mode decomposition

Each frame is decomposed into a signal-dependent set of IMFs using EMD.

Step 4: Feature extraction with 1D-CNN

In this step, the 1D-CNN acts as a feature extraction module to obtain a compact representation of each IMF component. Each IMF sequence is fed into the 1D-CNN, and the output of the last dense layer is used as a 16-dimensional deep feature vector. These feature vectors are used to characterise the local structural and statistical properties of the IMF components and then are used for clustering and embedding-location selection.

Step 5: Clustering with KMeans++ approach

Features are clustered into three clusters using KMeans++ approach

Step 6: Compute energy and variance for each cluster:

Cluster energy is defined as the following :

$$E_c = \frac{1}{|C_c|} \sum_{f \in C_c} \|f\|^2 \quad (5)$$

, where:

- *Ec*: Average energy of cluster *Cc*
- |*Cc*|: Total number of feature vectors in cluster *Cc*
- *f*∈*Cc*: A feature vector belonging to cluster *Cc*
- $\|f\|^2$: Squared Euclidean norm (energy) of vector f

This metric measures the overall signal strength of feature vectors in the cluster. A higher Ec indicates that the cluster contains stronger, more stable IMFs suitable for robust watermark embedding.

In addition, cluster variance is calculated as follows:

$$V_c = \frac{1}{|C_c|}\sum_{f\in C_c}\mathrm{Var}(f) \tag{6}$$

, where:

- Vc: Average variance of feature vectors in cluster Cc
- Var(f): Statistical variance of the elements in feature vector f

This variance metric reflects the internal consistency of feature vectors within the cluster. A lower Vc implies that the feature components are less dispersed, making the cluster more stable and suitable for consistent watermark recovery.

Step 7: Select target cluster

Choose the cluster with the highest energy and the lowest variance as target cluster.

Step 8: Train XGBoost classifier

In the proposed audio watermarking framework, the XGBoost classifier is employed as a supervised decision model to identify the IMFs that are most suitable for watermark embedding and to support blind extraction. After extracting the 16-dimensional feature vectors from the IMF components, KMeans++ clustering is applied with k = 3. The cluster with higher average energy and lower variance is selected as the target cluster. Feature vectors assigned to this target cluster are labelled as “1”, while feature vectors belonging to the two remaining clusters are labelled as “0”. As a result, although the clustering stage uses three clusters, the supervised learning problem is formulated as a binary classification task, namely target IMF versus non-target IMF. The resulting binary-labelled feature matrix is then used to train the XGBoost classifier. In this setting, the input of XGBoost is the 16-dimensional IMF feature matrix, and the corresponding labels are derived from the KMeans++ clustering results based on the energy–variance criterion. During the extraction phase, the trained XGBoost model predicts the target IMFs from the received audio signal, allowing the watermark bits to be recovered without requiring the original host signal.

XGBoost aims to minimize a regularized objective function, which is defined as follows:

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{7}$$

, where l is the loss function (e.g., logistic loss for binary classification), $\hat{y}_i$ is the predicted label, and $\Omega(f_k) = \gamma T + \frac{1}{2}\lambda\|w\|^2$ is the regularization term penalizing the complexity of each tree f_k, with T being the number of leaves and the leaf weights. During training, the model uses both first- and second-order derivatives of the loss to optimize tree splits more efficiently than traditional gradient boosting methods.

Step 9: Insert watermark bits using SVD

In the proposed watermarking scheme, SVD plays a pivotal role in facilitating robust embedding and extraction of watermark bits within the selected IMFs. To this end, the selected IMF vectors are first reshaped into a two-dimensional matrix before applying SVD. The embedding process specifically modifies the two dominant singular values (S1 and S2) of the IMF matrices. Two watermark bits are encoded based on the relative difference parameter D as follows:

$$D = \frac{S_1 - S_2}{S_1 + S_2} \tag{8}$$

, where D is the relative difference between the two dominant singular values S_1 and S_2.

Depending on the watermark bit-pair (2-bit modulation), the value of D is adjusted within specific intervals controlled by parameter α\alpha:

- For bit pair "00": D∈[0,α]
- For bit pair "01": D∈[α,2α]
- For bit pair "10": D∈[2α,3α]
- For bit pair "11": D∈[3α,4α]

Once D is modified accordingly, the new singular values (S1′,S2′) are recalculated by:

$$S_1' = \frac{S_1 + S_2}{2}(1 + D) \quad (9)$$

$$S_2' = \frac{S_1 + S_2}{2}(1 - D) \quad (10)$$

The singular values are modified according to the quantization parameter α and embedded watermark bits. These modified singular values are then employed to reconstruct the modified IMF via inverse SVD:

$$IMF_{\text{Modificd}} = US'V^T \quad (11)$$

, where IMF′ is the reconstructed intrinsic mode function with modified singular values *S*′.

Step 10: Apply inverse EMD

Reconstruct modified frame from adjusted IMFs.

Step 11: Restore watermarked audio signal

Concatenate all modified frames and restore watermarked audio signal.

In addition, the embedding process pseudocode is also provided below:

Pseudocode 1: Embedding phase

Function Embedding

```
INPUTS: audio_signal, watermark_image, parameters
OUTPUTS: watermarked_signal

START
  bitstream ← IMAGE_TO_BITSTREAM(watermark_image)
   frames ← SPLIT_NON_OVERLAPPING(audio_signal, frame_len)
  watermarked_frames ← []
  bit_idx ← 1
  FOR EACH frame IN frames:
        IMFs ← EMD(frame)
        selected_IMFs ← SELECT_IMFs(IMFs, selected_IMF_count)
   END FOR
   features ← []
   FOR EACH imf IN selected_IMFs:
        f ← 1DCNN_EXTRACT(imf)
        APPEND(features, f)
   END FOR
   Clusters ← KMEANS++(features, k)
   FOR c IN Clusters:
        Ec ← CLUSTER_ENERGY(features, labels==c)
        Vc ← CLUSTER_VARIANCE(features, labels==c)
        Score[c] ← RANK(Ec HIGH, Vc LOW)
   END FOR
   target_cluster ← ARGMAX(Score)
```


labeled_features ← {Label all *features* IN *target_cluster* as 1 and remaining as 0}
TRAINXGBOOST(*labeled_features*)
FOR EACH *imf* IN *target_cluster*:
IF *bit_idx* + 2 > LENGTH(*bitstream*): BREAK
M ← RESHAPE_TO_MATRIX(*imf*)
U, S, V ← SVD(*M*)
S_1, S_2 ← SELECT_TOP2(*S*)
b_1b_2 ← bitstream[*bit_idx : bit_idx*+2]
D ← (S_1-S_2) DIV (S_1+S_2)
Bit mapping: "00"→*D*∈[0,α], "01"→*D*∈[α,2α], "10"→*D*∈[2α,3α], "11"→*D*∈[3α,4α]
S'_1, S'_2 ← ADJUST_SINGULARS(S_1, S_2, D, α)
S′ ← REPLACE_TOP2(*S*, S'_1, S'_2)
imf′ ← *U* × *S′* × *V*
REPLACE(*IMFs, imf, imf′*)
bit_idx ← *bit_idx* + 2
END FOR
watermarked_frames ← IEMD(*IMFs*)
watermarked_signal ← CONCAT(*watermarked_frames*)
RETURN *watermarked_signal*
END

4.2 . Watermark Extraction Process

The extraction phase reverses the embedding process by applying EMD, deep feature extraction, and classification to identify watermarked IMFs, enabling accurate recovery of the embedded watermark bits.The details of the proposed scheme are given as follows:

Input:

Received audio signal and trained XGBoost classifier

Step 1: Frame the audio signal

Divide into non-overlapping frames.

Step 2: Apply empirical mode decomposition

Each frame is decomposed into IMFs

Step 3: Feature extraction with 1D-CNN

Use 1D-CNN to extract 16 features from each IMF

Step 4: Select IMFs for extraction

Use trained XGBoost classifier is used to determine the lables of the IMFs. IMFs with label "1" are identified as target IMFs.

Step 5: Reshape selected IMFs to Matrix

Reshape each selected IMFs into the matrix forms.

Step 6: Apply SVD

Decompose the IMF matrix using SVD and then choose two most values of singular values as $S_1^{''}$ and $S_2^{''}$.

Step 7: Calculate relative difference parameter

$$D' = (S_1^{''} - S_2^{''})/(S_1^{''} + S_2^{''}) \quad (12)$$

, where D′ is the computed difference during watermark extraction used to decode the embedded bits.

Step 8: Decode bits from D'

Based on D' value:

D∈[0,α]→00, D∈[α,2α]→01, D∈[2α,3α]→10, and D∈[3α,4α]→11

Step 9: Reconstruct bitstream

Recovering the watermark sequence through bit pairs recovered from frames

Output: Reconstructed watermark image from the extracted watermark sequence.

In addition, extraction pseudocode is present in the following:

Pseudocode 2: Extraction phase

Function Extraction

```
INPUTS: received_audiosignal, parameters
OUTPUT: extracted_image
START
  frames ← SPLIT_NON_OVERLAPPING(received_audiosignal, frame_len)
  FOR EACH frame IN frames:
      IMFs ← EMD(frame)
      selected_IMFs ← SELECT_IMFs(IMFs, selected_IMF_count)
  END FOR
  features ← []
bit_idx ← 1
  FOR EACH imf IN selected_IMFs:
      f ← 1DCNN_EXTRACT(imf)
      IF PREDICT_XGBOOST(f)=target_IMF THEN
          M ← RESHAPE_TO_MATRIX(imf)
          U, S, V ← SVD(M)
          S"1, S"2 ← SELECT_TOP2(S)
          D' ← (S"1-S"2) DIV (S"1+S"2)
      # Bit decoding:
      SWITCH (D)' :
          [0,α] : extracted_stream(bit_idx:bit_idx+2)  ← "00",
          [α,2α]: extracted_stream(bit_idx:bit_idx+2)  ← "01",
          [2α,3α]: extracted_stream(bit_idx:bit_idx+2)← "10",
          [3α,4α] extracted_stream(bit_idx:bit_idx+2) ← "11"
       END SWITCH
      bit_idx←bit_idx+2
  END FOR
  extracted_image ← BITSTREAM _TO_ IMAGE (extracted_stream)
  RETURN extracted_image
END
```

5. Experimental Results and Analysis

All experiments, including audio processing, image embedding, and performance evaluation, were conducted entirely within Colab using Python.

5.1 Dataset Description

For the experimental validation of the proposed audio watermarking method, a curated dataset comprising four distinct audio files and a binary watermark image was used. The audio samples represent four major music genres- Rock, Jazz, Blues, and Classical. Each sample is provided in WAV format, recorded at a sampling rate of 44.1 kHz (see Fig.2). The watermark images used in the experiments, shown in Fig. 3, are a binary images with a resolution of 140×140 pixels. This experimental design guarantees reproducibility,

effective implementation of audio processing and watermark embedding processes as well as gives a standardized setting to test the proposed audio watermarking scheme in relation to various music genres.

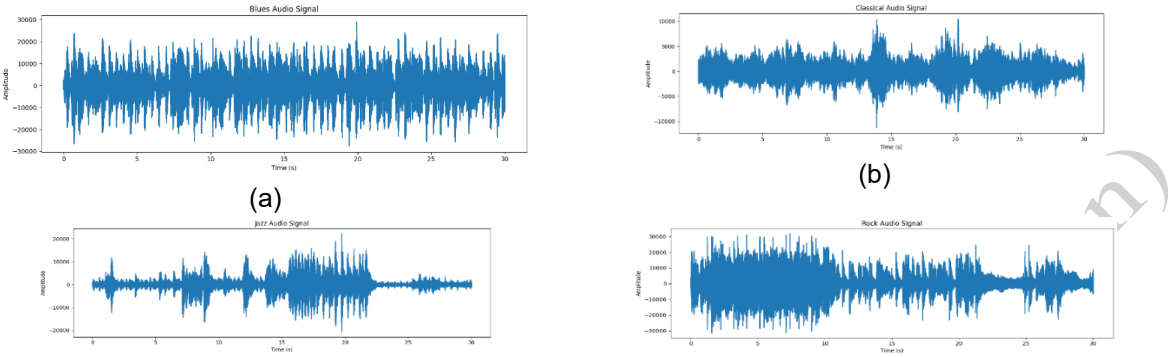

Fig.2: Cover audio signals from different geners: (a) Blues, (b) Classical, (c) Jazz, and (d) Rock

Fig.3: Watermark images

5.2. Evaluation Metrics

In order to measure the performance of the proposed audio watermarking scheme quantitatively, four metrics were used: signal-to-noise ratio (SNR), bit error rate (BER), normalized correlation (NC) and embedding payload.

The SNR can be used to indicate the invisibility of the embedded watermark by the relative powers of the original (cover) audio signal with the added distortion signal and it gives a clue as to how easily the watermark has been embedded. It is defined according to Eq.(13):

$$SNR = 10 \cdot \log_{10}\left(\frac{\sum_{i=1}^{N} S[i]^2}{\sum_{i=1}^{N} (S[i] - \hat{S}[i])^2}\right) \tag{13}$$

, where $S[i]$ and $\hat{S}[i]$ denote the cover and received audio samples, respectively.

Besides SNR, the clarity of the watermarked audio was assessed by Object Difference Grade (ODG), a well known objective measure of perceptual audio quality[27-29]. The ODG derived through the application of the Perceptual Evaluation of Audio Quality (PEAQ) algorithm. The algorithm is formally outlined in the ITU-R BS.1387-1 standard, with measured ODG values ranging from -4 to 0, where -4 indicates Very Annoying (Bad) and 0 represents Imperceptible (Excellent)[30]. The BER, as defined in Eq. (14), serves as a critical metric for assessing the robustness of a watermarking system by quantifying the proportion of incorrectly extracted bits, thereby reflecting the precision and reliability of watermark recovery in the face of various attacks.

$$BER = \frac{1}{M \times M}\sum_{i=1}^{M}\sum_{j=1}^{M}\left(W(i,j) \oplus \widehat{W}(i,j)\right) \times 100 \tag{14}$$

, where W and $\widehat{W}$ are the original and extracted watermark images, and $M \times M$ is the total number of watermark bits.

Normalized Correlation (NC):is a metric used to assess the structural similarity between the original watermark image and the extracted one:

$$NC = \frac{\sum_{i=1}^{M}\sum_{j=1}^{M}\left(W(i,j) \times \widehat{W}(i,j)\right)}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{M} W(i,j)^2} \times \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{M} \widehat{W}(i,j)^2}} \quad (15)$$

, where NC measures similarity between the original watermark *W* and the extracted watermark $\hat{W}$.

Attack Configuration . Two sets of attacks were considered in order to assess the robustness of the proposed audio watermarking method. The first group are the common signal processing attacks widely used in the audio watermarking study, such as additive noise, low-pass filtering, MP3 compression, re-quantization, re-sampling and cropping. These attacks are typical degradations that can occur in audio transmission, compression, storage, format conversion, and local signal manipulation. The second group is a subset of attacks from the StirMark audio watermarking benchmark, for a second benchmark-based robustness evaluation.

The following settings were adopted for the signal-processing attacks. For studying the robustness of the embedded watermark against noise contamination, the watermarked audio signal was corrupted by the additive white Gaussian noise with an SNR level of 49dB in the additive noise attack. To evaluate the robustness of the watermark against spectral information loss, the low-pass filtering attack was employed by using a low-pass filter with a cutoff frequency of 4 kHz to attenuate high-frequency components. The watermarked audio signal was compressed at a bitrate of 64 kbps and decoded back to the waveform domain for evaluating robustness against lossy audio coding in the case of MP3 compression attack. The watermarked signal was re-quantized to 8-bit resolution from 16-bit resolution and re-quantized to 16-bit resolution to simulate quantisation degradation in the re-quantization attack. For the resampling attack, the sampling rate was downsampled from 44.1 kHz to 22.05 kHz and then upsampled to 44.1 kHz to test robustness against sampling-rate conversion. In the cropping attack, 10 samples were picked from each segment of 441 samples and then replaced by adding white noise to assess the impact of local removal and replacement of samples on watermark recovery. In addition to these signal-processing attacks, selected StirMark benchmark attacks have been used to complement the robustness analysis under standardised watermarking attack scenarios.

Embedding payload (Capacity) It measures the number of bits that can be embedded in the host signal per unit of time, and is normally expressed in bits/s (bits per second). It is calculated as:

$$Capacity = \frac{B \times F_s}{N} \quad (16)$$

, where *B* is the average number of embedded bits per frame, *Fs* is the sampling frequency, and *N* is the number of samples per frame.

In the proposed method, *B* depends on the number of target IMFs selected for embedding in each frame. Since two watermark bits are embedded in each selected target IMF, *B* can be expressed as 2 × *Ntarget*, where *Ntarget* denotes the average number of target-cluster IMFs per frame. Therefore, the capacity reflects the number of watermark bits inserted per second according to the proposed frame-level embedding structure.

5.3. Parameter Configuration

Extensive experiments were conducted to study the effect of two main parameters, i.e., modulation gap (α) and the number of selected IMFs used for watermark embedding, to find the best parameter settings of the

proposed audio watermarking framework. Table 2 Selected IMF count Selected IMF count is the number of the IMFs selected for embedding after EMD decomposition, feature extraction, and clustering. This is not necessarily the same as the total number of IMFs created by EMD, because EMD produces a set of IMFs that depends on the signal and is based on the local oscillatory structure of each frame of the audio. In this work, we experimentally evaluated different selected numbers of IMF such as 3, 5, 7 and 9 in terms of SNR, ODG, BER and embedding capacity. The results of these experiments are shown in Table 2.

Table 2: Effect of modulation gap (α) and selected IMFs count on watermarking performance

Modulation Gap	IMF Count	SNR (dB)	ODG	BER(%)	Capacity (bps)
0.01	3	49.5	-0.18	4.6	1320
	5	46.2	-0.31	3.7	2050
	7	40.7	-0.72	6.6	2735
	9	36.8	-1.05	8.9	3320
0.025	3	48.1	-0.23	2.8	1580
	5	**45.1**	**-0.28**	**1.9**	**2350**
	7	39.1	-0.85	6.0	2965
	9	35.4	-1.18	8.4	3525
0.05	3	48.7	-0.21	4.8	1450
	5	45.7	-0.33	4.0	2180
	7	39.9	-0.78	7.3	2855
	9	36.1	-1.12	10.1	3410
0.075	3	49.3	-0.19	6.4	1335
	5	46.4	-0.30	5.8	2030
	7	41.3	-0.68	9.1	2690
	9	37.5	-1.00	11.9	3235
0.1	3	50.1	-0.16	8.3	1225
	5	47.0	-0.27	7.5	1885
	7	42.2	-0.62	10.9	2535
	9	38.4	-0.93	14.3	3050

The analysis of the proposed method with different parameter values shows that the proposed method with different parameter values shows that the overall best performance is achieved with the combination of α=0.025 and selected IMF count=5. This configuration achieved a BER of below 2%, indicating a high accuracy of watermark recovery. Also, the SNR value of 45.1 dB and ODG value of −0.28 confirm the high transparency of the watermarked signal without any noticeable disturbance to the listener. Furthermore, the embedding capacity is 2350 bps, which is a good value when compare to other settings.

Note that k∈[3,9] is the number of IMFs produced by EMD for each audio frame according to the local oscillatory structure of the signal. On the contrary, the selected IMF count mentioned in Table 2 is the number of IMFs used for watermark embedding after the feature extraction and clustering process. The experimental determination of the selected IMF count was done by evaluating different values in terms of SNR, ODG, BER and embedding capacity. The optimal compromise in terms of transparency, extraction accuracy and payload capacity is obtained by selecting five IMFs with α = 0.025 as in Table 2. Other implementation parameters, apart from these parameters, are as follows:

- Frame length: Each audio signal is split into non-overlapping frames of 64 samples.
- CNN architecture: 1D-CNN with two convolutional layers (16 and 32 filters, kernel size = 3, activation = ReLU), batch normalisation, global average pooling and 16-dimensional dense output. This module based on 1D-CNN is used to extract a 16-dimensional deep feature vector from each

IMF component. The obtained feature vectors are considered as the representation space for the next clustering stage.

- KMeans clustering: k=3, with KMeans++ initialisation for stability and convergence. KMeans++ clusters the 16-dimensional IMF feature vectors into three clusters. The average energy and variance are calculated for each cluster, and the cluster with higher energy and lower variance is chosen as the target cluster for watermark embedding. We then use these cluster assignments as labels for the corresponding feature vectors. Thus, a labelled dataset is constructed from the deep IMF features and the corresponding cluster-based labels.
- XGBoost Trainer Parameters The XGBoost classifier was trained globally using the 16-dimensional IMF feature vectors extracted from all tested audio genres and not per file. The feature vectors of the target-cluster, after KMeans++ clustering with k=3, were labelled 1, while the remaining vectors were labelled 0. The classifier was set with 100 estimators, max depth 3, learning rate 0.1, subsample ratio 0.8 and column sampling ratio 0.8. The XGBoost-based embedding-location prediction was evaluated using 10-fold cross-validation before being used for blind extraction.

These parameters were experimentally determined in order to get a compromise between robustness, imperceptibility and embedding capacity.

5.4. Performance Evaluation

The results of evaluating the proposed scheme on four music files with different geners in the unattacked state and under different attacks are presented in Table 3.

Table 3: Performance of the proposed audio watermarking system for four original audio files in the no-attack

Audio File	Genre	SNR (dB)	ODG	BER	NC	Capacity (bps)
Blues	Blues	45.12	-0.28	0	1	2344
Classical	Classical	44.95	-0.30	0	1	2392
Jazz	Jazz	45.07	-0.29	0	1	2360
Rock	Rock	45.26	-0.26	0	1	2304
Average		**45.1**	**-0.28**	**0**	**1**	**2350**

According to the results Table 3, the proposed audio watermarking system has been able to maintain a BER of zero and NC of one in all music genres including Rock, Jazz, Blues and Classical, which indicates a completely accurate watermark extraction. Also, the SNR is in a suitable range between 44.95 dB and 45.26 dB, which indicates high transparency and no degradation of the audio quality of the host audio signal. The watermark embedding capacity is also reported to be 2350 bits per second on average, which is a good value for a watermarking system. Overall, the results in the table show that the proposed system has an ideal performance in the absence of attacks, both in terms of transparency and accuracy of watermark retrieval.

Fig.4 presents the audio signals before and after the embedding process. As illustrated, no noticeable differences can be observed between the original and watermarked signals.

#	Orignal Signal	Watermarked Signal	#	Orignal Signal	Watermarked Signal

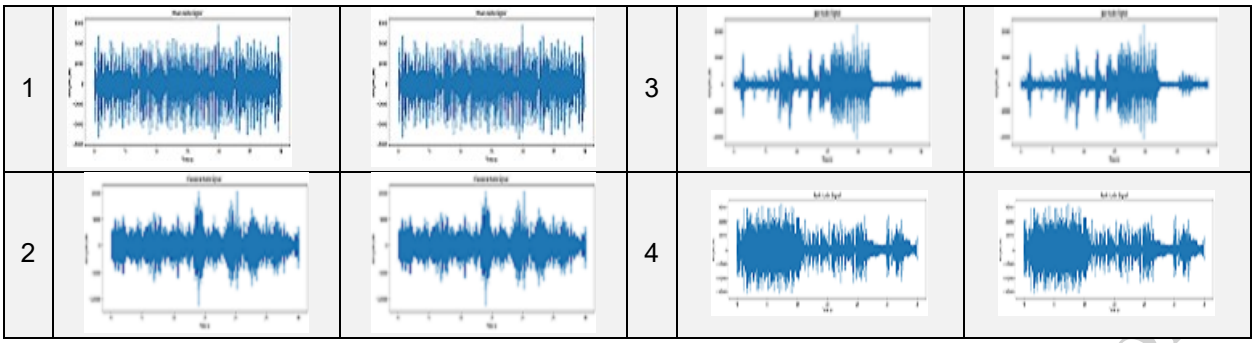

Fig. 4. Tested audio files before and after the embedding process.

In order to evaluate the robustness of the proposed audio watermarking scheme, various attacks were applied to watermarked audio files using Stirmark software, and the results are given in Tables 4 and 5.

Table 4: Evaluation of the proposed scheme on different audio files against Stirmark attacks in terms of BER metric

Attacks	Music Files			
	Blues	**Classical**	**Jazz**	**Rock**
Nothing	0.0	0.0	0.0	0.0
AddBrumm	3.12	2.87	3.45	3.76
Amplify	0.54	0.42	0.48	0.61
Compressor	0.06	0.04	0.03	0.05
ExtraStereo	0.85	0.66	0.73	0.89
FFT_Invert	0.09	0.07	0.08	0.1
Invert	0.0	0.0	0.0	0.0
Flippsample	8.23	7.31	7.89	8.65
LSBzero	0.04	0.03	0.05	0.06
Smooth	0.96	0.82	0.89	1.03
FFT_Real_Reverse	7.42	6.55	6.91	7.78
Exchange	0.88	0.69	0.74	0.91
Stat	5.12	4.36	4.78	5.63
Average	**2.10**	**1.83**	**2.00**	**2.27**

Table 5: Evaluation of the proposed scheme on different audio files against Stirmark attacks in terms of NC metric

Attacks	Music Files			
	Blues	**Classical**	**Jazz**	**Rock**
Nothing	1.0	1.0	1.0	1.0
AddBrumm	0.9783	0.9816	0.9762	0.9751
Amplify	0.9962	0.9974	0.996	0.9951
Compressor	0.9994	0.9993	0.9995	0.9992
ExtraStereo	0.9923	0.9941	0.993	0.9917
FFT_Invert	0.9984	0.9982	0.9983	0.9981
Invert	1.0	1.0	1.0	1.0
Flippsample	0.9327	0.9415	0.9362	0.9286
LSBzero	0.9991	0.9988	0.9987	0.9985
Smooth	0.991	0.9935	0.9924	0.9902
FFT_Real_Reverse	0.9356	0.9438	0.9391	0.931
Exchange	0.9932	0.9951	0.994	0.9927
Stat	0.9528	0.9612	0.9573	0.9497
Average	**0.9822**	**0.9850**	**0.9831**	**0.9808**

Besides the standard tests carried out in the Stirmark benchmark, the proposed audio watermarking scheme has also been severely evaluated against various signal processing attacks to determine its practicality. These attacks consist of additive noise, re-sampling, re-quantization, filtering, MP3 compression, and cropping- all of which are common distortion types in audio transmission, compression and editing. In this case, the quantitative assessment of the method performance is done in terms of the two traditional measures of its efficiency, namely BER and NC. In these assessments of various audio genres, including Blues, Classical, Jazz, and Rock ones, the corresponding results are summarized in Tables 6 and 7. The performance of the method on the basis on these tables shows the consistency in the performance of these methods across several signal types and across different types of attack conditions.

Table 6: Evaluation of the proposed scheme on different audio files against common signal processing attacks in terms of BER metric

Attack	Music Files			
	Blues	**Classical**	**Jazz**	**Rock**
Noise addition	2.87	3.99	1.29	4.54
Cropping	2.94	2.78	1.7	5.93
Low-pass filter	2.52	3.2	1.17	4.31
MP3 compression	3.73	4.88	3.99	1.98
Re-quantization	3.28	3.57	4.04	5.74
Re-sampling	5.16	1.92	3.16	1.7
Average	**3.42**	**3.39**	**2.56**	**4.03**

Table 7: Evaluation of the proposed scheme on different audio files against common signal processing attacks in terms of NC metric

Attack	Music Files			
	Blues	**Classical**	**Jazz**	**Rock**
Noise addition	0.9713	0.960	0.9871	0.9545
Cropping	0.9705	0.9721	0.9829	0.9405
Low-pass filter	0.9747	0.9679	0.9883	0.9568
MP3 compression	0.9625	0.9509	0.96	0.9801

Re-quantization	0.9672	0.9642	0.9595	0.9423
Re-sampling	0.9482	0.9808	0.9683	0.983
Average	**0.9657**	**0.9660**	**0.9744**	**0.9595**

In order to test the strength of the proposed audio watermarking scheme, the watermarked audio files were tested to different signal processing attacks such as additive noise, cropping, low-pass filtering, MP3 compression, re-quantization, and re-sampling. The watermarks were extracted and the result was compared with the original watermark by BER and NC measures. As an example, following additive noise, mean BER in all music genres was 3.17 % and NC was 0.9682, which means that the watermark was not erased significantly. Similarly, under MP3 compression, the BER was 3.65% and NC was 0.9635, demonstrating strong robustness. Overall, the results confirm that the proposed scheme can reliably recover the watermark even after common audio processing attacks, maintaining high fidelity and structural similarity with the original watermark.

The higher BER values obtained under the common signal-processing attacks in Table 6 in contrast to the Stirmark attacks in Table 4 can be attributed to the stronger impact of operations such as re-sampling, re-quantization, cropping and MP3 compression on the temporal and spectral structure of the watermarked signal. Such operations can alter the sample alignment, quantisation levels and frequency content, disturbing the selected IMF components and affecting the singular value based embedding intervals. In contrast, some Stirmark attacks contain controlled benchmark distortions, which do not necessarily change the local IMF structure as much. The maximum mean BER in Table 6 is 4.03%, but the corresponding NC values are high and the extracted watermark still has a high structural similarity with the original watermark. This BER is acceptable for practical ownership verification and DRM-oriented copyright protection where watermark detectability and correlation-based verification are generally sufficient. Future implementations could add an error-correction coding layer for applications requiring near-zero bit recovery.

Table 8 gives a representative comparison of the proposed watermarking scheme—using EMD, SVD and XGBoost with that of various classical, hybrids and intelligent watermarking schemes of existing forms in terms of SNR(dB) and embedding capacity (bps). As it can be seen, the suggested approach reaches the SNR of 45.1 dB and the capacity of 2350 bps, which places it among the competitive performance among both parameters at the same time. This trade-off is one of the major obstacles in audio watermarking as that payload flexibility frequently has the cost of perceptual transparency.A number of classical methods, including EMD [7], IDEAW[24], and DCT[19] show poor results in the two criteria with a low SNR of less than 36 dB and a low capacity of less than 100 bps. Such techniques have low scalability, and they are not effective in the modern high-capacity audio applications. Hybrid approaches give varying levels of trade-offs. As an example, DCT+FUZZY[17] achieves a high SNR of 49.8 dB, meaning superior imperceptibility; however, it operates at a middle level on capacity of 589.3 bps. On the other hand, FFT-based[13] embedding has the advantage of supporting the large range of capacity (1 kbps to 8 kbps) but with a poor SNR stability (33 to 58 dB), and hence, its audio quality under various circumstances may be a concern. Interestingly, DWT+ENSEMBLE [18] and LWT+SVM[16] approaches aim at a trade-off and have given the SNR values of 43.82 dB and 42.65 dB with capacities of 1225 bps and 900 bps, respectively. Despite that, they are not up to the quality of the data payload of the proposed scheme.Other intelligent-based schemes, such as KNN+DWT [10] and SVD+SVM[12] have moderate to good SNR in the range of 41.17 dB and 42.07 dB, respectively, but are either limited in capacity or not robust against high payloads.

Table 8: Comparing the proposed audio watermarking scheme with related works in terms of SNR and capacity criteria

Method	**SNR(dB)**	**Capacity(bps)**
EMD[7]	25.41	46.9
KNN+DWT[10]	41.17	1600
QR+GA[11]	25.89	159
SVD+SVM[12]	42.07	172.41
FFT[13]	33 to 58	1k to 8k
LWT+SVM[16]	42.65	900
DCT+FUZZY[17]	49.8	589.3

DWT+ENSEMLE[18]	43.82	1225
DCT[19]	26.86	64
FrCT+DTCWT[21]	32.12	91.19
IDEAW [24]	35.41	56
IF-SVD[23]	40.05	600
Proposed (EMD+SVD+XGBoost)	45.1	2350

5.5. Discussion on Computational Complexity

The computational cost of the proposed method should be interpreted in relation to its adaptive embedding strategy. The proposed framework is computationally more demanding than traditional rule-based watermarking methods, because of the usage of EMD decomposition, IMF-level feature extraction, SVD-domain embedding and XGBoost-assisted target-IMF prediction. Yet this extra cost is introduced to improve the quality of embedding-location selection. The proposed method selects stable and perceptually suitable IMF components to improve the robustness of the watermark and reduce the perceptual distortion on the host audio signal.

In terms of computational complexity, the presented framework is in an expected range compared to recent hybrid and intelligent watermarking methods discussed in this study, as many recent methods also rely on decomposition methods, transform-domain processing, optimisation, or machine-learning-based decision mechanisms. Moreover, the KMeans++-based label generation and XGBoost training are conducted offline at the configuration stage, and only the trained model is used for target-IMF prediction during the operational phase. Therefore, the main online computational costs involve EMD decomposition, IMF-level feature extraction, SVD operations, and XGBoost inference.

While the method proposed is more complex than simple classical rule-based schemes, this complexity is mainly introduced to improve embedding-location selection, robustness and imperceptibility. Compared to recent hybrid and intelligent watermarking methods, the computational cost is in a comparable range, while the proposed method provides improved embedding stability and increased payload capacity by adaptive IMF selection and multi-bit SVD-domain embedding.

6. Conclusion and future works

This paper suggested an effective audio watermarking algorithm using advanced signal decomposition algorithms and soft computing which could enhance the key issue of the audio watermarking systems which is the tradeoff, between transparency, robustness and embedding capacity. In this method, the decomposition of the audio signal was carried out into intrinsic mode functions describing the local signal behavior with the advanced EMD signal decomposition tool. Then, deep statistical features were extracted from these functions using an 1D-CNN, and suitable embedding locations were identified using KMeans++ clustering. Later, the watermark embedding operation was done by applying 2-bit quantization modulation to the dominant singular values of the selected components by using SVD and maintained the perceptual quality of the signal. To properly identify the embedding locations in the extraction process, an XGBoost model was trained by using the labeled feature acquired via clustering. Its performance on the evaluation test indicated that the scheme offered high transparency, complete watermark reconstruction under clean conditions (i.e BER = 0%, NC = 1) and high resistance to common signal processing attacks such as additive noise, filtering, MP3 compression, re-sampling, re-quantization, and cropping, with the maximum average BER remaining at 4.03% and the average NC values remaining high across the tested audio genres.

The obtained results revealed that the proposed scheme can achieve an SNR of 45.1 dB and an embedding capacity of 2350 bps which is competitive when compared with many classical, hybrid and intelligent watermarking techniques while maintaining a good trade-off between perceptual transparency and payload capacity.This demonstrated its high performance to enable preserving high level of audio quality with embedded payloads, and it is a good candidate to be applied in practice of audio watermarking. The proposed method obtains promising results, but still some limitations exist. The multi-stage structure of EMD, 1D-CNN, KMeans++, SVD and XGBoost increases computational complexity which may limit its real-time and IoT-based deployment. Moreover, the severe synchronisation and desynchronisation attacks can disturb the frame

alignment and embedding locations. This indicates the necessity of more robust synchronisation mechanisms in the future work.

In future work, the proposed framework can be augmented with the inclusion of synchronisation and resynchronisation mechanisms to improve its robustness against severe desynchronisation attacks. Moreover, lightweight architectures can be designed to reduce the computational cost and support real-time and IoT based applications. The robustness of the method against deep-learning based manipulations, adversarial attacks and GAN-generated deepfake audio can be further studied. Further, reinforcement learning can be explored to enable adaptive embedding strategies that dynamically trade-off transparency, robustness and the payload capacity.

Authors contributions

Authors have contributed equally in preparing and writing the manuscript.

Availability of data and material

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Competing interests

The authors declare that there is no conflict of interest regarding the publication of this paper.

Funding statement

No funding was received for this research.

REFERENCES

[1] Mosleh M, Setayeshi S, Mosleh M. Presenting a Novel Audio Watermarking Based on Discrete Wavelet Transform. *Int J Comput Electr Eng*. 2011;3(3). doi: https://doi.org/10.7763/IJCEE.2011.V3.348

[2] Dhar PK. A blind audio watermarking method based on lifting wavelet transform and QR decomposition. In: *8th International Conference on Electrical and Computer Engineering*. Dhaka, Bangladesh: IEEE; 2014:136-139. doi: https://doi.org/10.1109/ICECE.2014.7026978

[3] Liu J, He X. A review study on digital watermarking. In: *2005 International Conference on Information and Communication Technologies*. Karachi, Pakistan: IEEE; 2005:337-341. doi: https://doi.org/10.1109/ICICT.2005.1598603

[4] Cho JW, Park HJ, Huh Y, Chung HY, Jung HY. Echo watermarking in sub-band domain. In: *Digital Watermarking: Second International Workshop, IWDW 2003*. Seoul, Korea: Springer; 2004:447-455. doi: https://doi.org/10.1007/978-3-540-24624-6_37

[5] Ko BS, Nishimura R, Suzuki Y. Time-spread echo method for digital audio watermarking. *IEEE Trans Multimedia*. 2005;7(2):212-221. doi: https://doi.org/10.1109/TMM.2005.843352

[6] Bassia P, Pitas I, Nikolaidis N. Robust audio watermarking in the time domain. *IEEE Trans Multimedia*. 2001;3(2):232-241. doi: https://doi.org/10.1109/6046.923822

[7] Khaldi K, Boudraa AO. Audio watermarking via EMD. *IEEE Trans Audio Speech Lang Process*. 2012;21(3):675-680. doi: https://doi.org/10.1109/TASL.2012.2227734

[8] Mandic DP, Rehman NU, Wu Z, Huang NE. Empirical mode decomposition-based time-frequency analysis of multivariate signals: The power of adaptive data analysis. *IEEE Signal Process Mag*. 2013;30(6):74-86. doi: https://doi.org/10.1109/MSP.2013.2267931

[9] Peng H, Li B, Luo X, Wang J, Zhang Z. A learning-based audio watermarking scheme using kernel Fisher discriminant analysis. *Digit Signal Process*. 2013;23(1):382-389. doi: https://doi.org/10.1016/j.dsp.2012.08.004

[10] Latifpour H, Mosleh M, Kheyrandish M. An intelligent audio watermarking based on KNN learning algorithm. *Int J Speech Technol*. 2015;18:697-706. doi: https://doi.org/10.1007/s10772-015-9298-0

[11] Mohsenfar SM, Mosleh M, Barati A. Audio watermarking method using QR decomposition and genetic algorithm. *Multimed Tools Appl*. 2015;74:759-779. doi: https://doi.org/10.1007/s11042-013-1720-x
[12] Mosleh M, Latifpour H, Kheyrandish M, Mosleh M, Hosseinpour N. A robust intelligent audio watermarking scheme using support vector machine. *Front Inf Technol Electron Eng*. 2016;17:1320-1330. doi: https://doi.org/10.1631/FITEE.1500349
[13] Pourhashemi SM, Mosleh M, Erfani Y. Audio watermarking based on synergy between Lucas regular sequence and Fast Fourier Transform. *Multimed Tools Appl*. 2019;78:22883-22908. doi: https://doi.org/10.1007/s11042-019-7574-0
[14] Abdelwahab KM, Abd El-atty SM, El-Shafai W, El-Rabaie S, Abd El-Samie FE. Efficient SVD-based audio watermarking technique in FRT domain. *Multimed Tools Appl*. 2020;79:5617-5648. doi: https://doi.org/10.1007/s11042-019-08279-x
[15] El-Gazar S, El-Dolil S, Abbas AM, Dessouky MI, El-Rabaie ESM, El-Dokany IM, et al. Speech Watermarking using a Hybrid Strategy of both Empirical Mode Decomposition and Singular Value Decomposition. *Menoufia J Electron Eng Res*. 2020;29(1):39-49. doi: https://doi.org/10.21608/mjeer.2020.53103
[16] Pourhashemi SM, Mosleh M, Erfani Y. Presenting an intelligent extraction method in audio watermarking systems based on lifting wavelet transform and support vector machine. *J Soft Comput Inf Technol*. 2020;9(1):34-47.
[17] Mosleh M, Setayeshi S, Barekatain B, Mosleh M. A novel audio watermarking scheme based on fuzzy inference system in DCT domain. *Multimed Tools Appl*. 2021;80:20423-20447. doi: https://doi.org/10.1007/s11042-021-10649-x
[18] Pourhashemi SM, Mosleh M, Erfani Y. A novel audio watermarking scheme using ensemble-based watermark detector and discrete wavelet transform. *Neural Comput Appl*. 2021;33:6161-6181. doi: https://doi.org/10.1007/s00521-020-05376-5
[19] Wu Q, Ding R, Wei J. Audio watermarking algorithm with a synchronization mechanism based on spectrum distribution. *Secur Commun Netw*. 2022;2022:2617107. doi: https://doi.org/10.1155/2022/2617107
[20] Alshathri S, Hemdan EED. An efficient audio watermarking scheme with scrambled medical images for secure medical internet of things systems. *Multimed Tools Appl*. 2023;82:20177-20195. doi: https://doi.org/10.1007/s11042-022-14188-x
[21] Yamni M, Daoui A, Karmouni H, Sayyouri M, Qjidaa H, Motahhir S, et al. An efficient watermarking algorithm for digital audio data in security applications. *Sci Rep*. 2023;13:18432. doi: https://doi.org/10.1038/s41598-023-45049-y
[22] Lai WH, Chou TY, Chou MC, Schuller BW. Robust Audio Watermarking based on empirical mode decomposition and group differential relations. *J Audio Eng Soc*. 2023;71(3):100-117. doi: https://doi.org/10.17743/jaes.2022.0077
[23] Naqash KI, Malik SA, Parah SA. Robust audio watermarking based on iterative filtering. *Circuits Syst Signal Process*. 2024;43:348-367. doi: https://doi.org/10.1007/s00034-023-02478-6
[24] Li P, Zhang X, Xiao J, Wang J. IDEAW: Robust Neural Audio Watermarking with Invertible Dual-Embedding. *arXiv*. 2024;2409.19627. doi: https://doi.org/10.48550/arXiv.2409.19627
[25] Dutta AK, Lall B, Joshi SD. Empirical mode decomposition techniques: A simulated review. In: *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. Delhi, India: IEEE; 2023:1-5. doi: https://doi.org/10.1109/ICCCNT56998.2023.10306989
[26] Weiss S, Proudler IK, Barbarino G, Pestana J, McWhirter JG. On properties and structure of the analytic singular value decomposition. *IEEE Trans Signal Process*. 2024;72:2260-2275. doi: https://doi.org/10.1109/TSP.2024.3390549
[27] Kabal P. An examination and interpretation of ITU-R BS. 1387: Perceptual evaluation of audio quality. *TSP Lab Tech Rep*. McGill University; 2002:1-89.
[28] Chen CJ, Huang HN, Tu SY, Lin CH, Chen ST. Digital audio watermarking using minimum-amplitude scaling on optimized DWT low-frequency coefficients. *Multimed Tools Appl*. 2021;80:2413-2439. doi: https://doi.org/10.1007/s11042-020-09696-3

[29] Salah E, Narima Z, Khaldi A, Redouane KM. Survey of imperceptible and robust digital audio watermarking systems. *Multimed Tools Appl*. 2025;84:3635-3681. doi: https://doi.org/10.1007/s11042-024-19375-6

[30] Karajeh H, Khatib T, Rajab L, Maqableh M. A robust digital audio watermarking scheme based on DWT and Schur decomposition. *Multimed Tools Appl*. 2019;78:18395-18418. doi: https://doi.org/10.1007/s11042-019-7178-8