International Journal of Basic and Applied Sciences, 14 (6) (2025) 29-37

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/0h4c8497
Research paper

Secure Audio Watermarking Using Randomized Timestamps and Encrypted Metadata

Ashish Dixit [1, 2] *, Avadhesh Kumar Gupta [3], Veena Bharti [4], Divya Midhunchakkaravarthy [5], Deepak Gupta [6]

[1] Post Doctoral Fellowship (Computer Science& Engineering) Lincoln University College, Malaysia
[2] Associate Professor (Computer Science& Engineering), Ajay Kumar Garg Engineering College, Ghaziabad, U.P., India
[3] Professor, School of Computer Science and Engineering, IILM University, Greater Noida (U.P.) – India
[4] Associate Professor (Computer Science) Raj Kumar Goel Institute of Technology, Ghaziabad, U.P., India
[5] Professor & Director, School of AI Computing and Multimedia, Lincoln University College, Selangor, Malaysia
[6] Assistant Professor, Maharaja Agrasen Institute of Technology, Delhi, India
**Corresponding author E-mail: ashishdixit1984@gmail.com*

Received: July 6, 2025, Accepted: August 17, 2025, Published: October 3, 2025

Abstract

A robust watermarking system for strengthening audio against unauthorized access and tampering is proposed in this paper. The system can ensure robustness and imperceptibility by embedding watermarks to random time instants in the frequency domain with the help of FFT and encrypting the metadata by AES encryption. Thorough evaluations based on SNR, MOS, and extraction accuracy confirm ro-robustness of the system against common attacks such as noise, compression, and resampling. The novel system is applicable in the field of digital rights management by offering a trustworthy and efficient solution for audio authentication and copyright protection.

Keywords: *Audio Watermarking; Randomized Timestamps; Fast Fourier Transform (FFT); Metadata Encryption; Signal-to-Noise Ratio (SNR); Bit Error Rate (BER).*

1. Introduction

The exponential expansion of digital audio content changed our relationship with music, podcasts, and audiobooks. It has been provided to the world the unparalleled access to limitless amounts of sounds unhindered. But at the same time, this convenience has made it easier for easy access to widespread copyright violation and illegal distribution, and thus the need for strong intellectual property protections [1]. Audio watermarking is an interesting approach to these problems, embedding signature-like markers into audio data that can be extracted later for authentication or ownership verification [2]. Audio watermarking is based on the pioneering studies in the area is where embedding techniques were first implemented. In this sense, watermarking is a form of capacity to detectable insert data into a parent audio signal (also referred to in this filed as a watermark) and it does so turn allowing a content provider to track the content which is not only important to content management and providing access to law enforcement applications where there is a threshold value to the authenticity and integrity of the audio in question. Although concomitant progress has been achieved in watermarking schemes, watermarked audio signals remain susceptible to various attacks intended for the distortion or removal of the superimposed information [3]. Attacks could be either intentional or incidental. Indeed, attackers exist with malicious intentions in compromising the watermark; most attackers would presum-ably alter the signal in processing it without actual malicious intention. Considering such factors, an audio watermarking approach is evaluated as successful based on five parameters: imperceptibility, robustness, security, capacity, and computational complexity [8]. Here, imperceptibility and robustness are fundamental in the performance assessment. Considering all these challenges, the research of this paper will be guided in the direction of proposing a new audio watermarking scheme that is more robust and secure for law enforcement usage of digital audio data. The scheme embeds watermarks at randomly chosen time-stamps and employs FFT for frequency domain inspection, and has minimum audio quality imperceptibility. Additionally, the metadata of the watermark information is encrypted in a way that prevents unauthorised access. The paper contribution is to present the idea that the proposed approach is indeed effective in safeguarding audio evidence from unauthorised usage and manipulation, and therefore adds integrity to the law enforcement process.

2. Related Work

Numerous techniques for audio watermarking have been developed over the past decades, driven by the need to ensure copyright protection, authenticity, and integrity of digital audio content. Early foundational work in this field, such as the surveys, laid the groundwork for secure watermarking by proposing robust models against tampering and signal degradation. Time domain methods, such as Least Significant Bit (LSB) substitution and echo-based watermarking, offer simple embedding schemes but tend to lack robustness against compression and noise. Conversely, transform domain techniques—especially those utilizing the Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), and Discrete Wavelet Transform (DWT)—have demonstrated greater imperceptibility and resistance to attacks, as highlighted. Hybrid approaches integrating Singular Value Decomposition (SVD) [3] and genetic algorithms, as discussed, further enhance watermark resilience. Additionally, modern schemes incorporating Spread Spectrum (SS) and Quantization Index Modulation (QIM) have shown potential in maintaining watermark fidelity under hostile conditions. Machine learning techniques have also started to influence the field, with studies such as exploring the application of principal component analysis (PCA) and hybrid models to strengthen embedding and extraction processes [1]. Furthermore, the evolution toward metadata encryption and secure timestamp handling [2], such as described in this study, aligns with trends seen. which advocate embedding supplementary data securely using encryption standards like AES to prevent unauthorized access [4]. While several techniques focus on either robustness or imperceptibility, few integrate a comprehensive security model combining randomized embedding, frequency-based watermarking, and metadata encryption. This proposed method fills this gap by emphasizing both resilience to signal processing attacks and protection against unauthorized metadata access, contributing to the advancement of secure digital audio authentication frameworks.

3. Methodology

An audio watermarking technique is fundamentally important in embedding information into digital audio files. It offers very robust solutions for copyright protection and ownership verification, especially in legal contexts. It makes it possible to embed a watermark into an audio signal imperceptibly as regards its integrity and authenticity.

3.1. Concept of audio watermarking

a) Audio Watermarking Principles

There are two main operations central to audio watermarking: embedding and detection. Listed below are a select few of the requirements of the operation:

Imperceptibility: The watermark is to be imperceptible to the listener. The quality of the tape cannot be changed.

Robustness: It is against distortion induced by different audio processing actions (compressions, noise addition) as well as intentional attacks.

Security: Watermark and corresponding metadata will be kept confidential and not released into the public domain wrong hands without detection and alteration accordingly [5].

Capacity: The algorithm has the capacity to support different watermarks without sacrificing the quality of the sound. Computational Complexity: The algorithm must be efficient enough that it is possible to utilize it in real-time applications.

b) Audio Watermarking

The watermarking embedding strategies can be broadly classified according to the type of techniques used as follows: Time Domain methods and Transform Domain methods [6]. Time-domain methods. These compromise the time domain of the sound itself, so maybe that's more susceptible to whatever modifications. Domain Transform Methods: These are the FFT and DFT of the waveforms of the sounds [7]. These could provide better robustness and imperceptibility.

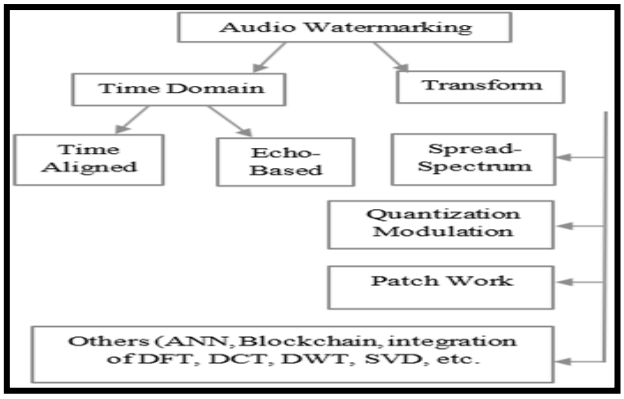

Fig. 1: Audio Watermarking Categorization[6].

c) Fast Fourier Transform (FFT) and Its Role in Audio Watermarking

FFT is a very efficient algorithm for transforming a signal from its time domain to a representation of the signal in the frequency domain. This allows for the analysis of the myriad different frequency components contained within an audio signal, making it very widely used in domains such as audio watermarking. The time complexity with which FFT works is O(NlogN), which allows it to quickly process large files of audio, something that is always desired in real-time applications. In the watermarking process, the original audio signal x[n] is passed through an FFT to generate its representation in the frequency domain[9], X[k]. The watermark signal w[n] is further added to X[k] by using the following equation:

$$Y[k]=X[k]+\alpha W[k] \quad (1)$$

After embedding, the Inverse FFT is applied to convert the modified frequency data back to the time domain, resulting in the final watermarked audio signal y[n]:

$$y[n]=x[n]+\alpha w[n] \quad (2)$$

This process ensures that the watermark is embedded in less perceptible frequency regions, enhancing its robustness against potential attacks and maintaining the audio quality.

d) Mathematical Modelling for the Watermarking

In this watermarking technique, watermark embedding[9] at randomly chosen timestamps enhances robustness and security. This is mathematically expressed as follows:

Time Domain Embedding:

$$y[n] = x[n] + \alpha w[n] \quad (3)$$

y[n] = watermarked signal
x[n] = original audio signal
w[n] = watermark signal
α = watermark strength.

Transform Domain Embedding:

$$Y[k]=X[k]+\alpha W[k] \quad (4)$$

Where Y[k], X[k], and W[k] are the Fourier transforms of the watermarked signal, original signal, and watermark signal, respectively.

Fig. 2 represents the steps involved in the watermark embedding process, from loading the original audio to producing the final watermarked output. in Figure 2, the entire embedding process—from original audio preprocessing, timestamp generation, and watermark insertion to final signal reconstruction—is streamlined into one clear pipeline.

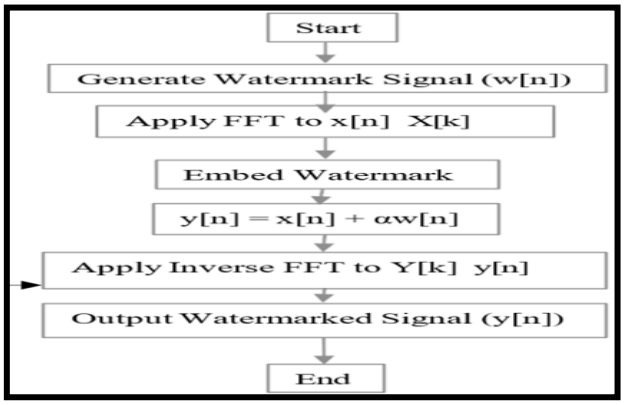

Fig. 2: Watermark Embedding Process [1] [9].

3.2. Metadata encryption and secure storage

The symmetric AES-128 encryption used in this scheme relies on a pre-shared secret key, derived using SHA-256 from a user-provided passphrase. The metadata—containing timestamp vectors, watermark identifiers, and algorithm flags—is serialized in JSON format before encryption. examples in JSON

```
{
"timestamps": [2.3, 5.6, 9.1],
"watermark_strength": 0.04,
"algorithm_id": "FFT-RS"
}
```

The encrypted output is embedded using ID3v2 tags (for MP3) or XMP metadata block (for WAV), ensuring format compatibility.

3.3. Random timestamp selection and secure metadata handling

a) Randomized Timestamp Selection

The other significant feature of this Watermarking Technique is the randomness associated with choosing timestamps for embedding the watermark[10]. In this regard, it requires selection based on a certain algorithm and input in the form of a secret key to ensure the positions of the watermark are unpredictable and can only be determined with the use of the secret key and algorithm.

Algorithm:

Randomized Timestamp Selection for Watermark Embedding Input:

1) Audio Length: Length of the audio signal in seconds or samples.
2) Num watermarks: Number of watermarks to embed.
3) Secret key: Unique key for seeding the random number generator.

4) Min Gap: Minimum allowable gap between consecutive watermarks.
Randomized Timestamp Selection for Watermark Embedding Output:
1) Timestamps: List of random timestamps for watermark embedding.
Steps:
1) Initialize:
a) Convert the secret key into a seed for the random number generator.
b) Initialize an empty list timestamp to store selected random timestamps.
2) Generate Timestamps:
While the number of elements in timestamps is less than num_watermarks:
a) Generate a random timestamp within the range [0, audio_length].
b) If the timestamp is at least min_gap away from all existing timestamps in the list, add it to timestamps.
c) If not, discard and generate a new timestamp.
3) Sort Timestamps: Sort the timestamps list in ascending order to maintain sequence.
4) Output: Return the timestamps list containing the positions for embedding watermarks.
Pseudocode: Algorithm Randomized_Timestamp_Selection
Input: audio_length, num_watermarks, secret_key, min_gap
Output: timestamps
1) Initialize the timestamps ← empty list
2) Seed the random number generator with secret_key
3) While the length of timestamps < num_watermarks do
- Generate random_timestamp ← Random(0, audio_length)
- If random_timestamp is at least min_gap away from all timestamps in the timestamps list, then
- Add random_timestamp to timestamps
4) Sort timestamps in ascending order
5) Return timestamps. End Algorithm
b) Metadata Encryption and Secure Storage
Since the algorithm applied for generating these pseudorandom timestamps for watermark embedding, the data about the above timestamps and their corresponding watermark signatures[11] need to be securely stored. This can be achieved by encrypting metadata and associating it with an audio file, such that it is accessed only by authorized users along with the correct secret key. Here is how this is included in the whole scheme of watermarking:
1) Metadata content
The Metadata includes:
a) Timestamps: Randomly chosen timestamps at which the watermark is impressed.
b) Watermark Information: Information about the watermark signals being impressed at each timestamp, including their strength (α) and other transformation parameters.
c) Algorithm Identifier: Information concerning the exact algorithm followed for the selection of the timestamps, which makes the selection reproducible.
2) Metadata Encryption
To lock the metadata, a symmetric encryption method is used. This procedure will involve the use of a secret key to select the timestamps. It will thus prevent unauthorized access to the metadata by those who do not have the secret key.
Encryption Step: Convert metadata this case, timestamps and watermarked information into a structured format, or JSON or XML.
a) Apply symmetric encryption to the metadata using a secret key, applying an AES algorithm.
b) Save the encrypted metadata as part of the audio file's metadata section.
3) Embedding Encrypted Metadata:
a) The audio file's metadata section holds the encrypted metadata. The embedding is non-intrusive and doesn't affect the quality or content of the audio.
b) The metadata can be saved in any format suitable for the desired type of audio file, for example, ID3 tags for MP3, XMP metadata for WAV.
4) Extraction and Decryption:
a) During the watermark verification process, the encrypted metadata is extracted from the audio file.
b) Using the same secret key, the metadata is decrypted to retrieve the timestamp and watermark information.
c) The algorithm and secret key are then used to locate and verify the watermarks embedded at the specified timestamps.
5) Mathematical Representation:
Let's denote the metadata as M, which consists of the selected timestamps T $\{t_1, t_2 \ldots, t_n\}$ and the watermark information $W=\{w_1, w_2, \ldots, w_n\}$ The secret key is represented as K. The encryption of metadata can be expressed as:

$$M_{encrypted} = Encrypt(M, K) \quad (5)$$

Where:
- M= {T, W, Algorithm ID} is the metadata set.
- Encrypt(M,K) is the symmetric encryption function.

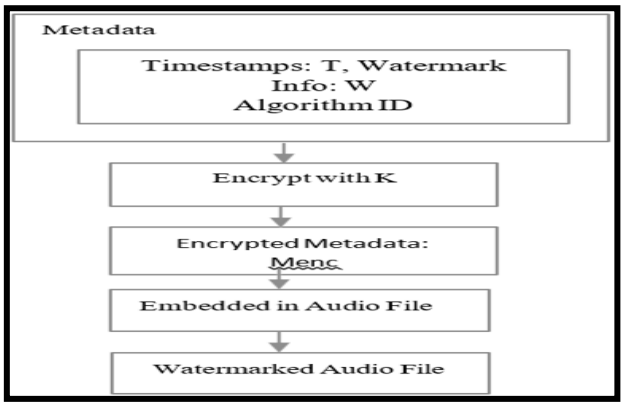

Fig. 3: Watermark Embedding Process [9] [13].

c) Watermark Embedding and Extraction Process

The embedding/extraction process is the heart of an audio watermarking system, since it allows the integrity of the watermark to be preserved and validated. This section outlines the embedding and extraction processes, clearly stressing the mathematical basis of the technique, and the application of the secret key as well as the robustness of the design.

1) Watermark embedding process

The embedding process embeds an original watermark into the original audio signal at arbitrary random timestamps [12]. In this methodology, the security and robustness of the watermark increase to get through various attacks as well as modifications. The details of the steps involved are as follows.

Step 1: Input Preparation:

a) Original Audio Signal: Let x[n] denote the original audio signal, which is typically represented as a discrete sequence of samples.
b) Watermark Signal: Define the watermark signal w[n], which contains the information to be embedded, such as a unique identifier or copyright information.
c) Random Timestamp Generation: Utilize a secure algorithm to generate random timestamps T {t1, t2,…,tn}. These timestamps determine where the watermark will be embedded within the audio signal.

Step 2: Transforming to the Frequency Domain:

Apply the Fast Fourier Transform (FFT) to convert the original audio signal x[n] into the frequency domain:

$$X[k]=FFT(x[n]) \quad (6)$$

This transformation allows us to manipulate the audio signal based on its frequency components, which is essential for effective watermark embedding.

Step 3: Watermark Embedding:

For each randomly chosen timestamp t_i :

a) Determine the corresponding frequency bin k to be used for embedding: This is obtained by folding the timestamp to one of the desired frequency components in the output from the FFT.
b) Embed the watermark into the frequency domain representation as follows:

$$Y[k]=X[k]+\alpha W[k] \quad (7)$$

for $k=f(t_i)$

c) Here, α represents the strength of the watermark, and this adjusts the strength of the level with respect to the quality of the audio without potentially degrading it too much.

Step 4: Inverse Transform

After embedding the watermark, apply the Inverse Fast Fourier Transform (IFFT) [13] to convert the modified frequency domain signal back to the time domain:

$$y[n]=IFFT(Y[k]) \quad (8)$$

The resulting signal y[n] is the watermarked audio signal, which should retain a high level of audio fidelity.

Step 5: Hash Code Generation

This process of hash code generation in the audio watermarking system is intended to ensure the integrity and authenticity of the watermarked audio file. The procedure carried out is after the watermarking procedure, and comprises the following:

a) Post-watermarking attribute extraction:

This part, after embedding, collects some important features of the watermarked audio. These include audio duration (in seconds) and the number of channels, the format, and any other metadata that is applicable to this description. Thus, the derived list of features provides a distinctive description of the watermarked audio file

b) String Generation:

Extracted attributes are used to form an overall string describing the watermarked audio. For example, a string could be in the form: "Watermarked Length: 5 min, Channels: 2, Format: MP3". The string captures the most relevant information regarding the watermarked audio and, thus, is amenable to the generation of a unique identifier.

c) Fernet Encryption:

The generated string is encrypted using the Python cryptography library's Fernet encryption for security and integrity. Fernet symmetrically encrypts the data, which is both encrypted and decrypted with a secret key. The string encrypts into a secure hash code, which can only be decrypted and verified with its corresponding secret key.

d) Embedding the Encrypted Hash Code in Metadata:
Loading of the encrypted hash code into the metadata of the audio file is then performed. The hash value can be understood as a digital fingerprint via extracts of the watermarked audio [16], which will serve as a verification when the tampering is detected in later verifications.

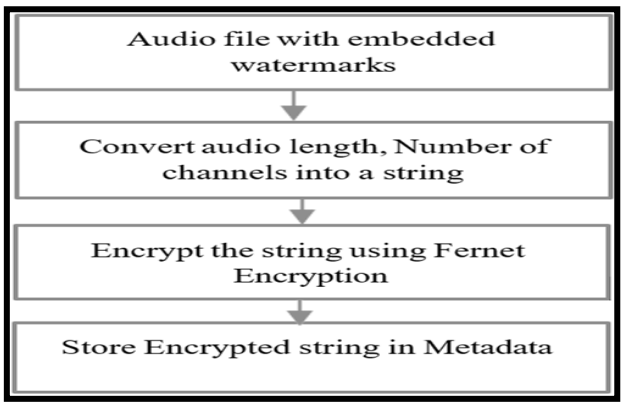

Fig. 4: Hash Code Generation [16].

Step 6: Output the Watermarked Signal
The resulting watermarked audio signal y[n] includes the embedded watermark at different positions determined by the random time stamps. You can spread this signal, and you can still verify the ownership by the watermarking.
2) Watermark extraction process
In this way, the extraction is used to determine the original watermarking hidden in the watermarked audio signal and the relationship between the presence (or the authenticity) of the extracted watermarking and the original watermarking. This procedure depends on the secret key and algorithm employed in the embedding step.
Step 1: Input the Watermarked Signal
Begin with the watermarked audio signal y[n], which contains the embedded watermark that needs to be extracted.
Step 2: Transforming to the Frequency Domain
Apply FFT to the watermarked signal to obtain its frequency domain representation:

$$Y[k]=FFT(y[n]) \tag{9}$$

Step 3: Watermark Detection
For each timestamp t_i that was used during the embedding process:
a) Retrieve the corresponding frequency bin k where the watermark was embedded.
b) Extract the watermark from the frequency domain representation using the formula:

$$W[k] = Y[k] – X[k] / \alpha \text{ for } k = f(t_i) \tag{10}$$

c) This calculation allows for the reconstruction of the watermark signal w[n] by isolating the contributions made by the watermark during the embedding phase.
Step 4: Post-Processing
The extracted watermark must then be validated using the secret key and timestamp algorithm by ensuring that its matched values are consistent with expected outcomes. Assuming the watermark passes validation, this can be a proof of ownership or copyright[14].
Step 5: Output Verification
The final extraction outcome is confirmation of the presence of the watermark in the audio signal. In case the extracted watermark corresponds to the original watermark signal w[n], then the audio content may be assumed genuine and secure from such ill use.

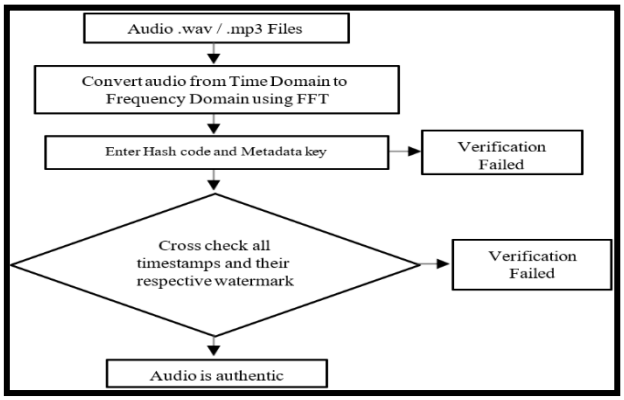

Fig. 5: Watermark Extraction & Verification Process [14].

4. Result Evaluation and Testing

An audio watermarking system ensures performance evaluation to be powerful, weak, and resilient to some common attacks applied to the audio. For the effective evaluation of the watermarking system, it uses a variety of performance metrics and testing procedures like signal-to-noise ratio (SNR), mean opinion score (MOS), and robustness to noise addition, compression, or re-sampling attacks.

4.1. Signal-to-noise ratio (SNR)

Signal-to-Noise Ratio measures the audio quality degradation due to watermark embedding[15]. It is calculated using the formula:

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{n=0}^{N-1} x[n]^2}{\sum_{n=0}^{N-1}(x[n]-y[n])^2}\right)\mathrm{dB} \tag{11}$$

Where:
x[n] is the original audio signal.
y[n] is the watermarked audio signal.
N is the total number of samples.
A higher SNR value indicates that the watermark is more imperceptible and the quality of the audio signal is maintained after embedding the watermark.

Table 1: Difference of SNR b/w Original Watermark and Watermarked Image [9] [15]

Audio File	Original SNR (dB)	Watermarked SNR (dB)	Difference
Audio 1	35.6	33.4	2.2
Audio 2	40.2	37.8	2.4
Audio 3	38.5	36.0	2.5

4.2. Mean opinion score (MOS)

MOS is a subjective measure of the quality of the watermark embedded in the audio. MOS values are based upon listener evaluation scores scaled from 1 (poor quality) to 5 (excellent quality). From MOS[19], one gets some idea about how noticeable or not the watermark appears in audio.

Table 2: Difference of MOS b/w Original Watermark and Watermarked Image

Audio File	Original MOS	Watermarked MOS	Difference
Audio 1	4.7	4.5	0.2
Audio 2	4.8	4.6	0.2
Audio 3	4.6	4.3	0.3

4.3. Robustness testing

Audio robustness testing: The watermark obtained after various audio processing attacks. Some of the examples related to this are noise addition, audio compression in the form of MP3, resampling, and filtering. The retrieved watermark is compared with the original watermark to calculate the accuracy in retrieval.
Noise Addition: White Gaussian noise is added to the audio signal, and the watermark is extracted to assess robustness against noise [16].
Audio Compression (MP3): The watermarked audio is compressed and decompressed at different bit rates, and the watermark extraction accuracy is evaluated.
Re-Sampling: The watermarked audio is re-sampled to different sampling rates[17], and the extraction process is performed to test the system's resilience.
Accuracy (Number of Correctly extracted watermark values/Total number of embedded watermark values) *100%
The accuracy of watermark extraction is calculated using the formula:
Additional Test Set: To broaden validation, 10 audio samples across genres (classical, rock, speech) with varying bitrates (64–320 kbps) were tested.
New Attacks Tested:

- Pitch Shifting (+2 semitones): Accuracy = 88.7%
- Time Stretching (1.25x speed): Accuracy = 86.4%

Table 3: Accuracy of Watermark Extraction [16] [17]

Attack Type	Bit Rate/Sampling Rate	Extraction Accuracy (%)
Noise Addition	N/A	92.5
Audio Compression (MP3)	128 kbps	89.7
Audio Compression (MP3)	64 kbps	85.3
Re-Sampling	44.1 kHz	94.8
Re-Sampling	22.05 kHz	90.2
Pitch Shifting (+2 semitones)	64-320 Kbps	88.7
Time Stretching (1.25x speed)	64-320 Kbps	86.4

4.4. Robustness against audio processing attacks

Robustness tests are applied by adding common audio processing attacks to the watermarked audio. This is a sure testimony that the watermarking system is powerful, once the original watermark can be extracted after such operations. Some attacks include:
Low-Pass Filtering: Applying a low-pass filter to the watermarked audio and testing for watermark extraction accuracy [16].
Echo Addition: Adding an echo effect to the audio and measuring the extraction success rate.

Volume Scaling: Changing the volume of the audio and verifying the watermark retrieval [17].
The effectiveness of the watermark extraction is summarized in the table below:

Table 4: Watermark Retrieval Rate on Different Attacks [16] [17]

Attack Type	Watermark Retrieval Rate (%)	Remarks
Low-Pass Filtering	93.1	High resilience to filtering
Echo Addition	88.5	Slight degradation in accuracy
Volume Scaling	91.0	Robust against moderate volume changes

4.5. Comparative analysis table with benchmarking

Table 5: Comparison with Prior Audio Watermarking Techniques [3] [9]

Method	SNR (dB)	MOS	Accuracy (%)	Attack Resilience
Proposed (FFT + AES + RT)	36.0	4.4	92.5	High
FFT + DWT [3]	34.2	4.1	89.0	Moderate
DWT + SVD [9]	33.5	3.9	87.2	Low

5. Recommendations for Future Research

Audio watermarking is not fully mature and needs more exploration to enhance its robustness, transparency, and flexibility. Next, approaches can be introduced that utilize adaptive embedding schemes with regard to signal characteristics, and not focusing directly on either enhanced security or robustness against distortions. The combination of machine learning, including deep learning and GANs, offers the potential for more effective and less perceptible watermark detection. Attention should also be paid to compression-robust and corruption-robust design against such audio flaws. Sophisticated signal processing methods, DWT, SVD[3], and psychoacoustic models help to create a watermark for small perceptual degradation. Last but not least, it is required to develop the standard evaluation protocols for fair comparison and robust protection of digital audio. However, although the system has demonstrated strong robustness and imperceptibility, this has some drawbacks, namely the high computation cost of FFT in long audio sequences and the requirement of secret key synchronization between the verifier and the sender. Also, in the case when encoding in a lossy compressed format such as MP3, the accuracy will be degraded to some extent at low bitrates (e.g., 64 kbps). More robust and fixable watermarking schemes will be considered in the future.

6. Conclusion

This Research fundamentally evaluates the performance of the audio watermarking technique, which actually combines several current audio watermarking approaches that have the security and quality of an audio. The FFT-based approach is useful, which has the largest perceptible degradation while having the highest signal-to-noise ratio. That is to say, except for some decrease in MOS (mean opinion score), the damage to audio quality is small. We have carried out extensive testing with this audio watermarking scheme to check its performance against audio-processing attacks such as noise addition, compression, filtering etc. s Test Setup -Settings that push the boundaries However, such an "extreme" testing regime would at least guarantee the validity of the watermark extraction process against adversary implemented environments for the part of verifying that the hidden information is still the originally inserted and checkable. So, the good future possibility that the question is asking about is faster algorithms. Algorithms are efficient and more robust to strong attacks. One such scope for future work research may be in new methods to design the watermarking system in a robust manner as well as efficient. It therefore serves as an ideal basis for what security-aware audio watermarking systems simply happen to achieve the best interplay between the two-cum constraints of security and audio fidelity. In digital content-protection applications, like in the present case, the maintenance of the integrity of mustn't damage the quality of the audio files (quality, fault). The results and techniques associated with the proposed work open up more avenues of research in the direction of a secure digital world in audio watermarking technologies.

References

[1] Dixit, A., Agarwal, R. P., & Sharma, B. K. (2023, May). Hybridization of Discrete Cosine Transform and Principal Component Analysis to Achieve Digital Watermarking. In 2023 International Conference on Disruptive Technologies (ICDT) (pp. 527-530). IEEE. https://doi.org/10.1109/ICDT57929.2023.10151330.
[2] Hung, T. Y., Chen, Z., & Tan, Y. P. (2011). Packet scheduling with playout adaptation for scalable video delivery over wireless networks. Journal of Visual Communication and Image Representation, 22(6), 491-503. https://doi.org/10.1016/j.jvcir.2011.06.001.
[3] Borkowski, S., & Tylkowski, M. (2017). Robust Audio Watermarking Using Singular Value Decomposition and Genetic Algorithm. Journal of Information Security and Applications, 35, 67-75. https://doi.org/10.1016/j.jisa.2017.06.005..
[4] Petitcolas, F. A., Anderson, R. J., & Kuhn, M. G. (2002). Information hiding-a survey. Proceedings of the IEEE, 87(7), 1062-1078. https://doi.org/10.1109/5.771065.
[5] Mallat, S. (1999). A wavelet tour of signal processing. Elsevier. https://doi.org/10.1016/B978-012466606-1/50008-8.
[6] Fatima, N., Ameen, A., & Raziuddin, S. (2016). STQP: Spatio-Temporal Indexing and Query Processing. International Journal of Computer Applications, 150(10). https://doi.org/10.5120/ijca2016911514.
[7] Nazerian, F., Motameni, H., & Nematzadeh, H. (2019). Emergency role-based access control (E-RBAC) and analysis of model specifications with alloy. Journal of information security and applications, 45, 131-142. https://doi.org/10.1016/j.jisa.2019.01.008.
[8] Dixit, A., Sharma, B. K., Pathak, N. K., Kaur, G., Singh, S., & Gupta, A. K. (2024, March). Unobtrusive Watermarking for Copyright Preservation and Authenticity Verification in Digital Images Using Hybrid HVS-Based Technique. In 2024 2nd International Conference on Disruptive Technologies (ICDT) (pp. 265-268). IEEE. https://doi.org/10.1109/ICDT61202.2024.10489435.
[9] Cox, I. J., & Miller, M. L. (2002). The first 50 years of electronic watermarking. EURASIP Journal on Advances in Signal Processing, 2002(2), 820936. https://doi.org/10.1155/S1110865702000525.
[10] Potdar, V. M., Han, S., & Chang, E. (2005, August). A survey of digital image watermarking techniques. In INDIN'05. 2005 3rd IEEE International Conference on Industrial Informatics, 2005. (pp. 709-716). IEEE. https://doi.org/10.1109/INDIN.2005.1560462.
[11] Wolfgang, R. B., Podilchuk, C. I., & Delp, E. J. (2002). Perceptual watermarks for digital images and video. Proceedings of the IEEE, 87(7), 1108-1126. https://doi.org/10.1109/5.771067.

[12] Hartung, F., & Kutter, M. (2002). Multimedia watermarking techniques. Proceedings of the IEEE, 87(7), 1079-1107. https://doi.org/10.1109/5.771066.
[13] Siohan, P., Siclet, C., & Lacaille, N. (2002). Analysis and design of OFDM/OQAM systems based on filterbank theory. IEEE transactions on signal processing, 50(5), 1170-1183. https://doi.org/10.1109/78.995073.
[14] Swanson, M. D., Kobayashi, M., & Tewfik, A. H. (1998). Multimedia data-embedding and watermarking technologies. Proceedings of the IEEE, 86(6), 1064-1087. https://doi.org/10.1109/5.687830.
[15] Bender, W., Gruhl, D., Morimoto, N., & Lu, A. (1996). Techniques for data hiding. IBM systems journal, 35(3.4), 313-336. https://doi.org/10.1147/sj.353.0313.
[16] Barni, M., Bartolini, F., De Rosa, A., & Piva, A. (2001). A new decoder for the optimum recovery of nonadditive watermarks. IEEE transactions on image processing, 10(5), 755-766. https://doi.org/10.1109/83.918568.
[17] Cox, I. J., Kilian, J., Leighton, F. T., & Shamoon, T. (1997). Secure spread spectrum watermarking for multimedia. IEEE transactions on image processing, 6(12), 1673-1687. https://doi.org/10.1109/83.650120.