Adjustable audio watermarking algorithm based on DWPT and psychoacoustic modeling

Mustapha Hemis · Bachir Boudraa · David Megías · Thouraya Merazi-Meksen

Abstract This paper presents a novel adjustable audio watermarking method with high auditory quality by exploiting the discrete wavelet packet transform (DWPT), psychoacoustic modeling and distortion compensated-dither modulation (DC-DM) quantization. While the DWPT is used to divide the audio frames into several frequency sub-bands, the psychoacoustic model is intergraded to determine the appropriate sub-bands for watermarking and to control the number of embedded bits in each one. Then, the DC-DM technique is used to embed the watermark bits into the appropriate DWPT coefficients. The synchronization code technique is adopted in the proposed method to withstand desynchronization attacks. In order to achieve an adjustable watermarking scheme, two regulator parameters are provided to manage the capacity-robustness trade-off. The performance of the watermarking scheme is evaluated by examining different host audio signals under various watermarking attacks. The results show excellent imperceptibility of watermarked signals with an average ODG of -0.3. In addition, the proposed scheme provides strong robustness against the attacks with low capacity. However, high capacity (about 2500 bps) can be achieved while maintaining a reasonable robustness. A comparison with some state-of-the-art audio watermarking schemes reveals that the proposed method provides competitive results.

Keywords Digital audio watermarking · Discrete wavelet packet transform · Quantization index modulation · Distortion compensated-dither modulation · Psychoacoustic modeling

M. Hemis · B. Boudraa · T. Merazi-Meksen
Speech Communication and Signal Processing Laboratory,
University of Sciences and Technology Houari Boumediene (USTHB),
P.O. Box 32, El-Alia, Bab-Ezzouar, Algiers 16111, Algeria
E-mail: mhemis@usthb.dz, bboudraa@usthb.dz, tmeksen@usthb.dz

D. Megías
Estudis d'Informàtica, Multimèdia i Telecomunicació,
Internet Interdisciplinary Institute (IN3),
Universitat Oberta de Catalunya, Barcelona, Catalona, Spain
E-mail: dmegias@uoc.edu

1 Introduction

With the rapid development of multimedia and social networks, digital documents can be easily copied, edited, and distributed without any authorization. The protection of intellectual property rights has become an urgent necessity. Digital watermarking [1, 2] has been introduced as a technique to solve problems as varied as the protection of the copyright, content authentication, fingerprinting and broadcast monitoring. Using this technique, hidden information –called watermark– is imperceptibly embedded into the host media (audio, image or video) and can be extracted later on to verify the authenticity.

In the context of audio watermarking, an effective scheme must satisfy the imperceptibility, robustness, capacity and security requirements [3]. The imperceptibility means that the watermarked audio signal must be perceptually similar to the original one. The capacity determines the maximum amount of data that can be embedded in the original signal, which is usually measured in the unit of bits per second (bps) for audio contents. Under the imperceptibility and capacity constraints, the watermark should also be robust against most signal processing attacks such as MP3 compression, noise addition, re-sampling, or re-quantization, among others. The watermarking process should also be secure so that only an authorized person can extract, remove or embed the watermark. In addition, the algorithm should be tunable to various degrees of robustness and capacity to be suitable for different applications [4].

In recent years, audio watermarking techniques have achieved significant progress, and several algorithms for embedding watermarks into audio data have been proposed. These algorithms can be broadly classified into different categories: spread spectrum-based schemes [5–8], patchwork-based schemes [9–12], echo hiding-based schemes [13–16], histogram-based schemes [17, 18] and quantization index modulation (QIM)-based schemes [19–25]. Among all these methods, the QIM shows great potential to achieve a good imperceptibility-robustness performance. This technique was introduced by Chen and Wornell [26]. Later on, many QIM-based audio watermarking schemes were proposed for time domain [19] and frequency domains, such as the discrete cosine transform (DCT), the discrete Fourier transform (DFT) or the discrete wavelet transform (DWT) [20–25].

In [20], Bhat at al. proposed a robust audio watermarking scheme in which the watermark is embedded by applying a QIM process to the norm of the singular values (SV) that are obtained by performing singular values decomposition (SVD) transform on the wavelet domain blocks. The quantization step is adaptively determined in order to increase robustness and decrease distortion. Despite the robustness against some signal processing attacks, the capacity of this method is quite low. Singh et al. [21] presented a robust technique for MPEG-1/Audio Layer II compressed domain. The watermark embedding process is performed by modifying the sub-band coefficients of an audio signal using QIM. The temporal and frequency masking obtained from a perceptual model of the Human Auditory System (HAS) are exploited to satisfy the imperceptibility, robustness and security requirements. Wang et al. [22] incorporated a support vector regression technique into the DWT-DCT structure, in which an adaptive QIM is performed on the audio signal. This approach uses the corresponding feature of the template in the training samples to achieve a favorable trade-off between imperceptibility and robustness. However, the reported imperceptibility results are inaccurate,

since distortion is evaluated in terms of peak signal-to-noise ratio (PSNR), which is not correlated to human perception. In addition, the robustness of the scheme is assessed against a very small set of attacks. Lei et al. [23] presented a robust audio watermarking scheme based on the Lifting Wavelet Transform (LWT) and SVD. The watermark is embedded into the SV of the low-frequency LWT coefficients using QIM. This scheme yields an estimated data payload of 170 bps. However, in many audio watermarking applications, a larger capacity is required. In [24], the authors proposed a blind audio watermarking algorithm based on the vector norm and the approximation coefficients of the DWT. The watermark is embedded in the vector norm of the segmented approximation DWT coefficients by applying QIM with an adaptive quantization step determined by the signal-to-noise ratio (SNR). This scheme is robust against several attacks, but it remains vulnerable to amplitude scaling.

The exploitation of human auditory properties in audio watermarking is one of the effective ways to achieve a convenient trade-off between imperceptibility, robustness and capacity. Several methods relying on perceptual models have been proposed in the literature. For instance, Tsai et al. [27] proposed an intelligent scheme using the DCT and a neural network. They exploited the auditory masking while selecting a suitable coefficient index from a DCT block. The neural network is employed to perform the watermark extraction. In this work, imperceptibility tests are not conducted and the robustness against attacks is unsatisfactory. Hu et al. [28] proposed a blind audio watermarking algorithm by combining the discrete wavelet packet transform (DWPT), the DCT, and exploiting the human auditory masking. The audio signal is decomposed into critical bands using DWPT. Then, DCT is applied to analyze the spectral content of these critical bands. Subsequently, the watermark is embedded in the DCT coefficients by using a perceptual-based QIM technique. As shown in the experimental results, when the capacity is increased, the robustness of this scheme against low bit rate MP3 compression and noise corruption is limited. The same authors proposed another work that exploits the double transform DWPT-SVD and the human auditory masking property [29]. In this scheme, the watermark bits are embedded by modifying the resulting singular values, after the DWPT-SVD transform, subject to perceptual criteria. This system provides a good imperceptibly-robustness trade-of, but the obtained capacity is relatively low. The modified DCT (MDCT) filter bank is used in [30]. The watermarking process consists in quantizing the MDCT coefficients using the QIM technique. For each frequency sub-band, a maximum watermark embedding capacity is calculated using a psychoacoustic model inspired from the MPEG-AAC standard. With this scheme, high capacity can be obtained at the expense of robustness. However, as revealed by the author, the system is fragile against all signal processing attacks. Fu et al. [31] combine the polyphase filter bank, the empirical mode decomposition (EMD) and the psychoacoustic model. Using the polyphase filter bank analysis, the original audio signal is decomposed into several sub-bands, and then, each of these sub-bands are segmented, and the EMD is applied to every segment to extract a set of intrinsic mode function (IMF) and a final residual. The watermark is embedded into this residual. The imperceptibility of the watermark is controlled by using the psychoacoustic model. However, this scheme is not endowed with a synchronization technique, making it fragile against geometrical attacks, such as cropping and jittering. Fallahpour and Megías [32] presented a high capacity audio watermarking scheme in the log-

arithm domain based on the absolute threshold of hearing (ATH). The key idea is to divide the selected frequency band into short frames and then embed the watermark by quantizing the samples according to the ATH. This scheme provides high embedding capacity with a reasonable robustness against attacks. However, since the HAS model is applied using only its passive properties, the imperceptibility results are, in some cases, unsatisfactory.

Most audio watermarking methods are designed in such a way that they satisfy a single objective with unvarying capacity and robustness. As already remarked, an effective watermarking scheme should allow tuning for varying robustness and capacity to be suitable for different applications. However, the watermarking tuning problem has been rarely addressed in the literature. In this paper, we propose a solution to this problem and develop an adjustable audio watermarking scheme by exploiting the benefits of DWPT, a psychoacoustic model, and distortion compensated-dither modulation (DC-DM) quantization. The proposed scheme can be tuned for different capacity and robustness, as required by the application.

The main contributions of this paper include: (i) the flexibility of DWPT is used to divide the audio frame into several frequency sub-bands; (ii) the DC-DM technique is used to embed the watermark bits into the suitable DWPT coefficients; (iii) an iterative algorithm based on psychoacoustic model is proposed to optimize the number of embedded watermark bits in each sub-band; (iv) a synchronization technique is integrated to resist desynchronization attacks; (v) a chaotic map is applied to encrypt the watermark, which enhances the security of the proposed scheme; and (vi) two regulator parameters (quantization step of DC-DM and a predefined threshold) are used to tune the robustness and capacity of the scheme while maintaining a high audio quality of the watermarked signal.

The remainder of this paper is organized as follows. Section 2 introduces background information, including DWPT, psychoacoustic models and the DC-DM scheme. Section 3 discusses the proposed audio watermarking method. The experimental results and a comparison with prior state-of-the-art audio watermarking schemes are presented in Sections 4 and 5, respectively. Finally, Section 6 summarizes our study.

2 Background

In this section, we provide a brief overview of the preliminaries that are used in this paper, namely DWPT, psychoacoustic modeling and DC-DM embedding.

2.1 Discrete wavelet packet transform

The wavelet transform is a time-scale signal analysis technique. It was developed as an alternative to the short term Fourier transform (STFT) to overcome the problems related to the properties of time and frequency resolutions. More specifically, in contrast to the STFT that provides uniform temporal resolution for all frequencies, the wavelet analysis provides a high temporal resolution and a low frequency resolution for high frequencies, and high frequency resolution and low

time resolution for low frequencies. This produces multi-resolution information for the entire signal.

The continuous wavelet transform (CWT) [33] of a signal $x(t)$ is defined as follows:

$$CWT(\alpha,\tau)=\frac{1}{\sqrt{\alpha}}\int x(t)\psi\left(\frac{t-\tau}{\alpha}\right)\mathrm{d}t, \tag{1}$$

where t, τ and α are, respectively, the time, the translation parameter and the scale parameter, and $\psi(t)$ is the transforming function, called mother wavelet.

In discrete time, a signal $x(n)$ can be equivalently transformed by the DWT [34] as follows:

$$DWT(m,n)=2^{-m/2}\sum_{k}x(k)\psi(2^{-m}k-n). \tag{2}$$

Equation 2 is the discretized version of Equation 1, with $\alpha=2^m$ and $\tau=2^m n$, where m, n and k are integers.

In practice, the DWT is often achieved by convolving the input signal with a pair of low and high pass quadrature mirror filters. Fig. 1 presents a single level one dimensional DWT decomposition and reconstruction (analysis and synthesis). Starting from a signal x, two sets of coefficients are computed: approximation coefficients A, and detail coefficients D. These coefficients are obtained by convolving x with the low-pass filter *Lo_D* for approximation, and the high-pass filter *Hi_D* for detail, followed by a dyadic decimation stage (downsampling by 2, typically denoted as "$\downarrow 2$"), as shown in Fig. 1(a). In synthesis process, the original signal can be reconstructed from the obtained approximation (A) and detail coefficients (D) by using two inverse filters *Lo_R* and *Hi_R*, preceded by an interpolation function (up-sampling by 2, typically denoted as "$\uparrow 2$"), as shown in Fig. 1(b).

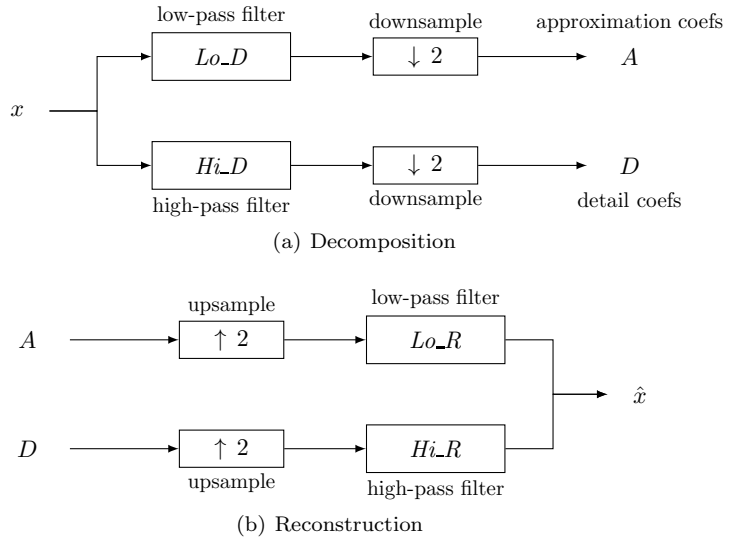
The DWPT has been proposed as an extension of the wavelet transformation. Contrarily to the DWT, where only the low frequency band is further decomposed into low and high frequency bands in subsequent levels of decomposition, both high and low sub-bands are subsequently decomposed at each level in the DWPT. The process continues until the desired decomposition is achieved. This type of decomposition is represented by a binary tree (Fig. 2).

The DWPT suffers from a problem known as shift variance behavior, which means that small shifts in the input signal can cause major variations in the distribution of energy between the DWT coefficients at different scales. This problem is usually solved by using shift invariance transforms, such as the stationary wavelet transform (SWT) [35], the dual-tree complex wavelet transform (DT CWT) [36], or the shift-invariant wavelet packet transform (SIWPD) [37]. However, such transforms involve substantially increased computational requirements.

In the proposed system, the DWPT is implemented without taking into account the shift-invariance problem. This issue will be addressed in the future research.

2.2 Psychoacoustic modeling

Human hearing has two main properties. Firstly, the limit on minimally audible energy levels differs depending on frequency. Secondly, frequency masking occurs, i.e. two sounds of close frequencies are emitted at the same time. According to the power of both sounds, it is possible that only the sound of stronger power is heard,



(a) Decomposition

(b) Reconstruction

Fig. 1 One level wavelet decomposition/reconstruction.

even though both sounds are perfectly audible separately. The frequency masking model defined in ISO-MPEG Audio Psychoacoustic Model 1, for Layer I [38] is used in the proposed method. This model is less complex and allows more compromises to simplify the calculations. It is initially designed to determine the maximum amount of inaudible quantization noise that can be introduced by the process of audio coding. In our method, the main purpose of using the psychoacoustic model is to increase the embedding capacity of the audio signal in an inaudible way.

The calculations of the frequency masking according to the ISO-MPEG audio psychoacoustic model are detailed in [39]. The key steps can be described as follows:

1. Spectral analysis to derive power spectral density (PSD).
2. Determination of the threshold in a noiseless environment (absolute threshold).
3. Identification of tonal and non-tonal (noise) components.
4. Decimation of the masked components (components below the absolute threshold).
5. Calculation of the individual masking thresholds.
6. Determination of the global masking thresholds.

Actually, the implementation of a psychoacoustic model is performed on each frame by dividing the spectrum into 32 frequency sub-bands. The signal-to-mask ratio (SMR) is then calculated, for each sub-band, as the ratio of signal spectrum to minimum of the global masking threshold (exemplified in Fig. 3). The smaller the SMR, the higher the insensitivity of the sub-band to watermarking distortion. Any modification made by the watermarking scheme under the global masking threshold is assumed to be imperceptible by the human ear. It is, therefore, possible to determine the appropriate sub-bands for watermark embedding, along with the number of bits to be embedded in each sub-band, in an inaudible way.

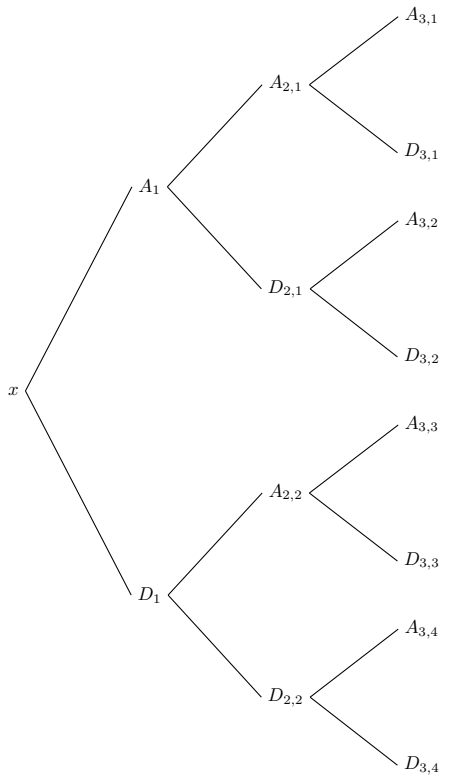


Fig. 2 A tree representation of 3-level DWPT decomposition.

2.3 DC-DM embedding

DC-DM is an extension of the traditional QIM method proposed by Chen and Wornell [26]. In basic QIM, the embedding function corresponds to the quantization of the host signal X by a quantizer Q_w dependent on the symbol w to be embedded:

$$\hat{X} = Q_w(X; \Delta), \tag{3}$$

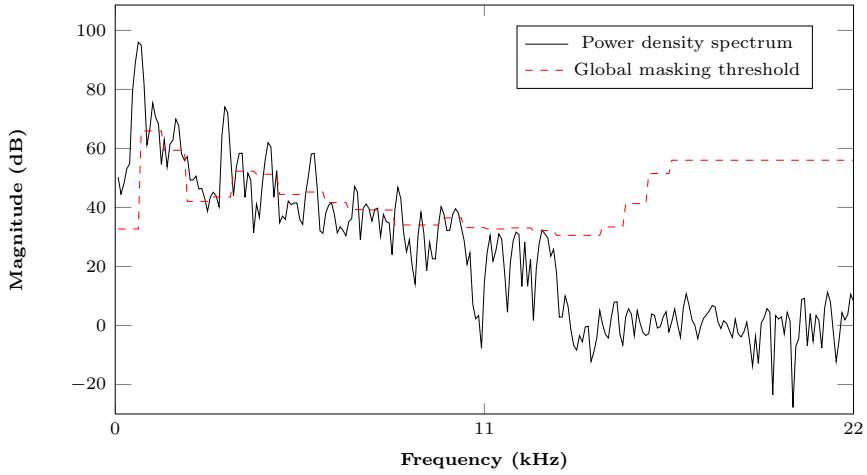
where $\hat{X}$ is the watermarked signal and Δ is the quantization step. In audio watermarking, the embedded symbol w is usually in a binary format, i.e. $w \in \{0, 1\}$.

In the DM quantization, the embedding function is defined by the following expression:

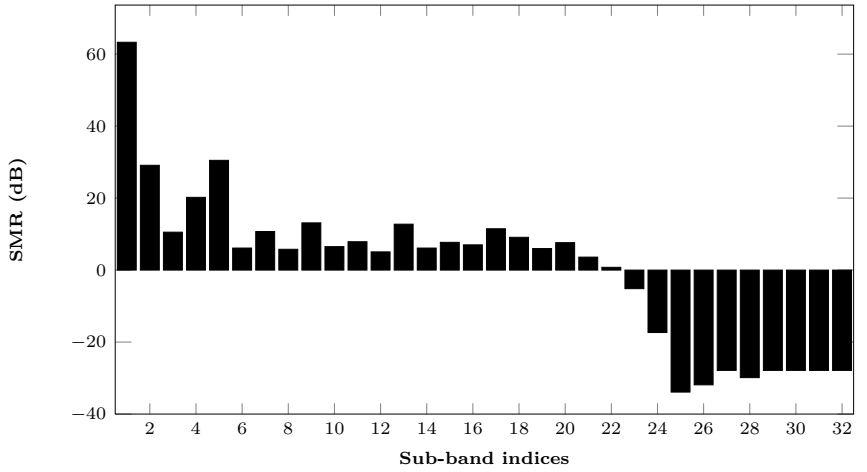
$$\hat{X} = Q(X - d_w; \Delta) + d_w, \tag{4}$$

where d_w is a key-dependent dither sequence. We choose d_0 pseudo-randomly with a uniform distribution over $[-\Delta/2, \Delta/2]$ and calculate d_1 using the following function:

$$d_1 = \begin{cases} d_0 + \frac{\Delta}{2}, \text{ if } d_0 < 0, \\ d_0 - \frac{\Delta}{2}, \text{ if } d_0 \geq 0. \end{cases} \tag{5}$$



(a) Representation of the global masking threshold



(b) SMR magnitude of the 32 sub-bands

Fig. 3 The MPEG-1 psychoacoustic model for a selected audio frame.

The DC-DM is proposed to improve the achievable rate distortion and robustness trade-off of DM methods. The embedding function can be given by:

$$\hat{X} = Q(X - d_w; \Delta/\alpha) + (1 - \alpha)\left[X - Q(X - d_w; \Delta/\alpha)\right], \tag{6}$$

where $\alpha \in [0, 1]$ is the compensated factor.

If $\tilde{X}$ is the received $\hat{X}$, the extraction is expressed by:

$$\tilde{w} = \underset{w \in \{0,1\}}{\arg\min} |\tilde{X} - Q(X - d_w; \Delta/\alpha)|. \tag{7}$$

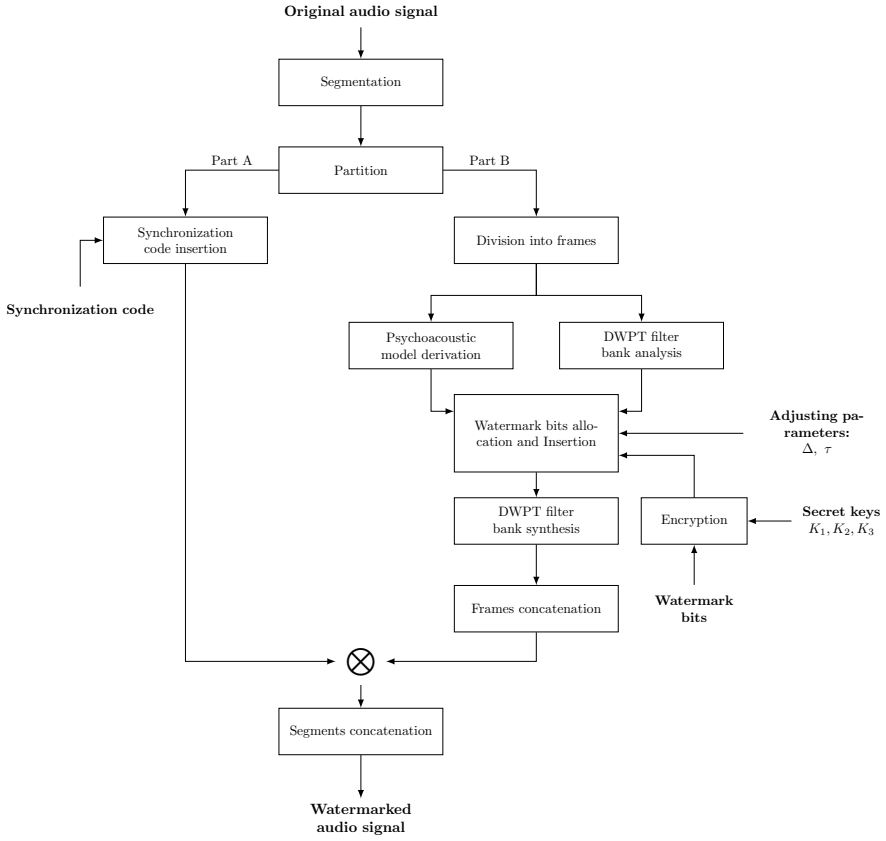


Fig. 4 Watermark embedding process

3 Design of the watermarking scheme

This section details both the watermark embedding and the watermark extraction methods.

3.1 Watermark embedding process

The proposed watermarking embedding process can be summarized as follows. The original audio signal is divided into long segments and each one is further divided into two parts. The synchronization code is embedded in the time domain of the first part, as described in Section 3.1.2, whereas the watermark information is embedded in the DWPT domain of the second part, as detailed in Section 3.1.3. The proposed watermark embedding process is depicted in Fig. 4.

Let $X = \{x(i), 1 \leq i \leq L_X\}$ denote the original audio signal with L_X samples. $W_0 = \{w_0(i), 1 \leq i \leq L_W\}$ $(w(i) \in \{0, 1\})$ is the watermark of length L_W, and $SYNC = \{sync(i), 1 \leq L_{sync}\}$ $(sync(i) \in \{0, 1\})$ is the synchronization code of length L_{sync}.

3.1.1 Watermark encryption

The original watermark must be encrypted prior to embedding to improve the security of the proposed scheme. In recent years, a new type of encryption based on chaotic maps has emerged. Due to their simplicity, rapidity and many chaos characteristics such as non-periodicity, unpredictability, and high sensitivity dependence to initial conditions, chaotic cryptosystems have received a lot of attention. However, any secure cryptosystem, either symmetric or public-key, may be used in the proposed scheme instead of chaotic maps.

In this paper, the Tent map [40] is selected to produce a chaotic sequence that is then used to encrypt the original watermark. The Tent map is defined by:

$$u(n+1) = \begin{cases} \mu u(n), & \text{if } u(n) < \frac{1}{2}, \\ \mu(1-u(n)), & \text{if } u(n) \geq \frac{1}{2}, \end{cases} \tag{8}$$

where $u(0) \in [0,1]$ is the initial parameter and $\mu \in [1,2]$ is the control parameter. These two parameters are used as secret keys K_1 and K_2.The output sequence $u(n)$ is then converted into a binary stream by the following formula:

$$u_b(n) = \begin{cases} 0, \text{ if } u(n) \geq \gamma, \\ 1, \text{ if } u(n) < \gamma, \end{cases} \tag{9}$$

where γ is a predefined threshold that can be used as a secret key K_3.

The encrypted watermark is finally generated by:

$$w(n) = u_b(n) \oplus w_0(n). \tag{10}$$

3.1.2 Embedding of synchronization codes

Similar to other audio watermarking schemes [8, 41–44], the proposed scheme is furnished with a synchronization technique to withstand the desynchronizing attacks. A Barker code of 32 bits is embedded in the time domain of the first part to locate the start position of the embedded watermark. Barker codes, which are subsets of pseudo number (PN) sequences, are commonly used for frame synchronization in digital communication systems. As discussed in previous works [43, 44], the good property of Barker codes is that they have a low correlation with a shifted version of themselves, which is very convenient for detecting the synchronization codes with no sample offset. In addition, embedding in the time domain reduces the computation cost while searching the code at the receiving end.

This paper makes use of the synchronization embedding technique presented in [42]. In this technique, the synchronization part A is first divided into L_{sync} sub-segments and then each bit of the synchronization code is embedded into each sub-segment. The embedding process is described in the following steps:

Step 1. Divide the segment A into L_{sync} sub-segments of equal lengths; each of which contains L_{SA} samples:

$$SA(i) = \{sa(i,j), 1 \leq j \leq L_{SA}\}, 1 \leq i \leq L_{sync}. \tag{11}$$

Step 2. Calculate the average of the external and internal samples of $SA(i)$, denoted by $A_{ext}(i)$ and $A_{int}(i)$, respectively:

$$A_{ext}(i) = \frac{sa(i,1) + sa(i,L_{SA})}{2}, \tag{12}$$

$$A_{int}(i) = \frac{1}{L_{SA}-2} \sum_{j=2}^{L_{SA}-1} sa(i,j), \tag{13}$$

Step 3. Calculate the parameter $\delta(i) = \max\{\delta_{\min}, \varphi|A_{ext}(i)|\}$ by using predefined parameters $\delta_{\min}$ and φ, which represent the minimum distance from the average of the external samples for bit-embedding and the distortion introduced with respect to the average of external samples, respectively.

Step 4. Embed each bit of the synchronization code $sync(i)$ into each sub-segment $SA(i)$. In order to embed a "1", the internal samples are changed such that their average is greater than that of the external samples. To embed a "0", the same idea is applied but by replacing internal samples such that their average is lower than that of the external ones.

- if $sync(i) = 1$ and $A_{int}(i) < A_{ext}(i) + \delta(i)$

$$\widehat{sa}(i,j) = sa(i,j) + d, \tag{14}$$

  with $d = A_{ext}(i) + \delta(i) - A_{int}(i)$ and $2 \le j \le L_{SA} - 1$.
- if $sync(i) = 0$ and $A_{ext}(i) < A_{int}(i) + \delta(i)$

$$\widehat{sa}(i,j) = sa(i,j) - d, \tag{15}$$

  with $d = A_{int}(i) + \delta(i) - A_{ext}(i)$ and $2 \le j \le L_{SA} - 1$.

Step 5. Concatenate the modified sub-segments $\widehat{SA}(i) = [\widehat{sa}(i,1), \widehat{sa}(i,2), \ldots, \widehat{sa}(i,L_{SA})]$ into one segment $\hat{A}$.

However, a fixed synchronization sequence could be exploited by the attackers, who may try to detect and remove the synchronization watermarks so as to make the watermark undetectable at the receiver's end. This problem can be circumvented by using different synchronization sequences for different audio files.

This problem can also be addressed by using different synchronization codes for the same file by means of a hash function. If h is a hash function, we can ask the embedder for the first synchronization code $SYNC_0$ (for example, having 32 bits). The following synchronization codes SYN_i, with $i > 0$ can be computed using the hash function as follows: $SYNC_i = h(SYNC_{i-1}, k_h)$, where k_h is the secret key of the hash function. In this way, an attacker cannot guess the synchronization codes if he/she does not know the initial synchronization sequence ($SYNC_0$) or the secret key (k_h). This solution requires transmitting both the first synchronization code and the secret key securely from the embedder to the receiver.

3.1.3 Watermark embedding

The watermark bits are embedded into the DWPT coefficients of the second part (B). The embedding process can be summarized as follows:

Step 1. The original segment B is divided into N_F quasi-stationary frames, each of L_F samples in length:

$$F(i) = \{f(i,j), 1 \leq j \leq L_F\}, 1 \leq i \leq N_F. \tag{16}$$

Step 2. Each frame is divided into N_S equal-width frequency sub-bands, obtained from the DWPT filter bank analysis:

$$S(n) = \{s(n,l), 1 \leq l \leq L_S\}, 1 \leq n \leq N_S, \tag{17}$$

where $s(n,l)$ denotes the l-th DWPT coefficient of the n-th sub-band. In our method, the number of sub-bands N_S is equal to 32, obtained by 5 levels DWPT decomposition (Fig. 5).

Step 3. Each frame is passed through a psychoacoustic model that determines the global masking threshold. For each frequency sub-band, $SMR(n)$ is obtained.

Step 4. Watermark bits allocation and embedding:
In our approach, a dynamic allocation of watermark bits is considered. This technique allows optimizing the number of embedded bits by considering the following properties:
- Selection of the suitable frequency sub-bands for embedding.
- Determination of the number of bits to be embedded at each sub-band.
- Determination of the suitable DWPT coefficients for embedding in each sub-band.

From the SMR, it can be determined which frequency sub-bands should receive most of the bits. Indeed, sub-bands with a minimum SMR can receive more bits without degrading audio quality. From Fig. 3(b), it can be noticed that high frequency sub-bands receive most part of watermark bits, guaranteeing imperceptibility. However, the watermark will be fragile against most signal processing attacks aimed to remove the watermark bits embedded in high-frequency sub-bands. In order to overcome this problem and ensure best trade-off between robustness and imperceptibility, the watermark bits should be distributed over the whole sub-bands. An iterative algorithm is applied to each frame to determine the appropriate sub-bands, the number of bits to be embedded in each sub-band, and the suitable DWPT coefficients. Following are the steps of the algorithm. The detailed process is depicted in Fig. 6.
For each sub-band n do:

(a) Embed the watermark bit $w(k)$ in the l-th DWPT coefficient using Equation 6.

(b) Calculate $SDR(n)$ (sub-band to distortion ratio) as the ratio of sub-band energy to distortion energy caused by the embedding

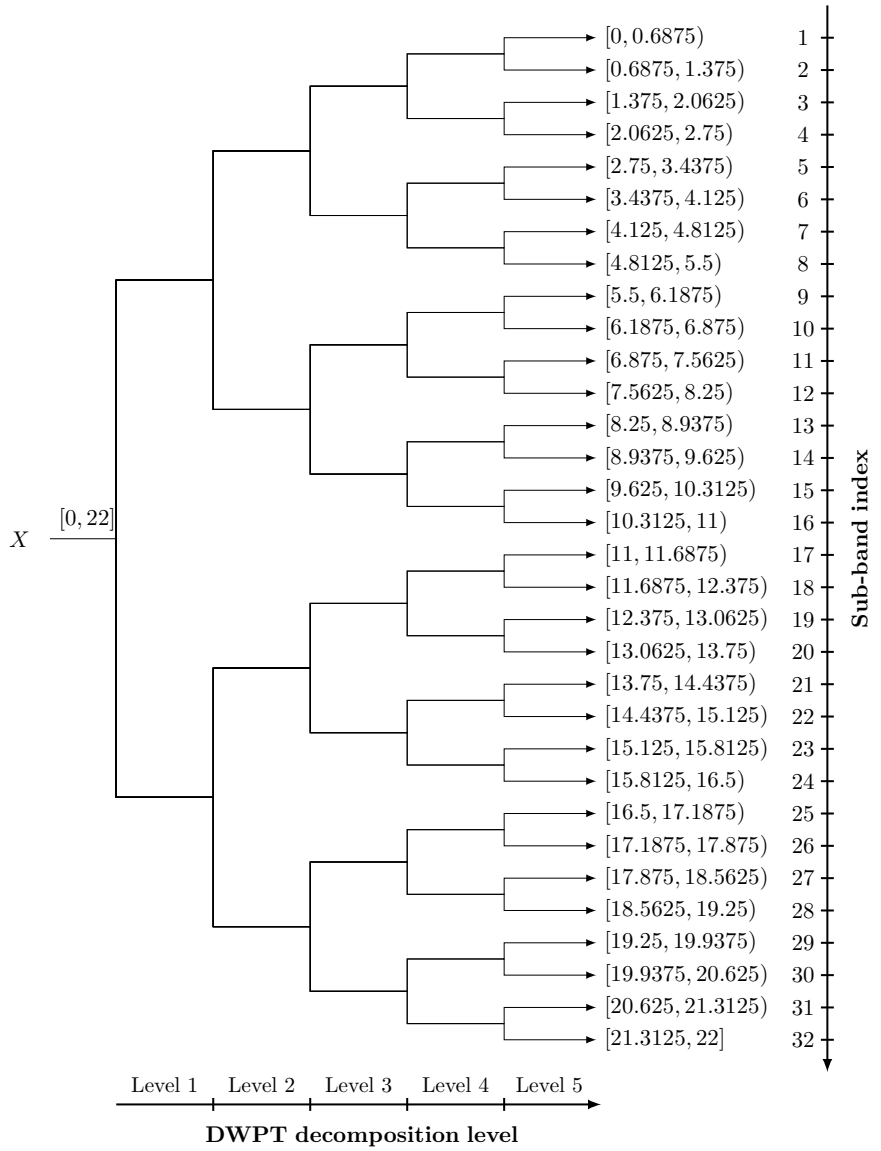


Fig. 5 5-level DWPT-based signal decomposition. The intervals in brackets of the resulting 32 sub-bands refer to the lower and higher cut-off frequencies in kHz.

of the watermark as follows:

$$SDR(n) = 10 \log \left(\frac{\sum_{l=1}^{L_S} [s(n,l)]^2}{\sum_{l=1}^{L_S} [s(n,l) - \hat{s}(n,l)]^2} \right), \qquad (18)$$

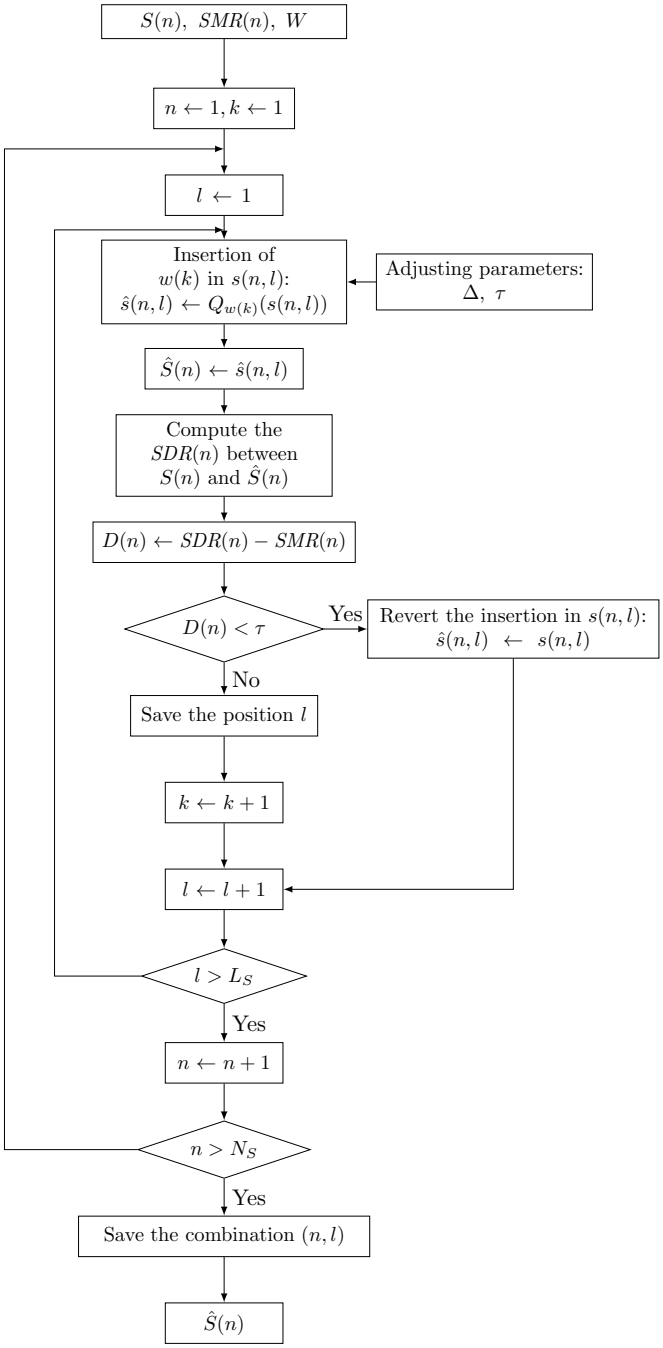


Fig. 6 Flowchart of the watermark bits allocation and the embedding process

where $s(n,l)$ and $\hat{s}(n,l)$ are the original DWPT coefficient and the modified (embedded) coefficient, respectively.

(c) Calculate the difference $D(n)$ between $SDR(n)$ and $SMR(n)$:

$$D(n) = SDR(n) - SMR(n). \tag{19}$$

(d) If $D(n) > \tau$, where τ is a predefined threshold, save the position l, increase k, select the next DWPT coefficient and go to Step 4.a.

(e) If $D(n) \leq \tau$, the distortion is supposedly audible and, thus, the considered coefficient is inadequate and the embedding must be reverted (i.e. the original coefficient is restored). Then, select the next DWPT coefficient and go to Step 4.a. If all the coefficients in the given sub-band satisfy this condition, this sub-band is inadequate for embedding.

This algorithm allows to control the number of embedded bits in each sub-band by taking into account their characteristics. Assuming m embedded bits into a given sub-band, the distortion caused by the watermarking will not be audible as long as its SDR is higher than its SMR (D must be the highest possible value). The threshold τ is introduced to control the imperceptibility: Increasing τ, results in better imperceptibility, but low embedding capacity, i.e. decreases the number of embedded bits (m).

Step 5. The modified sub-bands are passed through DWPT filter bank synthesis in order to obtain the watermarked frames denoted as $\hat{F}(i)$.

Step 6. Finally, the watermarked segment $\hat{B}$ is obtained by concatenation of all the modified frames.

3.1.4 Segment-by-segment embedding

The described embedding process, including synchronization code and watermark, is repeated for each audio segment. The resulting watermarked segments are then concatenated to form the watermarked audio signal $\hat{X}$.

3.2 Watermark extraction process

The watermark extraction process of the proposed technique consists in three key steps, namely, synchronization code detection, watermark extraction and decryption.

3.2.1 Detection of the synchronization codes

In order to locate the start position of the watermark, a search of the synchronization codes is performed along the audio signal. The extraction is carried out on a window of length L_A and repeated by moving the window one sample at a time until the synchronization code is detected. The extraction process is described as follows:

Step 1. Initialize the start position to $k = 1$.

Step 2. Define the extraction window $\tilde{A}$ of length L_A from $\tilde{x}(k)$ to $\tilde{x}(k+L_A)$.

Step 3. Divide $\tilde{A}$ into L_{sync} sub-segments $\widetilde{SA}(i)$ of L_{SA} samples.

Step 4. Calculate the external and internal average of $\widetilde{SA}(i)$ denoted by $\tilde{A}_{ext}(i)$ and $\tilde{A}_{int}(i)$, respectively.

Step 5. Extract each bit of synchronization code from each sub-segment $\widetilde{SA}(i)$ by using the following expression:

$$\widetilde{sync}(i) = \begin{cases} 1, \text{ if } \tilde{A}_{int}(i) > \tilde{A}_{ext}(i), \\ 0, \text{ otherwise.} \end{cases} \tag{20}$$

Step 6. Calculate the correlation between the extracted and the original code:

$$corr(SYNC, \widetilde{SYNC}) = \frac{\sum_{i=1}^{L_{sync}} sync(i)\widetilde{sync}(i)}{\sqrt{\sum_{i=1}^{L_{sync}} sync(i)^2}\sqrt{\sum_{i=1}^{L_{sync}} \widetilde{sync}(i)^2}}. \tag{21}$$

Step 7. If the correlation between $SYNC$ and $\widetilde{SYNC}$ is greater than or equal to a predefined threshold T_{sync}, then record the position k and proceed with the watermark extraction process. Otherwise, go to the next step.

Step 8. Increment k by 1 and repeat Steps 2–7.

3.2.2 Watermark extraction

Once the synchronization code is found, the watermark is extracted from the next L_B samples of the watermarked audio signal. Since we need to locate the positions of the appropriate sub-bands and the DWPT coefficients used in the embedding process, the extraction process is semi-blind. Hence, the embedding positions shall be transmitted separately from the sender to the receiver using a secured channel.

The watermark extraction can be summarized as follows:

Step 1. The audio segment $\tilde{B}$ containing the watermark is selected.

Step 2. Steps 1 and 2 of the embedding process are applied to the segment $\tilde{B}$ in order to obtain N_S frequency sub-bands.

Step 3. After selecting the sub-bands and the DWPT coefficients used in embedding process, the watermark bits are extracted. Each bit is extracted from each selected DWPT coefficient using Equation 7.

3.2.3 Watermark decryption

The extracted watermark $\tilde{W}$ from all the segments is finally decrypted using the Tent map generated with the private keys K_1, K_2 and K_3 saved in the embedding process. The original watermark $\tilde{W}_0$ is recovered by the following expression:

$$\tilde{w}_0(n) = u_b(n) \oplus \tilde{w}(n). \tag{22}$$

3.3 Blind extraction

As already discussed, the scheme proposed in this paper is semi-blind, since it requires sending the embedding positions to the receiver through a secured channel. Since the coefficients of the DWPT are either modified or remain unchanged according to a psychoacoustic model, the receiver does not have enough information to determine which coefficients have been modified during embedding, and which ones are kept unchanged.

Nevertheless, the proposed method can be modified for blind detection. A possible solution to this limitation is to select some coefficients of the audio signal that will remain unchanged during the embedding process. The remaining (non-fixed) coefficients can be first approximated by means of interpolation. If the interpolated value and the real one are similar within some threshold, the embedding equations can be applied using the interpolated coefficient instead of the real one. Since the interpolation operation can be reproduced by the receiver using the unchanged coefficients only, the extraction can be carried out without the necessity of transmitting the embedding positions. Of course, the embedding process must be carried out carefully in such a way that the receiver knows which specific interpolated coefficients have been used for embedding. To this aim, thresholds in the embedding process can be defined to avoid any ambiguity in the receiver side. This possibility, however, has not been implemented in this paper and is left for the future research.

3.4 Adjustable audio watermarking

By considering different capacities and levels of robustness, a large number of applications can be coveblack ranging from copyright protection to data transmission. Indeed, capacity and robustness are the main properties that define the type of application. For instance, watermarking for copyright protection requires a low capacity (few bits per second) and high level of robustness, contrasting with data transmission watermarking, which requires high capacity (thousands of bits per second) with less robustness. In addition, the imperceptibility of the watermark must be guaranteed regardless of the intended application. Thus, depending on the target application, a trade-off between imperceptibility, capacity and robustness must be attained.

In the proposed watermarking scheme, the following two parameters allow regulating the trade-off between the imperceptibility, capacity and robustness properties:

1. **Quantization step Δ:** as detailed above, watermark embedding is carried out by quantizing the DWPT coefficients using a DC-DM approach. A coarse quantization (high Δ value) of the coefficients results in better robustness and more distortion, whereas a fine quantization (low Δ value) leads to lower robustness and less distortion.
2. **Threshold τ:** this parameter is used to manage the imperceptibility-capacity trade-off. By decreasing the threshold, a high capacity can be reached, but at the cost of decreased imperceptibility. In fact, decreasing the threshold leads to increasing the margin of tolerance to distortion by including more DWPT co-

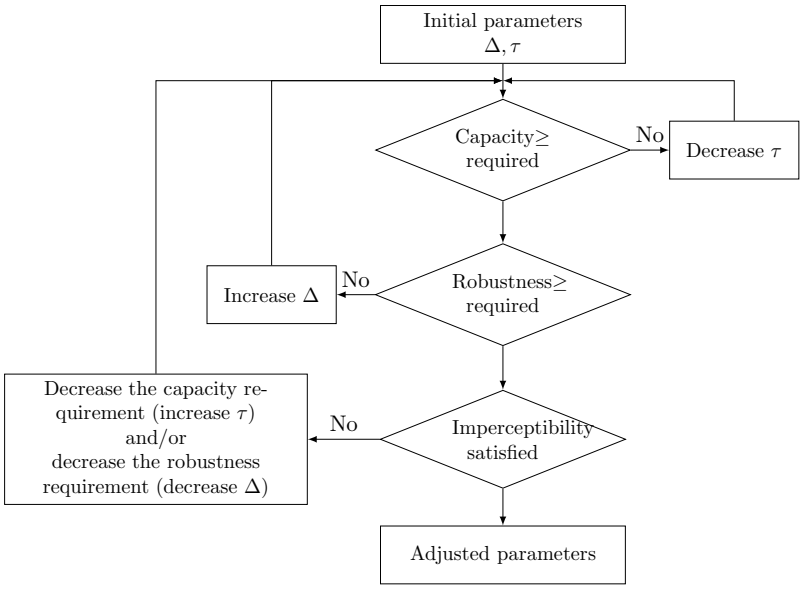


Fig. 7 Flowchart of the adjustment of the watermarking requirements

efficients for embedding in each sub-band, thus degrading the imperceptibility of the embedding distortion.

In order to achieve the desired results, a convenient adjustment of these two parameters must be carried out. Fig. 7 shows a flowchart detailing the tuning process for these parameters. It is worth pointing out that the proposed method automatically provides imperceptibility due to the use of the psychoacoustic model. Consequently, the tuning parameters will be used mainly to adjust the capacity/robustness trade-off according to the user's requirements. However, a final imperceptibility control is performed to ensure that there is no perceptible distortion. In case of perceptible distortion, the capacity and/or the robustness requirement is slightly reduced (by increasing τ and/or decreasing Δ) to obtain imperceptibility.

3.5 Security of the proposed system

The security of a watermarking scheme typically considers three different scenarios: unauthorized removal of the watermark, unauthorized detection of the watermark and unauthorized embedding of a new watermark. The typical way to obtain security, similarly as in cryptography, is the use of secret keys both in the embedding and the detection processes. The secret keys prevent the embedding positions from being totally deterministic, making the extraction of the watermark and the embedding of a new one very difficult or impossible for an attacker. In our work, the parameters Δ and τ determine the exact embedding positions, and these positions are transferred from the sender to the receiver through a secured channel. Without these secret positions, an attacker cannot remove the embedded watermark, embed a new watermark or extract the embedded watermark from a

Table 1 Test audio signals.

Source	Category	Audio signal	Duration (min:sec)
SQAM database	Speech	Male speech	0:22
	Vocal	Quartet	0:28
	Instrument	Trumpet	0:13
		Accordion	0:22
		Violin	0:29
	Orchestra	Choir	0:31
		Wind ensemble	0:18
Rust – "No, Really"	Full song	Molten	2:09
		Stop payment	2:09

marked content. In addition, without knowledge of the parameters Δ and τ, an attacker cannot even try to predict the embedding positions using the marked file. Thus, the secrecy of Δ, τ and the embedding positions provides security.

On the other hand, if the embedding positions were leaked to an eavesdropper, the attacker may succeed in removing the watermark. However, even in that case, the cryptographic keys K_1, K_2 and K_3 would prevent the extraction of a meaningful watermark –only the ciphertext would be available– or embedding a new one, since the attacker would not be able to generate a valid ciphertext without the cryptographic keys. Hence, the encryption of the watermark provides an additional layer of security against unauthorized extraction of the watermark or unauthorized embedding of a new watermark.

Finally, the possibility of using different synchronization codes would provide addtional security, as discussed in Section 3.1.2.

4 Experimental results

In this section, we present different experiments carried out to evaluate the performance of the proposed audio watermarking scheme.

A test corpus of nine audio signals of various styles, described in Table 1, is considered; each signal is in a waveform audio format file (WAVE), sampled with a frequency of 44.1 kHz and quantized with 16 bits per sample. We have selected 7 test signals from the EBU SQAM database[1] [45] specifically for the testing and evaluation of audio systems, and two full songs from the album Rust by No, Really [46]. The watermark used in the experiments is a binary sequence with enough bits to fill the whole host audio signal. The results are presented for three scenarios with different levels of capacity and robustness, namely:

- **Scenario 1:** Low capacity is considered ($<$ 100 bps). In this case, a high robustness against attacks can be reached.
- **Scenario 2:** A trade-off between robustness and capacity is considered in this scenario. Low robustness and capacity between 100 bps and 500 bps can be achieved in comparison to the first case.
- **Scenario 3:** By considering high capacity ($>$ 500 bps), lower robustness is obtained compared to the first and second scenarios. Nonetheless, the robustness remains acceptable.

[1] EBU: The European Broadcasting Union; SQAM: Sound Quality Material Assessment.

Table 2 The values of adjustable parameters Δ and τ with the corresponding capacity for different experiments.

Signal	Scenario	Δ	τ	Capacity (bps)
Male speech	1	0.9	20	70.54
		0.75	20	90.27
	2	0.7	16	158.59
		0.4	17	312.54
	3	0.2	19	681.81
		0.1	23	1075.50
Quartet	1	1.5	20	67.78
		1.3	20	91.64
	2	1	20	141.46
		0.4	23	340.21
	3	0.15	25	766.42
		0.08	27	1245.50
Trumpet	1	1.4	20	61.23
		1.1	20	98.30
	2	0.8	19	174.84
		0.35	19.5	399.22
	3	0.2	21	624.76
		0.07	28	1171.6
Accordion	1	0.8	18	59.50
		0.8	13.5	97.31
	2	0.7	10	180.77
		0.39	12	404.36
	3	0.15	20	761.86
		0.1	23	1083.9
Violin	1	0.6	20	56
		0.6	16	85.68
	2	0.5	12	197.68
		0.25	17	365.31
	3	0.14	20.5	622.58
		0.06	27	1118.3
Choir	1	2	25	72.19
		2	22	98.48
	2	1.5	23	153.22
		1.2	19	347.12
	3	0.8	18	677.74
		0.4	20	1328.1
Wind ensemble	1	1.4	20	65.22
		1.2	20	89.11
	2	0.8	19	224.38
		0.7	15	424.11
	3	0.5	15	721.05
		0.35	16	1101.3
Molten	1	2.1	20	80.54
		1.9	20	97.04
	2	1.2	20	221.82
		0.8	20	424.06
	3	0.35	24	870.02
		0.1	28	2538.7
Stop payment	1	2	20	69.26
		1.8	20	84.59
	2	1	20	237.88
		0.8	20	339.75
	3	0.2	27	1048.9
		0.1	27.5	2232.4

In our simulations, two experiments are performed in each scenario. Table 2 shows the values of the adjustable parameters (Δ and τ) with the corresponding capacity for the different experiments.

4.1 Audio quality assessment

An inaudible watermarking is undoubtedly the first constraint of an audio watermarking system. The audio quality of the proposed watermarking scheme has been assessed using SNR and perceptual evaluation of audio quality (PEAQ) [47, 48]. The SNR is computed by the following expression:

Table 3 Five-grade impairment scale used in ODG.

ODG	Description of impairments	Quality
0	Imperceptible	Excellent
−1	Perceptible, but not annoying	Good
−2	Slightly annoying	Fair
−3	Annoying	Poor
−4	Very annoying	Bad

Table 4 Capacity, SNR and ODG of two audio signals for all three scenarios.

Scenario	Violin			Stop payment		
	Capacity (bps)	SNR(dB)	ODG	Capacity (bps)	SNR(dB)	ODG
1	56.00	35.06	−0.18	69.26	44.50	−0.25
	85.68	33.11	−0.19	84.59	44.14	−0.25
2	197.68	28.76	−0.26	237.88	41.50	−0.23
	365.31	31.61	−0.30	339.57	40.67	−0.28
3	622.58	32.80	−0.52	1048.9	44.04	−0.25
	1118.3	34.57	−0.58	2232.4	43.41	−0.53

Table 5 Average of capacity, SNR and ODG for the remaining audio signals. Results are interpreted as mean [± standard deviation].

Scenario	Capacity (bps)	SNR(dB)	ODG
1	81.37 [± 14.28]	39.88 [± 3.62]	−0.075 [± 0.070]
2	279.08 [± 106.13]	36.03 [± 4.73]	−0.262 [± 0.153]
3	1046.30 [± 471.30]	36.54 [± 4.55]	−0.493 [± 0.133]

$$SNR = 10 \log \left(\frac{\sum_{i=1}^{L_X} x(i)^2}{\sum_{i=1}^{L_X} [x(i) - \hat{x}(i)]^2} \right). \tag{23}$$

The PEAQ renders an objective difference grade (ODG) ranging from −4 to 0 as shown in Table 3. In this work, the OPERA software [49], based on PEAQ Advanced, has been used to compute the ODG.

Table 4 presents capacity, SNR, and ODG values of two selected audio signals, namely, "Violin" and "Stop payment", for all three scenarios. The results show very good imperceptibility for all three scenarios. The ODG obtained for the two signals is in the range $[-0.58, 0]$ implying that the watermarked signals are perceptually indistinguishable from the original ones. Besides, the SNR of the watermarked audio signals is ranged between 28.76 dB and 45.32 dB, which is largely superior to the minimum recommended value (SNR higher than 20 dB [50]) proposed by the International Federation of Phonographic Industry (IFPI).

Table 5 provides the average of capacity, SNR and ODG values for the remaining audio signals with two different tunning settings per each scenario. Regardless of the embedding capacity (which could reach 2,500 bps), the audio quality of the proposed scheme remains steady.

The main reason of having the good imperceptibility results is due the exploitation of human auditory properties. In fact, by tuning the two regulator parameters, Δ and τ, we can easily maintain the distortion below the masking threshold, thus, achieving a good transparency of the watermarked signal. In addition, the use of DC-DM as an embedding technique, instead of a classical QIM, has also a positive effect in the imperceptibility of the proposed method. It should be noted that the compensated factor α has been set to 0.5 for all audio signals to achieve a trade-off between imperceptibility and robustness. An appropriate tuning of this parameter for each audio signal should further improve the performance of the proposed method.

4.2 Robustness

In this study, the bit error rate (BER) between the original and the recovered watermark, is examined to evaluate the robustness of the proposed method.

$$BER(W, \tilde{W}) = \frac{\sum_{i=1}^{L_W} w(i) \oplus \tilde{w}(i)}{L_W}, \tag{24}$$

where $\oplus$ stands for the exclusive OR operator.

4.2.1 Common signal processing attacks

A variety of common signal processing attacks have been considered including the following:

A. **Noise addition 30 dB:** White Gaussian noise is added to the watermarked audio signal until the resulting signal has an SNR of 30 dB.
B. **Noise addition 20 dB:** White Gaussian noise is added to the watermarked audio signal until the resulting signal has an SNR of 20 dB.
C. **Re-sampling:** The watermarked audio signal is down-sampled to 22.05 kHz and then up-sampled back to 44.1 kHz.
D. **Re-quantization 16-8-16:** The watermarked audio signal is quantized down to 8 bits/sample, and then re-quantized back to 16 bits/sample.
E. **Re-quantization 16-4-16:** The watermarked audio signal is quantized down to 4 bits/sample, and then re-quantized back to 16 bits/sample.
F. **Low-pass filtering:** A second-order Butterworth filter with cut-off frequency of 11 kHz is applied to the watermarked audio signal.
G. **MP3 compression 128 kbps:** MPEG-1 layer 3 compression/decompression at a bit rate of 128 kbps is applied to the watermarked audio signal.
H. **MP3 compression 64 kbps:** MPEG-1 layer 3 compression/decompression at a bit rate of 64 kbps is applied to the watermarked audio signal.
I. **Echo addition:** An echo signal with a delay of 10 ms and a decay of 10% is added to the watermarked audio signal.
J. **Amplitude scaling 90%:** The amplitude of the watermarked signal is scaled down to 90%.
K. **Amplitude scaling 110%:** The amplitude of the watermarked signal is scaled up to 110%.

Table 6 BER (%) of the proposed method against several attacks (corresponding to the imperceptibility results of Table 4).

Attacks	Violin						Stop payment					
	Scenario 1		Scenario 2		Scenario 3		Scenario 1		Scenario 2		Scenario 3	
A	0.000	0.000	0.000	0.000	0.039	0.025	0.000	0.000	0.000	0.000	0.002	0.062
B	0.000	0.000	0.000	0.019	0.050	0.089	0.000	0.000	0.000	0.000	0.099	5.327
C	0.000	0.000	0.000	0.066	0.072	0.059	0.000	0.000	0.020	0.055	0.543	1.003
D	0.000	0.000	0.000	0.019	0.039	0.040	0.000	0.000	0.000	0.000	0.000	0.043
E	0.000	0.000	0.000	0.009	0.039	0.025	0.000	0.000	0.000	0.000	0.007	0.160
F	0.000	0.000	0.000	0.028	0.078	0.289	0.000	0.000	0.000	0.000	0.247	1.770
G	0.000	0.000	0.000	0.094	0.430	1.381	0.000	0.000	0.056	0.089	1.725	4.388
H	0.000	0.000	0.070	0.245	0.897	3.068	0.000	0.055	0.288	0.433	5.290	10.015
I	0.000	0.000	0.000	0.057	0.294	1.539	0.000	0.000	0.026	0.053	2.090	4.716
J	0.000	0.000	0.000	0.057	0.487	1.443	0.000	0.000	0.013	0.053	1.823	4.636
K	0.000	0.000	0.000	0.075	0.338	1.381	0.000	0.000	0.000	0.005	1.727	5.426
L	0.000	0.000	0.000	0.000	0.061	0.108	0.450	0.433	0.537	0.490	0.495	0.567
M	0.000	0.282	1.047	2.435	4.758	9.263	0.135	0.258	2.048	3.331	12.827	18.786

L. **Cropping:** Three thousand samples are cropped from the watermarked signal at three different positions.
M. **Jittering:** One sample out of every 100,000 is removed.

Table 6 presents the watermark detection results against the described attacks for the two audio signals, namely "Violin" and "Stop payment", while the results for the remaining signals are given in Table 7. In the results, the three levels of robustness achieved with the proposed method are clearly noticeable. Scenario 1 exhibits very high robustness against all attacks with BER close to zero. This is because the signals are watermarked with a higher Δ on the premise of a good perceptual quality. Scenario 2 also provides high robustness against most of the attacks. Note that the cropping attack (row L) is significantly worse for the signals of the experiments in Table 7 compared to those of Table 6. This is due to the fact that most of the audio signals selected in the experiments of Table 7 are much shorter, and, hence, cropping 3,000 samples at three different positions results in more damage than the cropping of samples in longer audio signals.

Table 7 Average BER (%) for the remaining audio signals (corresponding to the imperceptibility results of Table 5).

Attacks	Scenario 1	Scenario 2	Scenario 3
A	0.000 [± 0.000]	0.000 [± 0.000]	0.080 [± 0.188]
B	0.000 [± 0.000]	0.001 [± 0.004]	0.926 [± 1.669]
C	0.004 [± 0.014]	0.177 [± 0.462]	0.694 [± 1.431]
D	0.000 [± 0.000]	0.001 [± 0.004]	0.076 [± 0.184]
E	0.000 [± 0.000]	0.001 [± 0.004]	0.110 [± 0.233]
F	0.000 [± 0.000]	0.054 [± 0.147]	0.983 [± 1.265]
G	0.011 [± 0.039]	0.121 [± 0.245]	2.076 [± 2.085]
H	0.045 [± 0.118]	0.376 [± 0.539]	4.438 [± 3.755]
I	0.000 [± 0.000]	0.104 [± 0.201]	2.298 [± 2.370]
J	0.000 [± 0.000]	0.141 [± 0.285]	2.293 [± 2.310]
K	0.000 [± 0.000]	0.057 [± 0.129]	2.214 [± 2.337]
L	9.538 [± 8.037]	9.275 [± 6.958]	9.074 [± 6.204]
M	0.171 [± 0.263]	1.816 [± 1.896]	8.761 [± 5.118]

Table 8 Variation of ODG and BER under different MP3 compression bit rates. The capacities considered for the two signals "Speech" and "Choir" are 1,075.5 bps and 1,328.1 bps, respectively.

MP3 bit rate (kbps)	Speech		Choir	
	ODG of attacked	BER (%)	ODG of attacked	BER (%)
192	0.00	1.581	0.00	0.000
128	−0.10	3.444	−0.23	0.000
96	−0.46	5.102	−0.57	0.019
80	−0.81	5.862	−0.78	0.033
64	−1.47	7.717	−1.16	0.086
48	−2.97	10.342	−3.48	11.185
32	−3.53	22.442	−3.54	31.061

For Scenario 3, even with a high capacity, the robustness of the proposed scheme is still acceptable with a BER below 10% except for desynchronization attacks. The watermarking system encounters some problems when dealing with cropping and jittering attacks in some audio signals. This is caused by the segmentation of the original audio signal into long segments, which decreases the number of embedded synchronization codes. In fact, by embedding more synchronization codes into the audio signals (i.e. considering shorter segments), the robustness against desynchronization attacks increases at the cost of a larger distortion in the audio signal. In this paper, it is considered preferable to sacrifice some embedding capacity as far as no perceptible distortion is detected in the watermarked audio signal.

4.2.2 MP3 compression with low bit rates

Table 8 presents the variation of ODG and BER, under different MP3 bit rates, for the audio signals "Speech" and "Choir". The tuning settings considered for these two signals correspond to Scenario 3 (high capacity, lower robustness) with capacities of 1,075.5 bps and 1,328.1 bps. The values of Δ and τ associated to these capacities are provided in Table 2.

The ODG reflects the distortion of the attacked file with respect to the marked one. It can be observed that, for bit rates below 64 kbps, the quality of the signal decreases and the resulting ODG is around −3 or worse, as already noticed in the scientific literature [51, 52]. This means that noise introduced in the resulting audio signal for bit rates lower than 64 kbps is between annoying and very annoying. In these conditions, it is completely reasonable that the proposed audio watermarking scheme, which is designed to provide imperceptibility guarantees (the embedding occurs only when the imperceptibility threshold is satisfied) can not maintain robustness. In any case, even for MP3 compression at 48 kbps, the resulting BER is around 10%, meaning that still 90% of the embedded bits can be recovered. This is quite a remarkable achievement for the proposed method that is designed to guarantee imperceptibility. For MP3 at 32 kbps, the BER is larger, but the ODG of the compressed signal is even below −3.5 (very annoying).

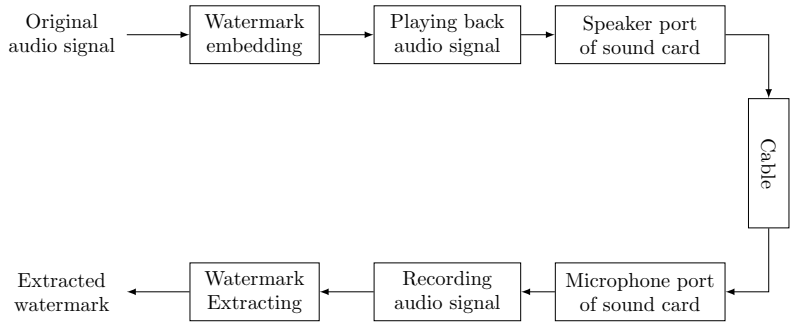


Fig. 8 Experimental model for the copy attack.

Table 9 BER (%) of the proposed method against copy attack.

Audio signal	Scenario 1		Scenario 2		Scenario 3	
Speech	0.000	2.216	5.605	7.795	8.300	9.011
Quartet	0.000	0.000	0.000	0.924	7.181	14.094
Trumpet	0.000	0.000	0.000	2.120	5.097	10.873
Accordion	0.000	0.000	0.000	0.124	1.211	2.462
Violin	0.000	0.000	0.035	0.925	2.077	3.762
Choir	0.000	0.000	0.000	0.009	0.119	1.921
Wind Ensemble	0.000	0.000	0.000	0.000	0.023	2.078
Molten	0.000	0.000	0.136	0.601	2.555	13.602
Stop Payment	0.000	0.000	0.124	0.763	5.378	12.243

4.2.3 Copy attack

In order to evaluate the robustness of the proposed system against the copy attack, we have used the experimental model presented in Fig. 8. The digital watermarked audio file is converted to an analog signal and, then, converted back to a digital signal using an IDT High Definition Audio CODEC sound card. The speaker output is connected to a microphone port using a 1-meter long 3.5 mm jack cable. Since the cable line may be considered as a clear environment, the distortion caused by this attack comes mainly from the digital-to-analog (DA) and analog-to-digital (AD) conversions. As discussed in [53], the degradation due to this conversion is the combination of volume change, additive noise and small time-scale modification (TSM).

Table 9 presents the robustness results (in terms of BER) against the copy attack for all nine selected audio signals. The tuning scenarios considered for these signals are provided in Table 2. The obtained results clearly exhibit the high robustness of the proposed system for both Scenario 1 and Scenario 2. For Scenario 3, the robustness remains acceptable and the BER does not exceed 14% in the worst case. These results show the robustness of the proposed audio watermarking scheme even in this difficult scenario.

Table 10 BER (%) of the proposed method against StirMark attacks (corresponding to the imperceptibility results of Table 4).

Attacks	Violin						Stop payment					
	Scenario 1		Scenario 2		Scenario 3		Scenario 1		Scenario 2		Scenario 3	
AddBrumm	2.771	2.978	3.087	1.491	1.468	1.856	0.022	1.610	2.758	2.490	2.088	3.788
AddDynNoise	0.000	0.000	0.000	0.038	0.377	1.338	0.000	0.000	0.010	0.062	2.106	5.866
AddFFTNoise	0.000	0.000	0.000	0.009	0.039	2.174	0.000	0.000	0.000	0.000	0.004	0.148
AddNoise	0.000	0.000	0.000	1.831	1.639	6.778	0.000	0.000	0.000	0.000	2.314	3.671
AddSinus	0.000	0.000	0.000	0.000	0.039	0.182	0.000	0.000	0.000	0.000	0.000	0.076
Amplify	0.062	0.080	0.122	1.104	3.412	6.87	0.000	0.009	0.802	1.498	9.298	13.381
BassBoost	0.000	0.040	0.16	0.595	1.15	2.134	1.157	1.353	1.809	2.089	4.579	6.421
BitChanger	0.000	0.000	0.000	0.000	0.039	0.022	0.000	0.000	0.000	0.000	0.000	0.024
Compressor	0.000	0.000	0.000	0.000	0.039	0.022	0.112	0.166	0.340	0.465	1.026	1.334
FFT_HLPassQuick	0.308	0.322	0.227	0.378	0.543	0.715	2.933	2.724	3.026	3.083	3.530	4.164
LSBZero	0.000	0.000	0.000	0.000	0.039	0.022	0.000	0.000	0.000	0.000	0.000	0.039
Noise_Max	0.000	0.000	0.000	0.028	0.166	6.778	0.000	0.000	0.000	0.000	0.005	0.738
RC_HighPass	0.739	0.644	0.715	1.55	3.118	5.418	2.708	2.613	2.899	3.292	6.731	8.834
RC_LowPass	0.000	0.000	0.000	0.019	0.044	0.040	0.000	0.000	0.000	0.000	0.013	0.323
ReplaceSamples	0.246	0.322	0.209	0.179	0.321	0.240	0.169	0.248	0.203	0.195	0.192	0.245
Smooth	0.000	0.000	0.000	0.302	1.545	4.554	25.778	25.808	24.199	24.203	25.307	27.699
Smooth2	0.000	0.000	0.000	0.066	0.310	1.243	0.000	0.000	0.000	0.025	0.593	2.160
Stat1	0.000	0.000	0.000	0.113	0.305	1.573	0.000	0.000	0.0622	0.087	2.562	6.431
Stat2	0.000	0.000	0.000	0.028	0.050	0.031	0.000	0.000	0.000	0.000	0.035	0.398

4.2.4 StirMark benchmark for audio

In addition to common signal processing attacks, the StirMark benchmark for audio [54] has also been applied to assess the robustness of the method. In the experiments, we have used StirMark for Audio version 0.2 with default parameters. Tables 10 and 11 show the detection results (BER) against 19 different StirMark attacks with the three tuning scenarios discussed above. As shown in Table 11, with the most robust scenario (Scenario 1), the proposed method achieves high robustness for almost all attacks. Only the *Smooth*, *Smooth2* and *Stat1* attacks seriously damage the embedded watermark for some of the tested signals (yielding BER larger than 20% for those signals). However, on average, the proposed scheme is robust (BER lower than 7%) for all the tested StirMark attacks.

The robustness of the proposed method could be improved in several ways. For example, the embedding of the watermark bits can be performed on group of coefficients instead of embedding one bit into each single coefficient. Such an approach should increase the robustness significantly, but at the price of reducing the embedding capacity. Furthermore, error correcting methods, such as Red-Solomon codes, could be integrated into our algorithm to enhance robustness against attacks.

The robustness of the proposed system can also be improved by using shift-invariance wavelet transforms. Indeed, the energy variation caused by a small shift in the input signal due to some attacks can probably affect the extraction of the watermark. Shift-invariance wavelet transforms can effectively increase the robustness against some attacks at price of a higher computational complexity.

4.3 Computational complexity

The computational complexity of the proposed scheme has been evaluated in terms of the CPU time required for the embedding and the extraction processes. The most intuitive way to assess the computational cost of the scheme is to compare the embedding and the extraction times, separately, with the playing time of the host audio signal. To determine the computational complexity, we have implemented

Table 11 Average of BER (%) against StirMark attacks for the rest of audio signals (corresponding to imperceptibility results of Table 5).

Attacks	Scenario 1	Scenario 2	Scenario 3
AddBrumm	1.603 [± 1.901]	2.371 [± 1.749]	2.742 [± 1.598]
AddDynNoise	0.004 [± 0.013]	0.080 [± 0.183]	2.186 [± 2.429]
AddFFTNoise	0.000 [± 0.000]	0.000 [± 0.000]	0.251 [± 0.413]
AddNoise	0.000 [± 0.000]	0.202 [± 0.481]	2.465 [± 2.362]
AddSinus	0.000 [± 0.000]	0.000 [± 0.000]	0.141 [± 0.249]
Amplify	0.269 [± 0.758]	1.358 [± 2.052]	8.497 [± 5.770]
BassBoost	0.754 [± 1.534]	1.208 [± 1.979]	3.416 [± 3.232]
BitChanger	0.000 [± 0.000]	0.000 [± 0.000]	0.066 [± 0.176]
Compressor	0.020 [± 0.041]	0.080 [± 0.115]	0.343 [± 0.450]
FFT_HLPassQuick	0.080 [± 1.459]	0.965 [± 1.598]	1.746 [± 2.461]
LSBZero	0.000 [± 0.000]	0.000 [± 0.000]	0.066[± 0.176]
Noise_Max	0.000 [± 0.000]	0.000 [± 0.000]	0.751 [± 1.169]
RC_HighPass	1.486 [± 2.358]	2.893 [± 4.798]	7.879 [± 6.882]
RC_LowPass	0.000 [± 0.000]	0.001 [± 0.004]	0.203 [± 0.316]
ReplaceSamples	0.293 [± 0.171]	0.248 [± 0.066]	0.301 [± 0.163]
Smooth	5.764 [± 14.262]	5.962 [± 14.113]	8.482 [± 13.914]
Smooth2	6.400 [± 15.682]	6.381 [± 15.594]	7.290 [± 15.574]
Stat1	6.545 [± 16.033]	6.640 [± 15.920]	8.568 [± 15.391]
Stat2	0.000 [± 0.000]	0.018 [± 0.064]	0.290 [± 0.608]

the watermark embedding and extraction processes in Matlab, on a PC with a 3.4 GHz Intel Core i7 processor with 8 GB of RAM.

Table 12 shows the results of the computational complexity for the three tuning scenarios, obtained after averaging the results for all nine selected audio signals. For these signals, the average playing time is 46.778 seconds. It can observed that the extraction process is much faster than the embedding counterpart. While the embedding process takes about 350% to 490% of the playing time, the extraction one only takes about 70% of the playing time. This difference is due to the use of the iterative process required by the embedding method, which is relatively costly from a computational point of view.

It can also be seen that the watermark length does not affect the computational complexity significantly in the detection process. The average of the detection times for the three different scenarios are similar. In fact, the search of the synchronization codes is considerably more complex than the extraction of the watermark bits. Consequently, the time required for the extraction of the watermark bits is insignificant compared to the time used in the detection of synchronization codes. It should be noted that the search of the synchronization codes has been implemented without any computational complexity optimization. The approach presented in [42], which optimizes the detection of the synchronization codes by limiting the number of checks, could be used to reduce the extraction time if required. However, even without optimization, the extraction time is lower than the playing time, which makes it possible to apply the proposed even in real time (i.e. to carry out watermark extraction while playing the file).

Table 12 Results of computational complexity.

Scenario	Watermark length (bit)	Embedding time (sec)	Extraction time (sec)
1	3803 [± 3766]	168.763 [± 168.747]	32.272 [± 32.231]
2	14541 [± 15505]	180.893 [± 176.645]	34.326 [± 33.317]
3	64866 [± 92236]	232.033 [± 301.970]	32.699 [± 32.884]

5 Comparative analysis

In order to validate the proposed scheme, a comparative study is performed with recent audio watermarking systems proposed respectively by Lei el al. [8], Peng and Wang [25], Mohsenfar et al. [55] and Al-Haj [56]. These schemes have been selected because they all use a semi-blind watermark detection. In order to establish a fair and objective comparison, the second scenario, detailed in Section 4, has been selected for the proposed system, hence the notation "Proposed 2" is used here. This scenario has been selected to match the capacity of the other systems, which lies in the range $[100, 500)$ bps. The values given in this section have been computed by averaging all the results obtained for Scenario 2 with all nine selected audio signals.

Table 13 Comparison of the proposed scheme with other semi-blind watermarking schemes in terms of capacity and imperceptibility.

Method	Capacity (bps)	SNR(dB)	ODG
Lei et al. [8]	256	42.51	–
Peng and Wang [25]	204.8	–	–
Mohsenfar et al. [55]	159	25.89	−0.57
Al-Haj [56]	258	38.17	−0.76
Proposed 2	**280.43**	**35.95**	**−0.26**

Table 13 shows a comparison results between the proposed scheme and other methods in terms of capacity and imperceptibility, while Table 14 reports the comparison in terms of robustness against different attacks. From these results, it can be seen that the proposed scheme outperforms the selected schemes with respect to capacity. The proposed scheme also shows high imperceptibility results, thus, outperforming the schemes [55, 56] with a better ODG. Unfortunately, the schemes [8, 25] do not report the imperceptibility results in terms of ODG, which makes it more difficult to establish a completely fair comparison with them. Nevertheless, the SNR of the proposed scheme remains comparable with that of scheme [8]. Regarding the robustness, the proposed method presents competitive results compared to other methods. Note that the parameters used in some attacks are not the same for all the schemes, making the comparison not completely fair.

Finally, it should be pointed out that, in contrast to the analyzed methods, the proposed scheme can be easily tuned to different levels of capacity and robustness, while maintaining high imperceptibility. The proposed scheme is remarkably robust for low capacity, but it can also reach high capacity with remarkable robustness.

Table 14 Comparison of the proposed scheme with other semi-blind watermarking schemes in terms of robustness.

Method	BER (%)				
	Noise addition	Re-quantization	Low-pass filtering	MP3 compression	Amplitude scaling
Lei et al. [8]	0 (20 dB)	0	0 (6 kHz)	0 (64 kbps)	5.1 (110%)
Peng and Wang [25]	3.47 (20 dB)	0	3.44 (4 kHz)	2.79 (64 kbps)	3.64 (150%)
Mohsenfar et al. [55]	0 (30 dB)	–	0 (9 kHz)	7 (64 kbps)	0.84 (–)
Al-Haj [56]	0 (20 dB)	0	0.19 (8 kHz)	0.07 (64 kbps)	0 (150%)
Proposed 2	**0** (20 dB)	**0**	**0.04** (11 kHz)	**0.35** (64 kbps)	**0.05** (110%)

6 Conclusion

A novel adjustable audio watermarking scheme is presented in this paper. The proposed scheme exploits the benefits of the DWPT, psychoacoustic modeling and the DC-DM quantization. The original audio signal is firstly segmented into long segments and each of them is partitioned into two parts. Then, a synchronization code is embedded in the time domain of the first part, whereas the watermark information is embedded into the coefficients of the DWPT of the second part. An algorithm has been developed to optimize the number of embedded bits in each frequency sub-band based on the masking threshold derived from a psychoacoustic model. Instead of using the classical QIM embedding method, this paper adopts the DC-DM embedding technique to improve both imperceptibility and robustness. Furthermore, a chaotic Tent map is applied to encrypt the watermark, which enhances the security of the proposed scheme. In the extraction phase, once the synchronization mark is found, the watermark is extracted from the appropriate DWPT coefficients. Since we need the positions of these coefficients, the proposed watermarking method is semi-blind.

The proposed method has been evaluated using different audio signals selected from both SQAM database and two songs of a pop music album. The imperceptibility and robustness properties of the scheme have been assessed using the PEAQ Advanced standard and the BER of the extracted watermark under various signal processing attacks, respectively. The results show that this method provides high audio quality (the ODG average of all the tested signals is -0.30) with a high robustness against most attacks. The proposed scheme can also reach high capacity (about 2,500 bps) while maintaining reasonable robustness. In addition, the system provides two regulator parameters facilitating the adjustment of the capacity-robustness trade-off. The comparison of the proposed method with other semi-blind schemes shows competitive results in terms of capacity, transparency and robustness.

The main limitation of the proposed method is the fact that the extraction process is semi-blind. Future work will focus on designing a completely blind system that does not require the transmission of the embedding positions to the receiver. In addition, also as future research, we envisage carrying out subjective audio tests to confirm the imperceptibility results obtained in the experiments using the advanced PEAQ standard.

Acknowledgements This work was partly supported by the Algerian Ministry of Higher Education and Scientific Research under the grants MESRS-FNR-2013-2016 and CNEPRU J02002201000031.

The third author of this work is partly funded by the Spanish Government through grants TIN2011-27076-C03-02 “CO-PRIVACY” and TIN2014-57364-C2-2-R “SMARTGLACIS”.

The authors thank Dr. Amna Qureshi for the proof-reading of the manuscript and the corrections she has suggested.

References

1. Petitcolas FA, Anderson RJ, Kuhn MG (1999) Information hiding-a survey. Proceedings of the IEEE 87(7):1062–1078
2. Cox IJ, Miller ML, Bloom JA, Honsinger C (2002) Digital watermarking, vol 53. Springer
3. Podilchuk C, Delp EJ, et al (2001) Digital watermarking: algorithms and applications. Signal Processing Magazine, IEEE 18(4):33–46
4. Arnold M (2000) Audio watermarking: Features, applications, and algorithms. In: IEEE International Conference on Multimedia and Expo (II), Citeseer, pp 1013–1016
5. Liu Z, Inoue A (2003) Audio watermarking techniques using sinusoidal patterns based on pseudorandom sequences. Circuits and Systems for Video Technology, IEEE Transactions on 13(8):801–812
6. Cvejic N, Seppänen T (2004) Spread spectrum audio watermarking using frequency hopping and attack characterization. Signal processing 84(1):207–213
7. Valizadeh A, Wang ZJ (2011) Correlation-and-bit-aware spread spectrum embedding for data hiding. Information Forensics and Security, IEEE Transactions on 6(2):267–282
8. Lei B, Zhou F, Tan EL, Ni D, Lei H, Chen S, Wang T (2015) Optimal and secure audio watermarking scheme based on self-adaptive particle swarm optimization and quaternion wavelet transform. Signal Processing 113:80–94
9. Yeo IK, Kim HJ (2003) Modified patchwork algorithm: A novel audio watermarking scheme. Speech and Audio Processing, IEEE Transactions on 11(4):381–386
10. Kalantari NK, Akhaee MA, Ahadi SM, Amindavar H (2009) Robust multiplicative patchwork method for audio watermarking. Audio, Speech, and Language Processing, IEEE Transactions on 17(6):1133–1141
11. Natgunanathan I, Xiang Y, Rong Y, Zhou W, Guo S (2012) Robust patchwork-based embedding and decoding scheme for digital audio watermarking. Audio, Speech, and Language Processing, IEEE Transactions on 20(8):2232–2239
12. Natgunanathan I, Xiang Y, Rong Y, Peng D (2014) Robust patchwork-based watermarking method for stereo audio signals. Multimedia Tools and Applications 72(2):1387–1410
13. Ko BS, Nishimura R, Suzuki Y (2005) Time-spread echo method for digital audio watermarking. Multimedia, IEEE Transactions on 7(2):212–221
14. Chen OT, Wu WC (2008) Highly robust, secure, and perceptual-quality echo hiding scheme. Audio, Speech, and Language Processing, IEEE Transactions on 16(3):629–638
15. Erfani Y, Siahpoush S (2009) Robust audio watermarking using improved TS echo hiding. Digital Signal Processing 19(5):809–814
16. Xiang Y, Natgunanathan I, Peng D, Zhou W, Yu S (2012) A dual-channel time-spread echo method for audio watermarking. Information Forensics and Security, IEEE Transactions on 7(2):383–392

17. Xiang S, Huang J (2007) Histogram-based audio watermarking against time-scale modification and cropping attacks. Multimedia, IEEE Transactions on 9(7):1357–1372
18. Xiang S, Kim HJ, Huang J (2008) Audio watermarking robust against time-scale modification and MP3 compression. Signal Processing 88(10):2372–2387
19. Bhat V, Sengupta I, Das A (2011) An audio watermarking scheme using singular value decomposition and dither-modulation quantization. Multimedia Tools and Applications 52(2-3):369–383
20. Bhat V, Sengupta I, Das A (2010) An adaptive audio watermarking based on the singular value decomposition in the wavelet domain. Digital Signal Processing 20(6):1547–1558
21. Singh J, Garg P, De AN (2012) Audio watermarking based on quantization index modulation using combined perceptual masking. Multimedia Tools and Applications 59(3):921–939
22. Wang XY, Qi W, Niu P (2007) A new adaptive digital audio watermarking based on support vector regression. Audio, Speech, and Language Processing, IEEE Transactions on 15(8):2270–2277
23. Lei B, Soon Y, Zhou F, Li Z, Lei H (2012) A robust audio watermarking scheme based on lifting wavelet transform and singular value decomposition. Signal Processing 92(9):1985–2001
24. Wang X, Wang P, Zhang P, Xu S, Yang H (2013) A norm-space, adaptive, and blind audio watermarking algorithm by discrete wavelet transform. Signal Processing 93(4):913–922
25. Peng H, Wang J (2011) Optimal audio watermarking scheme using genetic optimization. Annals of Telecommunications-Annales des Télécommunications 66(5-6):307–318
26. Chen B, Wornell GW (2001) Quantization index modulation: a class of provably good methods for digital watermarking and information embedding. Information Theory, IEEE Transaction on 47(4):1423–1443
27. Tsai HH, Cheng JS, Yu PT (2003) Audio watermarking based on HAS and neural networks in DCT domain. EURASIP Journal on Advances in Signal Processing 2003(3):1–12
28. Hu HT, Hsu LY, Chou HH (2014) Perceptual-based DWPT-DCT framework for selective blind audio watermarking. Signal Processing 105:316–327
29. Hu HT, Chou HH, Yu C, Hsu LY (2014) Incorporation of perceptually adaptive QIM with singular value decomposition for blind audio watermarking. EURASIP Journal on Advances in Signal Processing 2014(1):1–12
30. Pinel J, Girin L, Baras C, Parvaix M (2010) A high-capacity watermarking technique for audio signals based on MDCT–domain quantization. In: Proceedings of the International Congress on Acoustics (ICA), Sydney, Australia
31. Fu Z, Zhang P, Huang W, Wang L, Emmanuel S, Chen G (2015) Empirical mode decomposition based blind audio watermarking. Multimedia Tools and Applications 74(15):6019–6040
32. Fallahpour M, Megías D (2014) Secure logarithmic audio watermarking scheme based on the human auditory system. Multimedia Systems 20(2):155–164
33. Abbate A, DeCusatis C, Das PK (2012) Wavelets and subbands: fundamentals and applications. Springer Science & Business Media
34. Chan Y (2012) Wavelet basics. Springer Science & Business Media

35. Pesquet JC, Krim H, Carfantan H (1996) Time-invariant orthonormal wavelet representations. IEEE transactions on signal processing 44(8):1964–1970
36. Kingsbury N (2001) Complex wavelets for shift invariant analysis and filtering of signals. Applied and computational harmonic analysis 10(3):234–253
37. Cohen I, Raz S, Malah D (1997) Orthonormal shift-invariant wavelet packet decomposition and representation. Signal Processing 57(3):251–270
38. ISO/IEC JTC1/SC29/WG11 MPEG I (1993) Information technology - coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbit/s, part 3: Audio
39. Painter T, Spanias A (2000) Perceptual coding of digital audio. Proceedings of the IEEE 88(4):451–515
40. Yoshida T, Mori H, Shigematsu H (1983) Analytic study of chaos of the tent map: band structures, power spectra, and critical behaviors. Journal of statistical physics 31(2):279–308
41. Wu S, Huang J, Huang D, Shi YQ (2005) Efficiently self-synchronized audio watermarking for assured audio data transmission. Broadcasting, IEEE Transactions on 51(1):69–76
42. Megías D, Serra-Ruiz J, Fallahpour M (2010) Efficient self-synchronised blind audio watermarking system based on time domain and FFT amplitude modification. Signal Processing 90(12):3078–3092
43. Yang Hy, Wang Xy, Ma Tx (2011) A robust digital audio watermarking using higher-order statistics. AEU-International Journal of Electronics and Communications 65(6):560–568
44. Wang X, Shi Q, Wang S, Yang H (2016) A blind robust digital watermarking using invariant exponent moments. AEU-International Journal of Electronics and Communications
45. Waters G (1988) Sound quality assessment materialr recordings for subjective tests: Users handbook for the EBU–SQAM compact disk. European Broadcasting Union (EBU), Tech Rep
46. (????) No, really. rust. https://www.jamendo.com/album/7365/rust, last accessed on October 5, 2016
47. BS1387 IRR (1998) Methode for objective measurements of perceived audio quality
48. Kabal P (2002) An examination and interpretation of ITU-R BS. 1387: Perceptual evaluation of audio quality. TSP Lab Technical Report, Dept Electrical & Computer Engineering, McGill University pp 1–89
49. (????) Opera software by opticom. http://www.opticom.de/products/opera-demoversion.html, last accessed on October 5, 2016
50. Katzenbeisser S, Petitcolas F (2000) Information hiding techniques for steganography and digital watermarking. Artech house
51. Salovarda M, Bolkovac I, Domitrovic H (2005) Estimating perceptual audio system quality using PEAQ algorithm. In: Applied Electromagnetics and Communications, 2005. ICECom 2005. 18th International Conference on, IEEE, pp 1–4
52. Vercellesi G, Vitali A, Zerbini M (2007) MP3 audio quality for single and multiple encoding. In: Multimedia and Expo, 2007 IEEE International Conference on, IEEE, pp 1279–1282
53. Xiang S (2011) Audio watermarking robust against D/A and A/D conversions. EURASIP Journal on Advances in Signal Processing 2011:3

54. Lang A (????) Stirmark benchmark for audio (smba): Evaluation of watermarking schemes for audio. http://omen.cs.uni-magdeburg.de/alang/smba.php, last accessed on February 5, 2017
55. Mohsenfar SM, Mosleh M, Barati A (2015) Audio watermarking method using QR decomposition and genetic algorithm. Multimedia Tools Applications 74(3):759–779
56. Al-Haj A (2014) An imperceptible and robust audio watermarking algorithm. EURASIP Journal on Audio, Speech, and Music Processing 2014(1):1–12