Article

Blind Audio Watermarking Based on Parametric Slant-Hadamard Transform and Hessenberg Decomposition

Pranab Kumar Dhar ^{1,*}, Azizul Hakim Chowdhury ¹ and Takeshi Koshiba ²

¹ Department of Computer Science and Engineering, Chittagong University of Engineering and Technology (CUET), Chattogram 4349, Bangladesh; azizul.cuet.cse.31@gmail.com

² Faculty of Education and Integrated Arts and Sciences, Waseda University, 1-6-1 Nishiwaseda, Shinjuku-ku, Tokyo 169-8050, Japan; tkoshiba@waseda.jp

* Correspondence: pranabdhar81@cuet.ac.bd

Received: 26 January 2020; Accepted: 21 February 2020; Published: 26 February 2020

Abstract: Digital watermarking has been widely utilized for ownership protection of multimedia contents. This paper introduces a blind symmetric audio watermarking algorithm based on parametric Slant-Hadamard transform (PSHT) and Hessenberg decomposition (HD). In our proposed algorithm, at first watermark image is preprocessed to enhance the security. Then, host signal is divided into non-overlapping frames and the samples of each frame are reshaped into a square matrix. Next, PSHT is performed on each square matrix individually and a part of this transformed matrix of size $m \times m$ is selected and HD is applied to it. Euclidean normalization is calculated from the 1st column of the Hessenberg matrix, which is further used for embedding and extracting the watermark. Simulation results ensure the imperceptibility of the proposed method for watermarked audios. Moreover, it is demonstrated that the proposed algorithm is highly robust against numerous attacks. Furthermore, comparative analysis substantiates its superiority among other state-of-the-art methods.

Keywords: audio watermarking; copyright protection; parametric Slant-Hadamard transform; Hessenberg decomposition; Euclidean normalization

1. Introduction

Nowadays, the significant improvement of the internet makes it possible to easily access different multimedia data. Thus, various types of new challenges related to copyright protection and content tempering are introduced every day. Digital watermarking has been effectively utilized to tackle these new challenges. It is a process of embedding secret information into digital contents for authenticity. The major applications of digital watermarking include data authentication, fingerprinting, copyright protection, ownership protection, and broadcast monitoring [1]. The primal requirements of watermarking methods are (i) imperceptibility (ii) robustness, (iii) data payload, and (iv) security [2]. The imperceptibility property of a watermarking algorithm defines the indistinguishability between the host signal and watermarked signal. The robustness property of a watermarking algorithm is the ability to sustain the watermark against numerous signal processing attacks. Data payload of a watermarking algorithm defines the number of watermark bits that are embedded into the host signal. Security of a watermarking algorithm ensures that a watermark can be detectable only by an authorized person. The main challenge of a watermarking algorithm is to maintain a good trade-off among these requirements. In general, digital watermarking can be classified by different properties. On the basis of robustness property, digital watermarking can be classified into robust and fragile (or semi-fragile) watermarking. Moreover, the watermarking methods can be classified into blind,

semi-blind, and non-blind. While blind watermarking method can detect the watermark without the host signal, the non-blind method requires the host signal to extract the watermark and semi-blind method needs some information of host signal to extract the watermark. In this paper, we introduce a blind symmetric audio watermarking algorithm using a parametric Slant-Hadamard transform (PSHT), Hesssenberg decomposition (HD), and Euclidean normalization, which provides a good trade-off among imperceptibility, robustness, and data payload.

The remainder of this paper is organized as follows. Section 2 provides the related research that includes a brief summary of recent methods. Section 3 briefly describes the background information including PSHT and HD. Section 4 introduces the proposed watermarking method consisting of watermark preprocessing, watermark embedding, and extraction processes. Section 5 provides the experimental results and compares the performance of the proposed method with recent methods in terms of imperceptibility and robustness. Finally, in Section 6, the conclusion of this paper is presented.

2. Related Research

An extensive survey on audio watermarking techniques is described in [1,2]. According to the domain, watermarking is classified into time domain and transform domain techniques. Time domain techniques embed a watermark into the audio signal by modifying its coefficients directly [3]. This technique is easy to implement and requires few computational resources. On the other hand, the transform domain technique is applied to coefficients obtained as the result of transformation of either a whole audio or the frame of the audio. Some well-known and conventional transform domain techniques are discrete wavelet transform (DWT) [4], discrete cosine transform (DCT) [5], and fast fourier transform (FFT) [6]. Pandey et al. [3] presented a method that uses the pseudo-random gray sequence property. However, the imperceptibility result of this method is not quite high and robustness result is provided for very few attacks. Kaur et al. [4] suggested a method based on a mathematical model by using features such as energy, short time energy, and zero cross means, but robustness against some attacks is quite low. Tsai et al. [5] proposed a watermarking method based on energy averaging. However, the data payload of this method is not reported there. In [6], a watermarking scheme was proposed based on Lucas regular sequence (LRS) and FFT. However, this scheme shows less robustness against some of the common attacks. Dhar et al. [7] proposed a DCT-based algorithm using singular value decomposition (SVD) and exponential-log operations (ELO) where the watermark is embedded to the highest power of DCT coefficients, but robustness results against some common attacks were not reported. Karnjana et al. [8] introduced a method based on singular spectrum analysis (SSA) and psychoacoustic model (PM), but it shows quite low robustness against some common attacks. The authors of [9] proposed a multifunctional algorithm based on chaotic scrambling. However, the peak signal-to-noise ratio (PSNR) of this method is quite low. In [10], authors proposed a method based on DCT, singular value decomposition (SVD), entropy, and log-polar transformation (LPT). It shows good results for imperceptibility, but it does not show good robustness results against some common signal processing attacks. Hwang et al. [11] introduced a watermarking method based on quantization index modulation (QIM) and SVD, but the imperceptibility and robustness of this scheme is a little low. In [12], a watermarking method is proposed based on flexible segmentation (FS) and adaptive embedding (AE), but it provides low SNR and low robustness against some common attacks. Hu et al. [13] suggested a method in dual domain using flexible segmentation and adaptive embedding where binary watermark bits are inserted into discrete wavelet packet transform coefficients. However, it shows slightly poor results for imperceptibility. In [14], the authors introduced a watermarking algorithm using DWT and direct-sequence spread spectrum (DSSS). However, the robustness result of this method against some attacks is quite low. Irawati et al. [15] presented a method based on DCT and QR decomposition. The SNR of this method ranges between 11 dB to 27 dB, which is much lower than the basic requirement, and the bit error rate (BER) against some attacks is also quite high. Gupta et al. [16] suggested a watermarking method using lifting wavelet transform (LWT) and adaptive quantization. Although this method is blind, the SNR and normalized correlation (NC) of

this method is poor. In [17], a watermarking scheme is proposed using audio characteristics and scrambling encryption. This scheme shows high security; however, it has low robustness against some attacks. In [18], a watermarking scheme is suggested using empirical mode decomposition (EMD) where intrinsic feature of final residual is used to embed the watermark. It shows good robustness, but the objective listening test was not performed and data payload was also not reported there. Safitri et al. [19] presented a method using DWT, SVD, and BCH code where watermark bits are inserted using QIM. However, the PSNR of this method is little low and also robustness against some attacks was not conducted. A histogram-based audio watermarking using stationary wavelet transform (SWT) and synchronization is suggested in [20]. However, the data payload of this method is quite low and BER of this method against some attacks is quite high. An audio watermarking method using phase shifting is introduced in [21]. However, the PSNR result of this method is not reported and the robustness result against some attacks is quite low. From the above studies, we observed that some methods have low robustness, whereas some methods have less imperceptible or low data payload. To overcome the limitations stated above, in this paper, we suggest a blind symmetric audio watermarking algorithm based on PSHT, HD, and Euclidean normalization. To the best of our knowledge, this is the first audio watermarking algorithm that utilizes PSHT, HD, and Euclidean normalization jointly. The main features of the proposed algorithm are as follows: (i) it applies PSHT, HD, and Euclidean normalization unitedly; (ii) the logistic map is used for scrambling the watermark to safeguard the unauthorized detection; (iii) it embeds watermark into the largest value of the 1st column of Hessenberg matrix using a new embedding equation; (iv) watermark is extracted without the host signal; (v) it ensures the trade-off among imperceptibility, robustness, and data payload. Simulation results demonstrated that our proposed method is highly robust against numerous attacks. The BER of the proposed method varies from 0 to 6.54, whereas the BER of the recent methods [4–12] vary from 0 to 17.76. The PSNR of the proposed method varies from 43.81 to 47.75, whereas PSNR of the recent methods vary from 19.39 to 44.81. In other word, the proposed method outperforms state-of-the-art methods in terms of robustness and imperceptibility.

3. Background Information

3.1. Parametric Slant-Hadamard Transform (PSHT)

Parametric Slant-Hadamard transform (PSHT) was introduced by Agaian and this method is mostly used for signal processing [22]. PSHT mainly includes some parameters for which the fidelity, robustness, and imperceptibility property varies. Let f denote the original signal and F denote the transformed signal. Then, two-dimensional PSHT can be described as:

$$F = S_{2^n} f S_{2^n}^T, \quad (1)$$

where S_{2^n} represents a $2n \times 2n$ parametric slant-Hadamard matrix with real elements. The inverse transform to recover f from the transformed matrix F is given by:

$$f = S_{2^n}^T F S_{2^n}. \quad (2)$$

The parametric slant-Hadamard matrix with order 2^n is obtained from the matrix of order 2^{n-1} with the help of Kronecker product operator given as:

$$S_{2^n} = \frac{1}{\sqrt{2}} Q_{2^n} (I_2 \otimes S_{2^{n-1}}), \quad n > 1, \quad (3)$$

where I_2 represents the identity matrix of order 2 and Q_{2^n} denotes the matrix of recursion kernel property. Q_{2^n} can be described as follows:

$$Q_{2^n} = \begin{bmatrix} 1 & 0 & \vdots & 0_{2^{n-1}-2} & \vdots & 1 & 0 & \vdots & 0_{2^{n-1}-2} \\ a_{2^n} & b_{2^n} & \vdots & \vdots & \vdots & -a_{2^n} & b_{2^n} & \vdots & \vdots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0_{2^{n-1}-2} & \vdots & I_{2^{n-1}-2} & \vdots & \vdots & 0_{2^{n-1}-2} & \vdots & I_{2^{n-1}-2} & \vdots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 1 & \vdots & 0_{2^{n-1}-2} & \vdots & 0 & 1 & \vdots & 0_{2^{n-1}-2} \\ -b_{2^n} & a_{2^n} & \vdots & \vdots & \vdots & b_{2^n} & a_{2^n} & \vdots & \vdots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0_{2^{n-1}-2} & \vdots & I_{2^{n-1}-2} & \vdots & \vdots & 0_{2^{n-1}-2} & \vdots & -I_{2^{n-1}-2} & \vdots \end{bmatrix} \quad (4)$$

where O_M denotes the all-zero matrix of size $M \times M$ and $\otimes$ denotes the Kronecker product [22].

The parameters a_{2^n} and b_{2^n} are defined as:

$$a_{2^n} = \sqrt{\frac{3(2^{2n-2})}{4(2^{2n-2}) - \beta_{2^n}}} \text{ and } b_{2^n} = \sqrt{\frac{2(2^{2n-2}) - \beta_{2^n}}{4(2^{2n-2}) - \beta_{2^n}}} \quad (5)$$

The PSHT can be categorized into four groups based on the value of parameter β :

- (i) for all $\beta_{2^n} = 1$, it represents the classical slant transform;
- (ii) for $\beta_{2^n} = 2^{2n-2}$ and $n > 1$, it represents the Walsh-Hadamard transform;
- (iii) for $\beta_4 = \beta_8 = \dots = \beta_{2^n} = \beta$, and $|\beta| \leq 4$, it represents the constant beta slant transform;
- (iv) for $\beta_4 \neq \beta_8 \neq \dots \neq \beta_{2^n}$, $-2^{2n-2} \leq \beta_{2^n} \leq 2^{2n-2}$ and $n = 2, 3, 4, \dots$, it represents the multiple beta slant transform.

3.2. Hessenberg Decomposition (HD)

The Hessenberg decomposition (HD) decomposes a general square matrix A into the following form:

$$A = PHP \quad (6)$$

where P denotes orthogonal matrix and H is an upper triangular matrix [23].

4. Proposed Watermarking Algorithm

Let $Y = \{y(n), 1 \leq n \leq S\}$ be the host signal containing S samples and $W = \{w(k, l), 1 \leq k \leq M, 1 \leq l \leq M\}$ represent the binary watermark image. Let $w(k, l) \in (0, 1)$ be the pixel value at the point (k, l) that will be embedded into the host audio.

4.1. Watermark Preprocessing

For the enhancement of confidentiality, at first, a watermark should be preprocessed. The proposed method uses a logistic map which encompasses the chaotic characteristic to encrypt the binary watermark image and this feature will ensure the confidentiality of the proposed method. The mapping is defined as follow:

$$y(i+1) = \begin{cases} a \times y(i) \times (1 - y(i)), & \text{if } y(i) > 0 \\ a \times y(i) \times (1 - y(i)) + b, & \text{otherwise} \end{cases} \quad (7)$$

where $y(1) \in (0,1)$ and a, b are real parameters according to the map's initial condition. After this, a binary sequence is obtained with the help of the following equation:

$$z(i) = \begin{cases} 1, & \text{if } y(i) > T \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

where T represents a predefined threshold value, which depends on the real parameters a and b . Moreover, T is proportional to a and b , i.e., as the values of a and b increase, the value of T also increases and vice versa.

The original binary watermark image W is converted into an one dimensional sequence r , where $r = \{r(i), i = 1, 2, 3, \dots, M \times M\}$. Then, in the final stage of preprocessing, $r(i)$ is encrypted using $z(i)$ with the help of the following equation:

$$u(i) = z(i) \oplus r(i), \quad 1 \leq i \leq M \times M \quad (9)$$

where $\oplus$ is the exclusive-or (XOR) operation. After this encryption process, $u(i)$ cannot be found through random search. In this process, $y(1)$, a , and b can be used as a secret key K . The pseudo code of the watermark preprocessing is presented in Algorithm 1.

Algorithm 1: Watermark Preprocessing

Variable Declaration:

$W (i = 1, 2, \dots, M; j = 1, 2, \dots, M)$: the watermark image

$y(i+1) (i = 1, 2, \dots, M \times M)$: logistic mapping parameter

a, b : real parameters

$z(i) (i = 1, 2, \dots, M \times M)$: binary sequence

T : predefined threshold value

$r(i) (i = 1, 2, \dots, M \times M)$: new one dimensional sequence from W

$u(i)$: encrypted watermark sequence

Watermark Preprocessing Procedure:

Let $y(1) \in (0,1)$

for $i = 1: M$ do

calculate $y(i+1)$ using Equation (7)

calculate $z(i)$ using Equation (8)

calculate $u(i)$ using Equation (9)

end for

return encrypted watermark sequence

4.2. Watermark Embedding Process

The proposed watermark embedding procedure is shown in Figure 1 and is described as follows:

1. The host signal Y is firstly divided into $M \times M$ non-overlapping frames $F = \{F_1, F_2, F_3, \dots, F_{M \times M}\}$ and each frame F_i is converted into two-dimensional matrix C_i of size $m \times m$, where i represents the frame number.

2. PSHT is applied on each matrix C_i and transformed matrix T_i is obtained.

3. Then, each transformed matrix T_i is sub-divided into N non-overlapping blocks $B = \{B_j, 1 \leq j \leq N\}$ of size $n \times n$ and sum of the absolute mean of each block is calculated using the following equation:

$$Z_j = \sum_{k=1}^n \sum_{l=1}^n \frac{|B_j|}{n \times n} \text{ where } 1 \leq j \leq N \quad (10)$$

where $|B_j|$ denotes the absolute value of the j^{th} block B_j and Z_j denotes the absolute mean of the j^{th} block.

4. Find $Z_{max} = \max\{Z_1, Z_2, Z_3, \dots, Z_N\}$ of the blocks $\{B_1, B_2, B_3, \dots, B_N\}$, where max operation returns the largest value in $\{Z_1, Z_2, Z_3, \dots, Z_N\}$.

5. The Z_{max} is selected for decomposition and for simplicity it is represented as R_i . HD is then performed on the selected $n \times n$ matrix R_i , which is represented by:

$$R_i = P_i \times H_i \times P_i^T \quad (11)$$

where P_i denotes the orthogonal matrix and H_i denotes the Hessenberg matrix.

6. Euclidean normalization of the 1st column of the Hessenberg matrix H_i is calculated using the following equation:

$$n_i = \sqrt{\sum_{k=1}^n H_i^2(k,1)} \quad (12)$$

where $H_{i(k,1)}$ denotes the coefficient of k^{th} row and 1st column of Hessenberg matrix and $k \leq n$.

Let $d_i = \text{mod}(n_i, 1)$, where d_i is the fractional part of n_i and $x_i = \text{floor}(d_i \times s)$, where x_i is the integer part of n_i , and $s \geq 10$.

Figure 1. Watermark embedding process.

7. The watermark bit is embedded into the Euclidean normalization n_i of the 1st column of Hessenberg matrix H_i . Watermark is embedded using the following rule:

(i) when $\text{mod}(x_i, 2) = 0$, the following equation is used:

$$n'_i = \begin{cases} n_i - \frac{x-1}{s} & \text{if } u(i) = 1 \\ n_i + \frac{x}{2} & \text{if } u(i) = 0 \end{cases} \quad (13)$$

(ii) when $\text{mod}(x_i, 2) = 1$, the following equation is used:

$$n'_i = \begin{cases} n_i + \frac{x}{2} & \text{if } u(i) = 1 \\ n_i - \frac{x-1}{s} & \text{if } u(i) = 0 \end{cases} \quad (14)$$

8. Finally, the largest coefficient denoted by $H_{i(k,1)_{largest}}$ of the 1st column of Hessenberg matrix is modified using the following equation:

$$H'_{i(k,1)_{largest}} = \left(H_{i(k,1)_{largest}} \right) \times \frac{n'_i}{n_i} \quad (15)$$

9. The modified largest coefficient $H'_{i(k,1)_{largest}}$ is re-inserted into H_i to obtain the modified Hessenberg matrix H'_i and inverse HD is applied for obtaining the modified matrix R'_i , which can be defined as:

$$R'_i = P_i \times H'_i \times P_i^T \quad (16)$$

10. N non-overlapping blocks including the modified block are recombined to obtain T'_i . Inverse PSHT is applied to the T'_i to obtain the modified matrix C'_i .

11. Each watermarked frame F'_i is obtained by reshaping each modified matrix C'_i .

12. Finally, watermarked signal Y' is obtained by concatenating all the watermarked frames.

The pseudo code of the watermark embedding procedure is presented in Algorithm 2.

Algorithm 2: Watermark Embedding

Variable Declaration:

Y : host audio signal

F : segmented non-overlapping frame

$C_i (i = 1, 2, \dots, M \times M)$: frame represented in dimensional matrix with size $m \times m$

$T_i (i = 1, 2, \dots, M \times M)$: transformed matrix

$B_j (j = 1, 2, \dots, N)$: non-overlapping bloc

$Z_j (j = 1, 2, \dots, N)$: sum of absolute mean of the j^{th} block

$R_i (i = 1, 2, \dots, M \times M)$: block with maximum sum of absolute mean

$H_i (i = 1, 2, \dots, M \times M)$: Hessenberg matrix

$n_i (i = 1, 2, \dots, M \times M)$: the 2nd order Euclidean normalization

$x_i (i = 1, 2, \dots, M \times M)$: quantization coefficient for embedding

Watermark Embedding Procedure:

for $i = 1: M \times M$ do

convert the i^{th} frame coefficients into two dimensional matrix C_i

apply PSHT on C_i to obtain T_i

for $j = 1: N$ do

subdividing into non-overlapping block B_j

calculate the sum of absolute mean Z_j of each block B_j using Equation (10)

end for

select block R_i with maximum sum of absolute mean Z_{max}

apply HD on matrix R_i using Equation (11)

calculate n_i using Equation (12)

calculate d_i and x_i

update n_i into n'_i using Equations (13) and (14)

modify the largest Hessenberg coefficient $H_{i(k,1)_{largest}}$ using Equation (15)

apply inverse HD on matrix R_i^* using Equation (16)

apply inverse PSHT on T_i^*

reshape C_i properly

reshape F_i^* properly.

end for

return watermarked audio Y^*

4.3. Watermark Extraction Process

The proposed watermark detection procedure is shown in Figure 2. The blind extraction of the watermark is described in the following steps:

Figure 2. Extraction process.

1. The attacked watermarked audio Y^* is firstly divided into $M \times M$ non-overlapping frames and each frame is converted into two-dimensional matrix C_i^* .
2. T_i^* is obtained by applying PSHT on each matrix C_i^* .
3. T_i^* is sub-divided into N non-overlapping blocks $B^* = \{B_j^*, 1 \leq j \leq N\}$ and Z_j^* is calculated. After that, R_i^* is selected.
4. HD is then performed on R_i^* to obtain the matrices P_i^* and H_i^* . n_i^* is calculated from H_i^* .
5. Then, d_i^* and x_i^* are calculated from n_i^* .
6. The encrypted watermark sequence is extracted using the following rule:

$$u^*(i) = \begin{cases} 1 & \text{if } \text{mod}(x_i^*, 2) = 1 \\ 0 & \text{if } \text{mod}(x_i^*, 2) = 0 \end{cases} \quad (17)$$

7. Chaotic decryption is performed using the secret key K in order to find the binary watermark sequence with the following equation:

$$r^*(i) = z(i) \oplus u^*(i) \quad (18)$$

8. Finally, the watermark sequence is obtained after rearranging the binary sequence $r^*(i)$ into a square matrix W^* with size $M \times M$.

The pseudo code of the watermark extraction procedure is presented in Algorithm 3.

Algorithm 3: Watermark Extraction**Variable Declaration:** Y^* : attacked watermarked audio signal F : attacked watermarked frame C_i^* ($i = 1, 2, \dots, M \times M$): watermarked frame represented in two dimensional matrix with size $m \times m$ T_i^* ($i = 1, 2, \dots, M \times M$): modified transformed matrix B_j^* ($j = 1, 2, \dots, N$): modified non-overlapping block Z_j^* ($j = 1, 2, \dots, N$): sum of absolute mean of modified the j^{th} block R_i^* ($i = 1, 2, \dots, M \times M$): modified block with maximum sum of absolute mean H_i^* ($i = 1, 2, \dots, M \times M$): modified Hessenberg matrix n_i^* ($i = 1, 2, \dots, M$): modified the 2nd order Euclidean normalization x_i^* ($i = 1, 2, \dots, M$): quantization coefficient for extraction**Watermark Extraction Procedure:**for $i = 1: M \times M$ doconvert the coefficients of the i^{th} frame into two dimensional matrix C_i^* apply PSHT on C_i^* to obtainfor $j = 1: N$ dosubdividing into non-overlapping block B_j calculate the sum of absolute mean Z_j of each block B_j

end for

select block R_i^* with maximum sum of absolute mean Z_{max}^* apply HD on matrix R_i^* calculate n_i^* calculate d_i^* and x_i^* calculate $u^*(i)$ using the Equation (17)calculate $r^*(i)$ using the Equation (18)reshape $r^*(i)$

end for

return watermark W^* **5. Experimental Results and Discussion**

In this section, the performance of our proposed algorithm has been evaluated and compared with some state-of-the-art methods. In this study, we used 20 audio files belong to four different audio groups as host audio signals, which are given below:

Group 1: 05 files containing pop music;

Group 2: 05 files containing classical music;

Group 3: 05 files containing jazz music;

Group 4: 05 files containing rock music;

All audio files are mono-channel 16 bit with a 44.1 kHz sampling rate and they contain 262,144 samples (duration 5.94 s). The selected size of the frame for each audio is 256 samples. Therefore, we have 1024 frames for each audio. A binary watermark image and the corresponding encrypted watermark image with size 32×32 are shown in Figure 3. Thus, one watermark bit is embedded in each frame. In this study, constant beta slant transform is used with parameters $\beta_4 = -2, \beta_8 = -2, \beta_{16} = -2$. Moreover, the selected value of $y(1), b, T$, and s are 1, 1, 0.5, and 10, respectively. These parameters are considered to obtain a good trade-off between the imperceptibility and robustness. HD is applied on matrix R_i with size 8×8 for better computation cost of space and time.

Figure 3. (a) Binary watermark image. (b) Encrypted watermark image.

5.1. Imperceptibility Analysis

Imperceptibility property of the proposed algorithm is assessed by using both subjective and objective analysis.

5.1.1. Subjective Analysis

For ensuring imperceptibility, perceptual quality of watermarked audio should be calculated. In this study, 10 participants were blindly given both the original and watermarked signals and were asked to differentiate these two signals based on a subjective difference grade (SDG) that ranged from 5.0 to 1.0 (imperceptible to very annoying) as given in Table 1. The average result of subjective grading is presented in Table 2. The result shows that the mean opinion score (MOS) of the proposed method lies between 4.9 to 5.0 for all watermarked audios, which ensures the imperceptibility of the watermarked audio.

Table 1. Subjective and objective difference grades.

SDG	ODG	Description	Quality
5	0	Imperceptible	Excellent
4	−1	Perceptible, but not annoying	Good
3	−2	Slightly annoying	Fair
2	−3	Annoying	Poor
1	−4	Very annoying	Bad

Subjective evaluation was also conducted by another technique known as the ABX method. The test was evaluated with the help of 10 subjects. At first, each subject listened to both the host signal (A) and the watermarked signal (B). Then, they were given another unknown signal (X) and were asked to find out the unknown one. Five trials were conducted by each subject. Table 2 presents the results of the correct detection, which varied between 48% to 54%, indicating the high imperceptibility of the proposed method.

Table 2. Subjective and objective evaluation for different watermarked sounds.

Audio Signal	MOS	Correct Detection	SNR	ODG
Pop	4.90	54%	43.81	−0.46
Classical	5.00	48%	47.75	−0.35
Jazz	5.00	48%	47.08	−0.37
Rock	4.90	54%	47.60	−0.38
Average	4.95	51%	46.56	−0.39

5.1.2. Objective Analysis

The objective assessment is generally measured by the SNR of the watermarked audio. According to the standard of Industrial Federation of the Phonographic Society (IFPI), the ideal SNR of watermarked audio should be more than 20 dB for satisfying the imperceptibility property [7]. The SNR of the proposed method for various audio is given in Table 2. We observed that SNRs of various audios are greater than 40 dB, which satisfy the international standard.

Moreover, objective assessment was also conducted using object difference grade (ODG), which is one of the output obtained from the perceptual evaluation of audio quality (PEAQ) measurement based on ITU-R BS.1387 (International Telecommunication Union-Radio-communication Sector) standard [7]. The ODG score lies between 0 to −4 (imperceptible to very annoying) given in Table 1. The objective quality of different audios using the proposed method are evaluated in terms of ODG and the results are shown in Table 2. It is observed that all ODGs of our proposed algorithm range from −0.39 to −0.46, indicating that the original and watermarked audios are perceptual similar. Table 3 shows a comparative analysis between the proposed and several recent methods [4,12] in terms of SNR and MOS. From this comparison, it was observed that our proposed method shows better result in terms of SNR and MOS. In other words, subjective and objective analysis proves that the proposed method provides better performance than the other methods in terms of imperceptibility.

Table 3. A comparative analysis between the proposed and various methods in terms of imperceptibility.

Reference	Method	SNR	MOS
[4]	Energy averaging	41.47	-
[5]	Localized and self-adaptive algorithm	31.40	3.7
[6]	LRS-FFT	44.81	-
[7]	DCT-SVD-ELO	33.47	4.88
[8]	SSA-PM	25.61	-
[9]	Multifunctional algorithm	23.33	-
[10]	DCT-SVD-LPT	37.20	4.85
[11]	SVD-QIM	19.39	-
[12]	FS-AE	33.6	-
Proposed	PSHT-HD	46.56	4.95

5.2. Robustness Analysis

The robustness of our proposed algorithm has been evaluated using (1) normalized correlation (NC) and (2) bit error rate (BER). Define if appropriate.

Normalized correlation (NC) compares the similarities between two images. It is calculated as follows:

$$NC(W, W^*) = \frac{\sum_{k=1}^M \sum_{l=1}^M w(k, l) \cdot w^*(k, l)}{\sqrt{\sum_{k=1}^M \sum_{l=1}^M w(k, l) \cdot w(k, l)} \sqrt{\sum_{k=1}^M \sum_{l=1}^M w^*(k, l) \cdot w^*(k, l)}} \quad (19)$$

where W and W^* denote the original watermark and extracted watermark, respectively, and k, l denote the matrix indices. The value of NC ranges from 1 to 0. The correlation of the two images is very high when the NC is closer to one. On the other hand, the correlation of the images is very low when the NC is closer to zero.

BER is generally used to calculate the bit error rate between the original and extracted watermark, which is given by:

$$BER(W, W^*) = \frac{\sum_{k=1}^M \sum_{l=1}^M w(k, l) \oplus w^*(k, l)}{M \times M} \quad (20)$$

For evaluating the robustness, various common signal processing attacks were applied on the watermarked audio signals which are given below:

1. Noise addition: Additive white Gaussian noise (AWGN) was added with a watermarked signal until the signal had an SNR of 20 dB.
2. Cropping: A number of 1000 samples of the watermarked audio were removed from different positions, and then, these samples were replaced with the watermarked audio signal attacked by additive white Gaussian noise.

3. Re-sampling: The watermarked signal with a sample rate of 44.1 kHz was sampled to 22.05 kHz and again resampled by a rate of 44.1 kHz.
4. Re-quantization: The watermarked audio was quantized from 16 bit to 8 bit.
5. Compression: The watermarked signal was compressed using MPEG-1 layer 3 compression (128 kbps).
6. Noise Reduction: Noise reduction was successfully done from the watermarked audio with the help of “Hiss removal” function.
7. Echo addition: Echo signal containing a delay time of 150 ms and decay rate of 35% was applied to the watermarked signal.
8. Distortion: The watermarked audio signal was distorted within a range of 0 dB to −10 dB.
9. Amplification: The watermarked audio was amplified (enlarged) by 1.25 times of its original amplitude.
10. Delay: A delay time of 150 ms was used and the volume of the delayed signal contains 3% of the original signal.
11. Invert: The watermarked audio signal was fully inverted to obtain the inverted form of the actual watermark signal.
12. Low-Pass Filter: A low-pass filter with a cut-off frequency of 15,000 Hz was applied to the watermarked audio.

Tables 4 and 5 show the robustness result of our proposed algorithm in terms of NC and BER, which are obtained from various attacked watermarked audio signals. We observed that the proposed method recovers the watermark successfully from the attacked watermark audio signals for noise reduction, invert, and echo addition, as the NC values are 1 and BER values are 0.

Moreover, the proposed method shows good NC and BER values for amplification, distortion, delay, re-sampling, re-quantization, cropping, and low-pass filtering attack. The NC of the proposed method for various attacks varies from 0.9459 to 1. Moreover, the BER of the proposed method varies from 0 to 6.54 for various attacks. In other words, the NCs of our proposed method are greater than 0.9459 and BERs of the proposed method are less than 7%. Figures 4–7 show the extracted watermark images for different audios against various attacks. From these figures, we observed that watermark is extracted without any errors in most of the cases, which proves the high robustness of the proposed method.

Table 4. NC of extracted watermark for watermarked signal against various attacks.

Attack	Pop	Classical	Jazz	Rock
No attack	1	1	1	1
Noise Addition	0.9986	0.9995	0.9911	1
Noise Reduction	1	1	1	1
Echo Addition	1	1	1	1
Cropping	0.9978	0.9977	0.9988	0.9982
Re-quantization	0.9968	1	0.9992	1
Compression (MP3)	0.9566	0.9459	0.9619	0.9643
Re-sampling	0.9836	1	0.9943	0.9893
Distortion	0.9766	1	0.9895	0.9992
Amplification	0.9944	1	0.9871	1
Delay	0.9944	0.9976	0.9895	1
Invert	1	1	1	1
Low-Pass Filtering	0.9649	0.9871	0.9822	0.9919

Table 5. BER (%) of extracted watermark for watermarked signal against various attacks

Attack	Pop	Classical	Jazz	Rock
No attack	0	0	0	0
Noise Addition	0.37	0.88	1.07	0
Noise Reduction	0	0	0	0
Echo Addition	0	0	0	0
Cropping	0.24	0.026	0.14	0.20
Re-quantization	0.39	0	0.09	0
Compression (MP3)	5.18	6.54	4.59	4.30
Re-sampling	1.67	0	0.68	0.88
Distortion	2.83	0	1.27	0.09
Amplification	0.68	0	1.56	0
Delay	0.49	0.29	1.27	0
Invert	0	0	0	0
Low-Pass Filtering	4.54	1.56	2.15	0.98

Figure 4. Extracted watermark against different attacks for pop audio signal: (a) no attack, (b) noise addition, (c) noise reduction, (d) echo addition, (e) cropping, (f) re-quantization, (g) compression (MP3), (h) re-sampling, (i) distortion, (j) amplification, (k) delay, (l) invert, (m) low-pass filter. **Figure 5.** Extracted watermark against different attacks for classical audio signal: (a) no attack, (b) noise addition, (c) noise reduction, (d) echo addition, (e) cropping, (f) re-quantization, (g) compression (MP3), (h) re-sampling, (i) distortion, (j) amplification, (k) delay, (l) invert, (m) low-pass filter.

Figure 6. Extracted watermark against different attacks for jazz audio signal (a) no attack, (b) noise addition, (c) noise reduction, (d) echo addition, (e) cropping, (f) re-quantization, (g) compression (MP3), (h) re-sampling, (i) distortion, (j) amplification, (k) delay, (l) invert, (m) low-pass filter.

Figure 7. Extracted watermark against different attacks for rock audio signal (a) no attack, (b) noise addition, (c) noise reduction, (d) echo addition, (e) cropping, (f) re-quantization, (g) compression (MP3), (h) re-sampling, (i) distortion, (j) amplification, (k) delay, (l) invert, (m) low-pass filter.

Table 6 illustrates a comparative analysis between the proposed and some recent methods [4–12] in terms of noise addition, resampling, re-quantization, and MP3 compression. From this table, we observed that our proposed method shows less BER than the other recent methods for noise addition. Moreover, it shows better result than that of the methods presented in [4,6–8,10,11] for the re-sampling attack. For the re-quantization attack, it shows better result than that of the methods proposed in [5–8] and for MP3 compression, it shows better result than that of the methods suggested in [5–9,11,12]. From these results, we can conclude that our proposed method provides lower BER values against some common attacks compared with some recent state-of-the-art methods. Overall, our proposed method shows better performance than the recent state-of-the-art methods in terms of imperceptibility and robustness. This is because the watermark bits were inserted into the largest value of the 1st column of the Hessenberg matrix of PSHT coefficients of each frame using a quantization function.

Table 6. General comparison of several recent methods with proposed algorithm in terms of BER (%).

Reference	Method	Noise Addition	Resampling	Re-Quantization	MP3 Compression
Proposed	PSHT-HD	0.58(20 dB)	0.81(22.05 kHz)	0.12 (8 Bits/Sample)	5.15(128 kbps)
[4]	Energy averaging	-	8.0(22.05 kHz)	-	5.0(128 kbps)
[5]	Localized and self-adaptive algorithm	6.03(30 dB)	0(22.05 kHz)	0.14(8 bits/sample)	6.20(64 kbps)
[6]	LRS-FFT	5.17(-)	6.56(22.05 kHz)	4.94(8 bits/sample)	6.88(128 kbps)
[7]	DCT-SVD-ELO	0.91(-)	0.88(22.05 kHz)	0.23(8 bits/sample)	6.13 (32 kbps)
[8]	SSA-PM	2.50(36 dB)	6.06(22.05 kHz)	8.83(16 bits/sample)	9.44(128 kbps)
[9]	Multifunctional algorithm	4.22(-)	0(22.05 kHz)	-	7.48(32 kbps)
[10]	DCT-SVD-LPT	0.83(-)	1.56(22.05 kHz)	0(8 bits/sample)	3.91(128 kbps)
[11]	SVD-QIM	10.25(30 dB)	4.88(16 kHz)	-	17.76(128 kbps)
[12]	FS-AE	7.23(20 dB)	-	-	6.04(48 kbps)

5.3. Data Payload

Data payload defines the number of bits that can be embedded into the original signal over a unit of time. It is measured by bits per second. The data payload P is defined as follows:

$$P = \frac{B}{T} (bps) \quad (21)$$

where T indicates the time duration of the original audio signal and B indicates the number of watermark bits to be embedded into the host signal. The standard value for data payload is more than 20 bps [7]. The data payload value of our proposed scheme is 172.39 bps, which is much higher than the standard value.

5.4. Security Analysis

To enhance the security, the proposed scheme uses chaotic encryption. First, we encrypted the watermark using logistic mapping where a key K is used for both encryption and decryption. Second, there is another parameter β , which is used in the PSHT process. Different values of β shows different experimental results. Last, a quantization coefficient x was used for both embedding and blind extraction. Therefore, it is not possible to detect the embedded watermark without these three parameters.

5.5. Computation Time Analysis

The computation time of our proposed method including both the embedding and extraction processes is calculated and compared with that of the methods presented in [5,6,8], which is given in Table 7. We observed that the computation time for embedding process of our proposed method is 2.03 s, which is much lower than that of the methods given in [5,8], whereas it is slightly higher than that of the method reported in [6]. On the other hand, the computation time for detection process of our proposed method is 0.75 s, which is much lower than that of the methods given in [6,8]. From this point of view, it can be concluded that the proposed method has lower computational cost compared with other methods.

Table 7. Comparison of several recent methods with proposed algorithm in terms of computation time.

Reference	Method	Embedding Time(s)	Extraction Time(s)
[5]	Localized and self-adaptive algorithm	2.77–3.42	-
[6]	LRS-FFT	1.46	0.89
[8]	SSA-PM	258	1200
Proposed	PSHT-HD	2.03	0.75

6. Conclusions

In this paper, we proposed a blind symmetric audio watermarking algorithm based on two well-known transformation and decomposition techniques, namely PSHT and HD, which are used in audio watermarking for the first time. Watermark is embedded into the largest value of the 1st column of the Hessenberg matrix of PSHT coefficients of each frame using a new quantization function. By simulation, it is demonstrated that the proposed algorithm is highly robust against numerous attacks such as noise addition, noise reduction, echo addition, cropping, re-quantization, MP3 compression, re-sampling, distortion, amplification, delay, invert, and low-pass filter. In addition, the proposed algorithm is computationally faster and it has high data payload. Moreover, the audio quality tests ensure the high imperceptibility of the watermarked audios. Furthermore, comparative analysis substantiates its superiority among other state-of-the-art methods. These results verified the validity of our proposed algorithm for audio copyright protection. In the future, the proposed algorithm

will be compared with several recent state-of-the-art methods using the same dataset in terms of imperceptibility, robustness, and computation time.

Author Contributions: All authors contributed equally to the conception of the idea, the design of experiments, the analysis and interpretation of results, and the writing and improvement of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xiang, Y.; Hua, G.; Yan, B. *Digital Audio Watermarking: Fundamentals, Techniques and Challenges*; Springer: Singapore, 2017.
2. Cvejic, N. *Digital Audio Watermarking Techniques and Technologies: Applications and Benchmarks*; IGI Global: Hershey, PA, USA, 2007.
3. Pandey, M.K.; Parmar, G.; Gupta, R. Audio watermarking by spreading echo in time domain using pseudo noise gray sequence. In Proceedings of the IEEE International Conference on Industrial Instrumentation and Control (ICIC), Pune, India, 28–30 May 2015; pp. 740–743.
4. Kaur, A.; Dutta, M.K.; Soni, K.M.; Taneja, N. Localized & self-adaptive audio watermarking algorithm in the wavelet domain. *J. Inf. Secur. Appl.* **2017**, *33*, 1–15.
5. Tsai, S.E.; Yang, S.M. An effective watermarking method based on energy averaging in audio signals. *Math. Probl. Eng.* **2018**, *2018*, 6420314. [\[CrossRef\]](#)
6. Pourhashemi, S.M.; Mosleh, M.; Erfani, Y. Audio watermarking based on synergy between Lucas regular sequence and Fast Fourier Transform. *Multimed. Tools Appl.* **2019**, *78*, 22883–22908. [\[CrossRef\]](#)
7. Dhar, P.K.; Shimamura, T. Blind audio watermarking in transform domain based on singular value decomposition and exponential-log operations. *Radioengineering* **2017**, *26*, 552–561. [\[CrossRef\]](#)
8. Karnjana, J.; Unoki, M.; Aimmanee, P.; Wutiwanchai, C. Audio watermarking scheme based on singular spectrum analysis and psychoacoustic model with self-synchronization. *J. Electr. Comput. Eng.* **2016**, *2016*, 5067313. [\[CrossRef\]](#)
9. Liu, H.; Liu, X.; Shi, B.; Chen, T.; Wang, J. Multifunctional audio watermarking algorithm based on Chaotic Scrambling. *J. Comput. Methods Sci. Eng.* **2017**, *17*, 443–454. [\[CrossRef\]](#)
10. Dhar, P.K.; Shimamura, T. Blind SVD-based audio watermarking using entropy and log-polar transformation. *J. Inf. Secur. Appl.* **2015**, *20*, 74–83. [\[CrossRef\]](#)
11. Hwang, M.J.; Lee, J.; Lee, M.; Kang, H.G. SVD-based adaptive QIM watermarking on stereo audio signals. *IEEE Trans. Multimed.* **2018**, *20*, 45–54. [\[CrossRef\]](#)
12. Luo, Y.; Peng, D.; Sang, Y.; Xiang, Y. Dual-domain audio watermarking algorithm based on flexible segmentation and adaptive embedding. *IEEE Access* **2019**, *7*, 10533–10545. [\[CrossRef\]](#)
13. Hu, H.T.; Lee, T.T. High-performance self-synchronous blind audio watermarking in a unified FFT framework. *IEEE Access* **2019**, *7*, 19063–19076. [\[CrossRef\]](#)
14. Choudhary, S.; Nath, K.; Panda, J. Double layered audio zero-watermarking using DWT and DSSS. In Proceedings of the International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 6–8 April 2017; pp. 0419–0423.
15. Irawati, I.D.; Budiman, G.; Ramdhani, F. QR-based watermarking in audio subband using DCT. In Proceedings of the International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), Bandung, Indonesia, 5–7 December 2018; pp. 136–141.
16. Gupta, A.K.; Agarwal, A.; Singh, A.; Vimal, D.; Kumar, D. Blind audio watermarking using adaptive quantization and Lifting wavelet transform. In Proceedings of the IEEE 5th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 22–23 February 2018; pp. 556–559.
17. Weina, W. Digital audio blind watermarking algorithm based on audio characteristic and scrambling encryption. In Proceedings of the IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 25–26 March 2017; pp. 1195–1199.
18. Tang, X.; Ma, Z.; Niu, X.; Yang, Y. Robust audio watermarking algorithm based on empirical mode decomposition. *Chin. J. Electron.* **2016**, *25*, 1005–1010. [\[CrossRef\]](#)

19. Safitri, I.; Ginanjar, R.R.; Rizal, A. Adaptive multilevel wavelet BCH code method in the audio watermarking system. In Proceedings of the IEEE International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC), Yogyakarta, Indonesia, 26–28 September 2017; pp. 55–59.
20. Sulistyawan, V.N.; Budiman, G.; Safitri, I. Histogram-based audio watermarking with synchronization in stationary audio subband. In Proceedings of the IEEE International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), Bandung, Indonesia, 5–7 December 2018; pp. 195–201.
21. Sakai, H.; Iwaki, M. Audio watermarking method based on phase-shifting having robustness against band-pass filtering attacks. In Proceedings of the IEEE 7th Global Conference on Consumer Electronics (GCCE), Nara, Japan, 9–12 October 2018; pp. 343–346.
22. Agaian, S.; Tourshan, K.; Noonan, J.P. Parametric Slant-Hadamard transforms with applications. *IEEE Signal Process. Lett.* **2002**, *9*, 375–377. [[CrossRef](#)]
23. Seddik, H.; Sayadi, M.; Fnaiech, F.; Cheriet, M. Image watermarking based on the Hessenberg transform. *Int. J. Image Graph.* **2009**, *9*, 411–433. [[CrossRef](#)]

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).