symmetry

MDPI

Article

Adaptive and Blind Audio Watermarking Algorithm Based on Chaotic Encryption in Hybrid Domain

Qiuling Wu [1,2,*] and Meng Wu [1]

[1] College of Telecommunications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; wum@njupt.edu.cn
[2] College of Zijin, Nanjing University of Science and Technology, Nanjing 210046, China
* Correspondence: redpond2000@163.com; Tel.: +86-25-5268-9065

Received: 27 June 2018; Accepted: 13 July 2018; Published: 14 July 2018

Abstract: An adaptive and blind audio watermarking algorithm is proposed based on chaotic encryption in discrete cosine transform (DCT) and discrete wavelet transform (DWT) hybrid domain. Since human ears are not sensitive to small changes in the high-frequency components of the audio media, the encrypted watermark can be embedded into the audio signal according to the special embedding rules. The embedding depth of each audio segment is controlled by the overall average amplitude to effectively improve the robustness and imperceptibility. The watermark is encrypted by a chaotic sequence to improve the security of watermark, so only users who hold the correct key can accurately extract the watermark without the original audio signal. Experimental results show that the proposed algorithm has larger capacity, higher imperceptibility, better security, and stronger robustness when combating against signal-processing attacks than the involved audio watermarking algorithms in recent years.

Keywords: audio digital watermarking; chaotic encryption; adaptive embedding depth; blind extraction

1. Introduction

With the rapid development of the Internet and multimedia technologies, it is convenient to transmit digital information all over the world quickly. However, the information security problem has become a global issue to be solved [1–3]. A digital watermarking algorithm is an effective method to protect media content in the fields of copyright protection, fingerprint identification, broadcast monitoring, medical security, data authentication and so on. In recent years, it has become a hot topic in the field of communication and information security [4,5].

A digital watermarking algorithm can be applied to different multimedia carriers such as audio, image [6–8], data and video [9]. Because the audio media contains less redundant information, it is difficult to develop an audio watermarking algorithm. With the widespread use of audio media on the network, people begin to focus on the research of an audio watermarking algorithm. Over the past decades, many audio watermarking algorithms have appeared in different domains, such as the time and transform domains. In general, the time domain watermarking algorithm is easy to implement, but less robust in combating various digital signal processing attacks [2,4], such as the algorithms in literature [10,11]. Compared with the time-domain algorithms, the algorithms in the transform-domain, such as the discrete Fourier transform (DFT) [12–14], discrete cosine transform (DCT) [15,16], discrete wavelet transform (DWT) [17–21] and singular value decomposition (SVD) [22,23] and so on, are more robust because they explore human auditory properties and the features of audio signal. Natgunanathan [13] presented a blind watermarking algorithm by DFT for stereo signals. Pal [14] used audio signal as a carrier to transmit the data that needs to be

kept secret in the DFT domain. DCT has the characteristics of compressing the signal energy to its low frequency coefficient, which makes it widely used in data compression. Hu [15] designed an audio watermarking by modifying the vectors in the DCT domain. Natgunanathan [16] designed an audio watermarking method in a multilayer framework to improve capacity in the DCT domain. DWT decomposes the audio signal into different frequent bands both in the time and frequency domains, so the watermark algorithms designed by DWT are usually very robust. Qian [17] proposed an audio watermarking algorithm to solve the problem of content authentication and recovery in the encrypted domain. Chen [18] utilized DWT to design an adaptive method with poor robustness for resampling and low-pass filtering. Wu [19] proposed an audio watermarking algorithm by adjusting the high-frequency wavelet coefficients of each audio segment in the DWT domain. Hu [20] proposed a blind watermarking scheme to embed a binary watermark into a low-frequency approximation sub-band based on lifting wavelet transform (LWT). Li [21] used the norm ratio of approximate coefficients to design an audio watermarking scheme to balance the performance of the algorithm in the DWT domain.

All of these algorithms are designed in a single-transform domain, and there are many schemes designed in hybrid domains in recent years. Liu [24] proposed a scheme for audio signal tamper recovery and location tampering based on DWT and DCT. Hu [25] proposed an audio watermarking algorithm to achieve invisible data hiding based on DWPT, SVD and quantization index modulation (QIM) hybrid domains. In general, the watermarking algorithms designed in the hybrid domain have better performance than those designed in a single-transform domain according to the experimental results of the above literature.

An audio watermarking algorithm can be evaluated by four indexes which are robustness, imperceptibility, capacity and security [4,16]. Imperceptibility means that listeners cannot distinguish the difference between the original audio and the watermarked audio. Robustness indicates that the algorithm can extract the watermark accurately when the watermarked audio has suffered from external attacks. Capacity means the capability of the watermarked audio to accommodate the necessary information. Security refers to the fact that the user cannot obtain the watermark without the correct key. The watermark algorithm used for copyright protection must have good imperceptibility so as to prevent the audio media from losing its usage value after being embedded in watermarks. In addition, the algorithm must have strong robustness, because the audio media may suffer various attacks in the process of transmission, such as Gaussian noise, format conversion, resampling and other attacks which may cause the watermark to be lost. Security can be achieved through encryption in the process of watermark pretreatment. Capacity is another important index. Under the premise of ensuring the imperceptibility and robustness, the larger the capacity, the more useful information the watermark contains. Most audio watermarking schemes have disadvantages such as poor robustness, low capacity and weak audio quality. Therefore, the audio watermarking algorithm needs further research to improve its overall performance.

The purpose of this study is to combine all the useful features of DWT and DCT to design a practical audio watermarking algorithm in order to improve robustness, imperceptibility, security and capacity. The original audio is divided into multiple audio segments, and then each segment is decomposed by DWT to get the detail coefficients which are divided into two packets for carrying a 1 bit watermark. It is helpful to improve the imperceptibility and robustness of the algorithm by taking advantage of the overall average amplitude of each audio segment to adjust the embedding depth. The experimental results show the excellent performance of this algorithm, including large capacity, high imperceptibility, and strong robustness which can withstand 10 common attacks. Since the watermark has been encrypted before it is embedded, only the users who hold the secret key can obtain the watermark accurately, so the algorithm has excellent security performance.

This paper is organized as follows: Section 2 describes the proposed watermarking algorithm which consists of four subjects, including the pretreatment to the watermark picture, the principle of watermark embedding, the principle of watermark extraction, and the design of the adaptive

embedding depth. The detailed implementation steps of this proposed algorithm are described in Section 3. The experimental results are analyzed and compared with that of some relevant algorithms in recent years in Section 4. The conclusions are presented in Section 5.

2. Watermarking Algorithm in Hybrid Domain

Digital watermarking technology embeds an invisible watermark into digital media to achieve the purpose of protecting the copyright of this digital media according to the principle of data hiding. A general block diagram is shown in Figure 1. Watermarks are concealed into the audio medium which needs copyright protection to obtain the watermarked audio medium by the embedding algorithm. In the process of being used or transmitted, the watermarked audio media will suffer various attacks, such as Gaussian noise, format compression, and re-sampling. Watermarks can be extracted from the audio medium through the extraction algorithm. Embedding algorithm and extraction algorithms are the core of this audio watermarking algorithm. Encryption is usually carried out before the watermark is embedded in order to improve security.

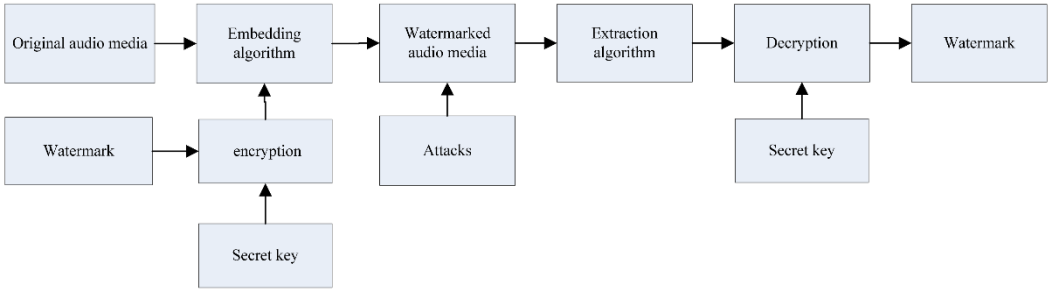

Figure 1. General block diagram of audio watermarking algorithm.

2.1. The Pretreatment to the Watermark Picture

It is assumed that the binary watermark to be embedded is a binary picture with two dimensions $L_1 \times L_2$, and it can be defined as:

$$W_1 = \{w_1(u,v), 1 \le u \le L_1, 1 \le v \le L_2\} \tag{1}$$

where $w_1(u,v) \in \{0, 1\}$ is the pixel value of this binary watermark. One dimensional binary stream is obtained after dimensionality reduction of the watermark picture.

$$W_2 = \{w_2(q), 1 \le q \le L\} \tag{2}$$

where $L = L_1 \times L_2$, and $w_2(q) \in \{0,1\}$. Using the logistic system shown in formula (3) to generate chaotic binary sequence $c(q)$:

$$x_{q+1} = \alpha x_q(1 - x_q) \tag{3}$$

$$c(q) = \begin{cases} 1 & x_q \ge \delta \\ 0 & x_q < \delta \end{cases} \tag{4}$$

where $0 < x_q < 1$, $1 \le q \le L$, and δ is the threshold. When $3.5699456 \le \alpha \le 4$, the system is located in a chaotic state.

Chaotic encryption is applied to W_2 to increase the security of this algorithm. The encrypted watermark is obtained from $w_2(q)$ and $c(q)$ by an XOR operation according to formula (5).

$$w_3(q) = w_2(q) \oplus c(q) \tag{5}$$

$$W_3 = \{w_3(q), 1 \le q \le L\} \tag{6}$$

Modulate $w_3(q)$ into a bipolar string according to Formula (7):

$$w(q) = \begin{cases} 1 & w_3(q) = 1 \\ -1 & w_3(q) = 0 \end{cases} \tag{7}$$

Using $Ch(x_1, \alpha, \delta)$ as a secret key to extract the watermark, only users who hold this key can extract the watermark correctly.

2.2. Principle of Watermark Embedding

The frequency band of audio signals that can be caught by human ears is mainly within the range of 300~3400 Hz. The low-frequency signal below 300 Hz and the high-frequency signal beyond 3400 Hz can be barely caught by human ears. The watermark information may be concealed into the audio signal because of the insensitivity of the human auditory system to the small changes of the high-frequency component.

It is assumed that A represents the original audio signal, and it can be defined as:

$$A = \{a(k), 1 \le k \le K\} \tag{8}$$

where $a(k)$ is the value of the kth sample point, and K is the length of this audio signal. Divide A into M audio segments A_l ($1 \le l \le M$), and each segment contains N sampling points. The r-level DWT is performed on A_l to get the r th level wavlet coefficients $De(r, n)$ ($1 \le n \le N/2^r$). Divide $De(r, n)$ into a former packet and a latter packet, namely $De_1(r, j)$ $De_2(r, j)$, and the two packets are shown in Formulas (9) and (10) in accordance with literature [4].

$$D_{e1}(r, j) = D_e(r, j),\ j = 1, 2, \ldots, N/2^{r+1} \tag{9}$$

$$D_{e2}(r, j) = D_e(r, \frac{N}{2^{r+1}} + j),\ j = 1, 2, \ldots, N/2^{r+1} \tag{10}$$

Two groups of DCT coefficients $C_1(r, j)$ and $C_2(r, j)$ are obtained from $De_1(r, j)$ and $De_2(r, j)$ by DCT, and then connect them to form $C(r, n)$ with $N/2^r$. Formulas (11)–(13) can be used to calculate the average amplitudes of $|C(r, n)|$, $|C_1(r, j)|$ and $|C_2(r, j)|$ to get M_l, M_{c1} and M_{c2}.

$$M_l = \frac{2^r}{N} \sum_{n=1}^{N/2^r} |C(r, n)| \tag{11}$$

$$M_{c1} = \frac{2^{r+1}}{N} \sum_{j=1}^{N/2^{r+1}} |C_1(r, j)| \tag{12}$$

$$M_{c2} = \frac{2^{r+1}}{N} \sum_{j=1}^{N/2^{r+1}} |C_2(r, j)| \tag{13}$$

In order to embed the watermark, $C_1(r, j)$ and $C_2(r, j)$ can be modified according to the following embedding rules:

$$C_1'(r, j) = C_1(r, j) \times \frac{(1 + \lambda w(q)) M_l}{M_{c1}} \tag{14}$$

$$C_2'(r, j) = C_2(r, j) \times \frac{(1 - \lambda w(q)) M_l}{M_{c2}} \tag{15}$$

where λ is the embedding depth and its span is within the interval of [0,1]. $C_1'(r, j)$ and $C_2'(r, j)$ are the watermarked DCT coefficients. Perform the inverse discrete cosine transform (IDCT) on $C_1'(r, j)$ and $C_2'(r, j)$ to get the watermarked coefficients $De_1'(r, j)$ and $De_2'(r, j)$, and then perform the inverse discrete wavelet transform (IDWT) to reconstruct the watermarked audio segment A_l'.

2.3. Principle of Watermark Extracting

The watermark extraction process is an inverse process of the watermark embedding process. Firstly, divide A' into M audio segments A'_l with N sample points, and then perform r-level DWT on A'_l to get $De'(r,n)$. Secondly, separate $De'(r,n)$ into two packets, and then perform DCT to get $C'_1(r,j)$ and $C'_2(r,j)$. Finally, the average amplitudes are calculated according to Formulas (16) and (17), and the watermark hidden in the audio signal is extracted according to the extraction formula. According to the embedding rules as shown in Formulas (14) and (15), if $w(q) = 1$, the average amplitudes of the two packets are:

$$M'_{c1} = \frac{2^{r+1}}{N} \sum_{j=1}^{N/2^{r+1}} |C'_1(r,j)| = (1+\lambda)M_l \tag{16}$$

$$M'_{c2} = \frac{2^{r+1}}{N} \sum_{j=1}^{N/2^{r+1}} |C'_2(r,j)| = (1-\lambda)M_l \tag{17}$$

The average amplitude of $C'_l(r,n)$ is:

$$M'_l = \frac{1}{2}[(1+\lambda)M_l + (1-\lambda)M_l] = M_l \tag{18}$$

It can be seen that the overall average amplitude of each audio segment does not change after the DCT coefficients are modified. Replace M_l with M'_l in Formulas (16) and (17) to get the average amplitudes of the modified packets.

$$M'_{c1} = (1+\lambda)M'_l \tag{19}$$

$$M'_{c2} = (1-\lambda)M'_l \tag{20}$$

According to Formulas (19) and (20), when $\lambda \geq 0$, $M'_{c1} \geq M'_{c2}$. If $w(q) = -1$, the average amplitudes of $C'_1(r,j)$ and $C'_2(r,j)$ are:

$$M'_{c1} = (1-\lambda)M'_l \tag{21}$$

$$M'_{c2} = (1+\lambda)M'_l \tag{22}$$

According to Formulas (21) and (22), when $\lambda > 0$, $M'_{c_1} < M'_{c2}$. It can be known from the above analysis that the watermark can be extracted from A'_l according to Formula (23).

$$w'_2(q) = \begin{cases} 1, & \text{if } M'_{c1} \geq M'_{c2} \\ 0, & \text{if } M'_{c1} < M'_{c2} \end{cases}, \; q = 1, 2, \ldots, L \tag{23}$$

Using $Ch(x_1, \alpha, \delta)$ as the secret key to generate the binary chaotic sequence $c(q)$, then obtain the decrypted picture $w'(q)$ according to the Formula (24).

$$w'(q) = w'_2(q) \oplus c(q) \tag{24}$$

2.4. The Design of the Adaptive Embedding Depth

The signal-to-noise ratio (SNR) can be used as a performance index to evaluate the quality of the watermarked audio for evaluating the performance of the watermarking algorithm, and it can be expressed as:

$$\mathrm{SNR}(A, A') = 10\lg\left\{ \frac{\sum_{k=1}^{K} A^2}{\sum_{k=1}^{K} (A' - A)^2} \right\} \tag{25}$$

where A and $A\prime$ denote the original and watermarked audio signals, respectively. Bit error rate (BER) can be used to evaluate the robustness of the proposed algorithm when resisting various attacks, and it can be expressed as:

$$\mathrm{BER}(w, w') = \frac{\sum_{q=1}^{L} w(q) \oplus w(q)'}{L} \times 100\% \tag{26}$$

where $\oplus$ stands for the exclusive-OR operator, and $w(q)$ and $w(q)'$ are the original watermark and the extracted watermark respectively, and L is the watermark length. The similarity between the original picture and the extracted picture can be tested by the normalized correlation (NC) coefficient shown as Formula (27).

$$NC(w, w') = \frac{\sum_{q=1}^{L} w(q) \times w'(q)}{\sqrt{\sum_{q=1}^{L} w(q)^2 \sum_{q=1}^{L} w'(q)^2}} \tag{27}$$

According to the watermark embedding principle in Section 2.2, this algorithm conceals the watermark by modifying $C_1(r, j)$ and $C_2(r, j)$ in which a small variation means that the algorithm makes minor modifications to the audio signal, which indicates that the imperceptibility of this algorithm is good.

When $w(q) = 1$, since the values of M_l, M_{c1} and M_{c2} are almost equal in the same audio segment, the variations of $C_1(r, j)$ and $C_2(r, j)$ can be expressed as:

$$\Delta C_1(r, j) = C_1'(r, j) - C_1(r, j), = C_1(r, j)[\frac{(1 + \lambda)M_l}{M_{c1}} - 1] \approx \lambda C_1(r, j) \tag{28}$$

$$\Delta C_2(r, j) = C_2'(r, j) - C_2(r, j) = C_2(r, j)[\frac{(1 - \lambda)M_l}{M_{c2}} - 1] \approx -\lambda C_2(r, j) \tag{29}$$

It can be seen that those two variations are determined by the embedding depth and their own amplitudes. The smaller λ is, the smaller $\Delta C_1(r, j)$ and $\Delta C_2(r, j)$ are, and the better the imperceptibility is. When $w(q) = -1$, the analysis process and results are similar to those mentioned above.

The extraction principle in Section 2.3 shows that the extraction process is achieved by comparing the average amplitudes of $C_1'(r, j)$ and $C_2'(r, j)$. The larger the average amplitude difference between $C_1'(r, j)$ and $C_2'(r, j)$ is, the smaller BER is, and the better the robustness of the algorithm. When $w(q) = 1$, the average amplitude difference between $C_1'(r, j)$ and $C_2'(r, j)$ is:

$$\Delta M' = \left|M_1' - M_2'\right| = |(1 + \lambda)M_l - (1 - \lambda)M_l| = 2\lambda M_l \tag{30}$$

It can be seen that $\Delta M'$ is determined by two factors: the embedding depth λ and the overall average amplitude M_l. The bigger λ is, then the larger $\Delta M'$, and the better the robustness. When $w(q) = -1$, the analysis process and results are similar to those mentioned above.

In conclusion, λ has an important influence on SNR and BER of this algorithm. In practical applications, when M_l is larger, a smaller λ can be chosen to obtain better imperceptibility, On the other hand, a larger λ can be chosen to obtain stronger robustness when M_l is smaller, which can balance SNR and BER of this algorithm.

The influence of λ on SNR and BER can be tested by the following experiments. Divide the original audio into M audio segments with 256 sample points, and perform 4-level DWT on each audio segment. Calculate M_l of each audio segment according to the embedding principle in Section 2.2. Five groups of audio segments are obtained according to the value of M_l from large to small, and each group contains $M/5$ audio segments which are named from Group 5 to Group 1, respectively. Finally, $M/5$ bit binary watermarks are embedded into each group. The experimental results about SNR and BER (under the Gaussian noise with 20 dB) are shown in Figure 2.

As shown in Figure 2, under the same embedding depth, the SNR of Group 5 with the larger amplitude is lower than the other groups, but the BER is superior to the other groups, which indicates that imperceptibility is not good but the accuracy of the watermark extraction is very high when the watermark is embedded in this audio segment with larger average amplitude. Therefore, a smaller embedding depth can be selected in Group 5 to enhance the imperceptibility of the algorithm.

The SNR of the Group 1 with the smaller amplitude is higher than the other groups, but the BER is inferior to the other groups, which indicates that when the watermark is embedded in this audio segment with smaller average amplitude, the imperceptibility is good but the accuracy of the watermark extraction is not high. Therefore, a larger embedding depth can be selected in Group 1 to improve the robustness.

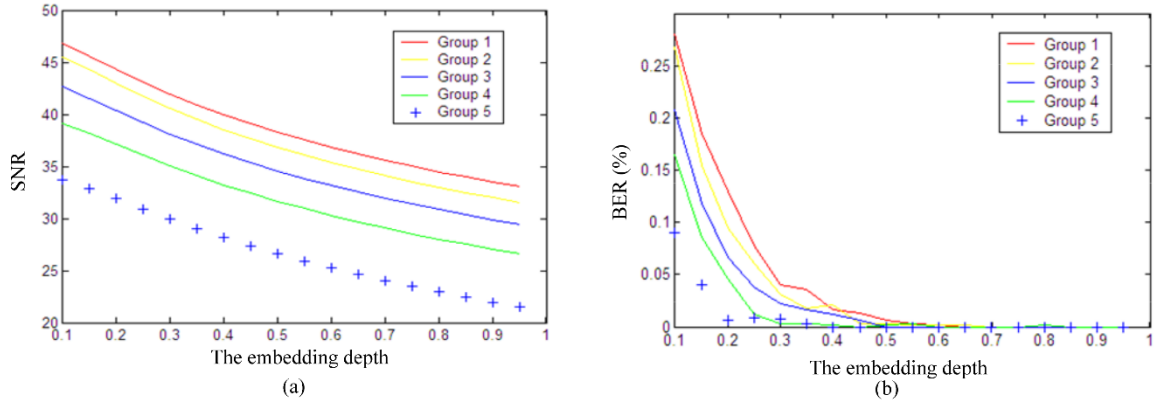

Figure 2. Performance comparison at different embedding depth of five groups: (**a**) the signal-to-noise ratio (SNR) comparison; (**b**) the BER comparison.

Those experimental results in Figure 2 are consistent with the above theoretical analysis results. The embedding depth can be adjusted according to the overall average amplitude of each audio segment so as to balance among imperceptibility and robustness. The embedded depth λ_l can be set up by the following formula:

$$\lambda_l = a + (b - a)\frac{Max - M_l}{Max - Min} \tag{31}$$

where $a, b \in (0, 1)$, $a < b$, $1 \leq l \leq M$, Max and Min are the maximum and minimum of M_l.

3. Detailed Implementation Steps

The embedding and extracting principle of this proposed audio watermarking technology are described in Section 2. The framework diagram of this algorithm is shown in Figure 3.

Two sets of data packets are obtained after the original audio media are carried out by DWT and DCT, and then a 1 bit encrypted watermark is hidden into the audio medium according to Formulas (14) and (15). When it is necessary to extract the hidden watermark in the audio media, two sets of data packets are obtained by executing DWT and DCT in the audio media, and the watermarks can be extracted by comparing the average magnitudes of this two packets according to Formula (23). The following steps are given for the implementation of this algorithm, including embedding watermark and extracting watermark.

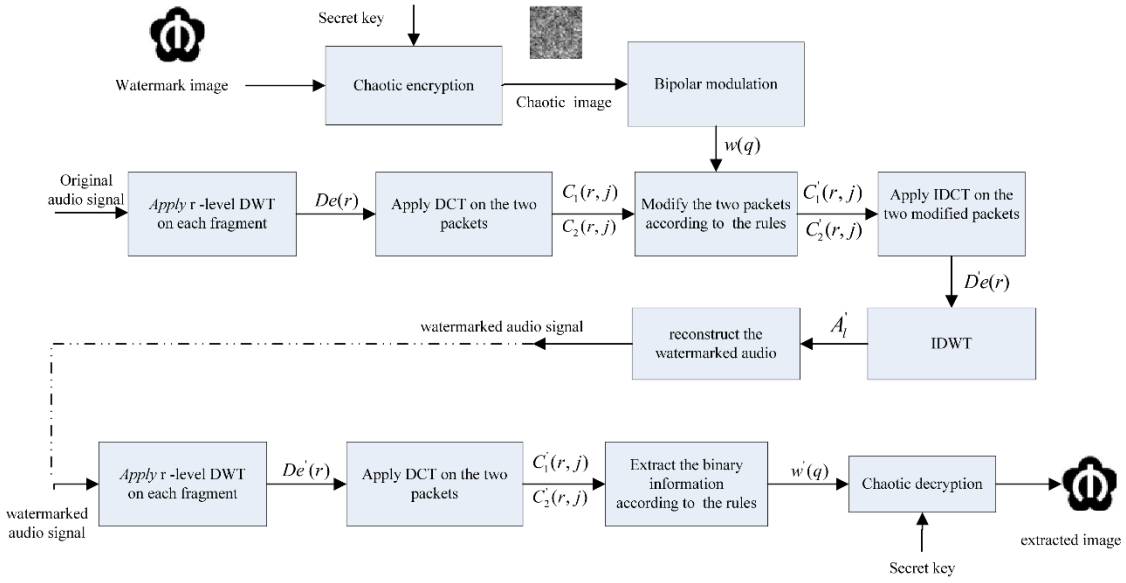

Figure 3. Framework diagram of this algorithm.

3.1. Implementation Steps for Embedding Watermark

The embedding procedure mainly includes the following steps:

Step 1: Convert the watermark picture into binary stream with the length of L, and then generate the binary chaotic sequence $c(q)$ according to Formulas (3) and (4), a bipolar string $w(q)$ is generated according to Formulas (5)–(7) ultimately.

Step 2: Add a group of "1111 1111" at the beginning of the bipolar string as the start sign and add a group of "-1-1-1-1-1-1-1-1" at the end of the bipolar string as the end sign.

Step 3: Divide A into M audio segments A_l with N sample points. $M \geq L + 16$.

Step 4: Perform the r-level DWT on A_l to get $De(r, n)$.

Step 5: Separate $De(r, n)$ into $De_1(r, j)$ and $De_2(r, j)$, and then implemented DCT on them to obtain $C_1(r, j)$ and $C_2(r, j)$.

Step 6: Calculate M_l, M_{c1} and M_{c2} according to Formulas (11)–(13).

Step 7: Repeat Step 4 to Step 6. Calculate the average amplitudes of all audio segments to obtain Max and Min.

Step 8: Calculate the adaptive embedding depth of each audio segment according to Formula (31). Embed a 1 bit watermark into each audio segment according to the embedding rules in Formulas (14) and (15).

Step 9: Perform IDCT on $C_1'(r, j)$ and $C_2'(r, j)$ respectively to get $De'(r, n)$.

Step 10: Perform IDWT on $De'(r, n)$ to reconstruct A_l'.

Step 11: Repeat Step 8 to Step 10 until the end of the embedding process.

Step 12: Recombine A_l' to obtain the whole watermarked audio A'.

3.2. Implementation Steps for Extracting Watermark

The extracting procedure mainly includes the following steps:

Step 1: Filter A' to reduce the out-of-band noise by low-pass filter.

Step 2: Divide A' into M audio segments A_l', and $M \geq L + 16$.

Step 3: Perform r-level DWT on A_l' to get $De'(r, n)$.

Step 4: Separate $De'(r, n)$ into $De_1'(r, j)$ and $De_2'(r, j)$, and then implement DCT on them to obtain $C_1'(r, j)$ and $C_2'(r, j)$.

Step 5: Calculate M'_{c1} and M'_{c2}.

Step 6: If $M'_{c1} > M'_{c2}$, the extracted binary is '1', otherwise, it is '0'.

Step 7: Repeat Step 3 to Step 6 until the end of the extracting process.

Step 8: When a group of "11111111" start sign appears in the extracted binary information, the watermark begins to be extracted. When a group of "0000 0000" end sign is present, the extraction is finished.

Step 9: Generate the binary chaotic sequence $c(q)$ according to Formulas (3) and (4), and then obtain the extracted picture according to Formula (24).

4. Experimental Results and Analysis

The detail experimental parameters are shown as follows: (1) the tested original audio consists of 20 songs, sampled at 44,100 Hz and 16 bit quantization; (2) three watermark pictures with different features are shown in Figure 4. The first picture is the logo of Nanjing Metro, and its outline is very clear. The English abbreviation of Nanjing University of Posts and Telecommunications is in the second picture, and its Chinese name is in the third picture; (3) the secret key is $Ch(0.2, 3.9, 0.5)$; (4) the length of each segment is 256; (5) the level of DWT is 4; (6) the detailed wavlet coefficient is $De(4)$; (7) the adaptive embedding depth is determined according to Formula (31), and $a = 0.1$, $b = 0.3$. The experimental environment is described in the following items: (1) the computer system is Microsoft Windows XP Professional; (2) MATLAB 6.5 is used as the programming language to write all programs; (3) Cool Edit Pro V2.1 can be utilized to carry out various attacks on audio media for testing the robustness.

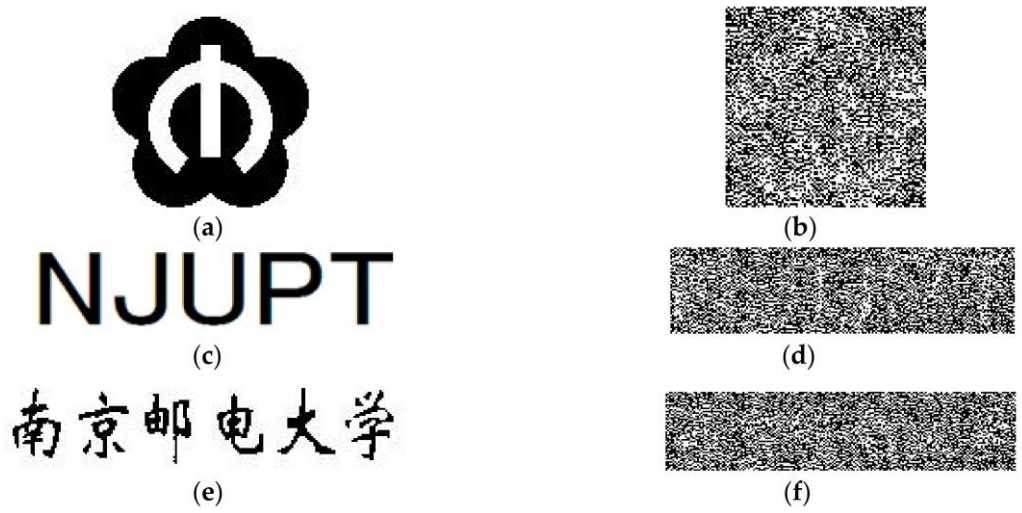

Figure 4. Three binary pictures and their encrypted pictures: (**a**) the first picture (100 × 100); (**b**) the first encrypted picture; (**c**) the second picture (200 × 50); (**d**) the second encrypted picture; (**e**) the third picture (200 × 50); (**f**) the third encrypted picture.

4.1. Capacity and Imperceptibility

The average result for SNR of the audio signals, BER and normalized correlation (NC) of the extracted watermarks and the capacity are shown in Table 1.

Each audio segment conceals 1 bit information according to the watermark embedding principle of this algorithm in Section 2.2, so the capacity is 172.27 bps. It can be seen from the average results in Table 1 that the SNR of this algorithm is 24.58 dB, higher than that in literature [4,18] with the same capacity, and both the capacity and SNR of this algorithm are better than those in literature [1,10,25], so this algorithm has better imperceptibility and larger capacity compared with the five related works.

Table 1. Average experimental results about capacity and imperceptibility (without attack).

Indexes	Proposed	[1]	[4]	[10]	[18]	[25]
SNR (dB)	24.58	N/A	23.49	21.37	18.42	20.32
Capacity(bps)	172.27	125	172.27	43.07	172.27	139.97
NC	1	/	1	/	/	/
BER (%)	0.00	0.00	0.00	/	/	0.12

Note that: / indicates no relevant datas are found in the selected literatures.

In the case of no attack, the waveform comparison charts of an audio clip (lasting about 3 s) before and after embedded watermarks are shown in Figure 5, and the corresponding spectrogram comparison charts are shown in Figure 6. Those two figures indicate the excellent imperceptibility.

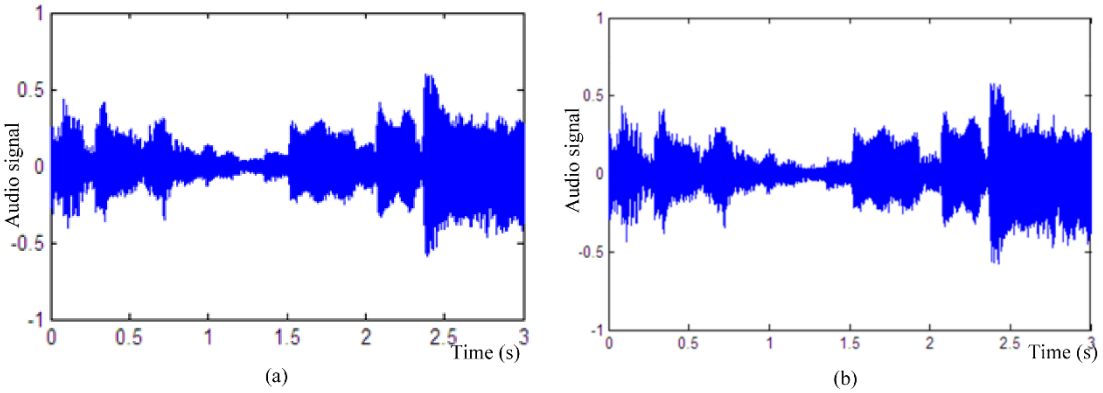

Figure 5. Waveform comparison charts of an audio clip: (**a**) original audio clip; (**b**) watermarked audio clip.

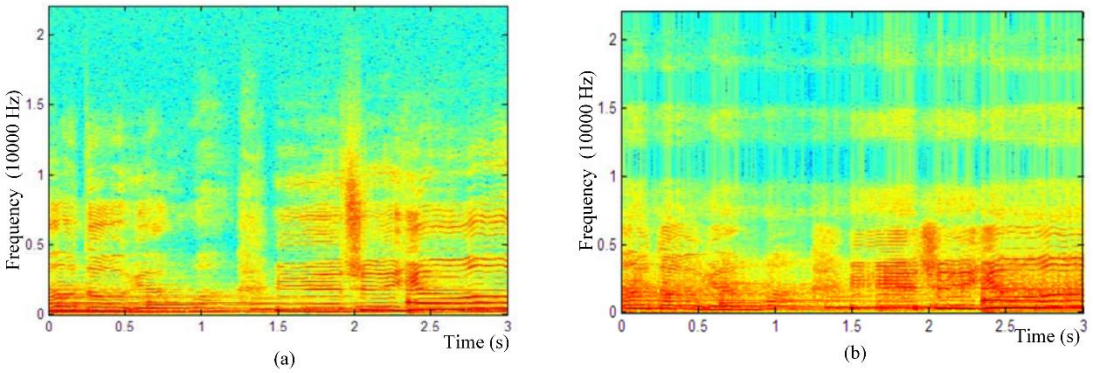

Figure 6. Spectrogram comparison charts of an audio clip: (**a**) original audio clip; (**b**) watermarked audio clip.

4.2. Robustness

Robustness is an important index for evaluating the watermarking algorithm performance. This study examines BER and NC to evaluate the robustness of this algorithm. There are several types of attacks applied to the watermarked audio signal:

(1) Gaussian noise: add 20 dB Gaussian noise.
(2) Gaussian noise: add 30 dB Gaussian noise.
(3) Gaussian noise: add 35 dB Gaussian noise.
(4) Amplitude scaling: reduce the amplitude of the watermark audio signal to 0.8.

(5) Amplitude scaling: amplify the amplitude of the watermark audio signal to 1.2.
(6) Low-pass filtering: apply low-pass filter with 4 kHz.
(7) MP3 compression: apply MP3 compression with 64 kbps.
(8) MP3 compression: apply MP3 compression with 128 kbps.
(9) echo interference: add an echo with 50 ms delay and 5% decay.
(10) Resampling: change the sampling rates by 44100-22050-44100 Hz.
(11) Requantization: change the quantization bits by 16-8-16 bits per sample.

Figures 7–9 show that the extracted pictures are very similar to the original pictures shown in Figure 4 when resisting various attacks except for the Gaussian noise with 20 dB, which indicates the strong robustness of this proposed algorithm.

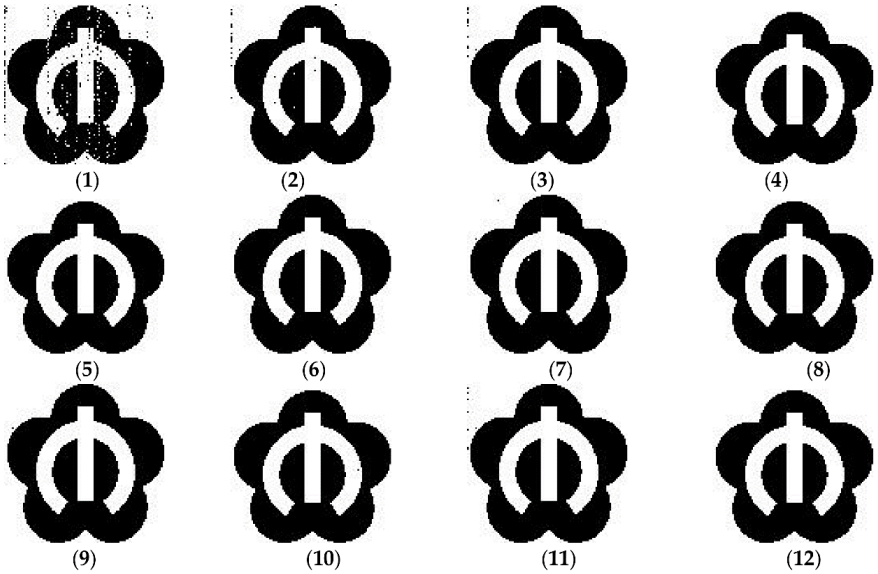

Figure 7. The extracted pictures of the first picture:(**1**) Gaussian noise (20 dB); (**2**) Gaussian noise (30 dB); (**3**) Gaussian noise (35 dB); (**4**) amplitude scaling (0.8); (**5**) amplitude scaling (1.2); (**6**) low-pass filtering; (**7**) MP3 compression (64 kbps); (**8**) MP3 compression (128 kbps); (**9**) echo interference; (**10**) resampling; (**11**) requantization; (**12**) without attack.

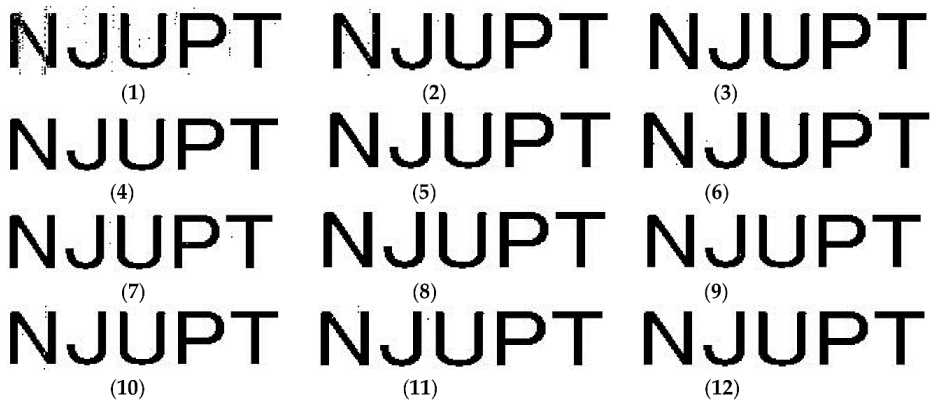

Figure 8. The extracted pictures of the second picture: (**1**) Gaussian noise (20 dB); (**2**) Gaussian noise (30 dB); (**3**) Gaussian noise (35 dB); (**4**) amplitude scaling (0.8); (**5**) amplitude scaling (1.2); (**6**) low-pass filtering; (**7**) MP3 compression (64 kbps); (**8**) MP3 compression (128 kbps); (**9**) echo interference; (**10**) resampling; (**11**) requantization; (**12**) without attack.

Figure 9. The extracted pictures of the third picture: (**1**) Gaussian noise (20 dB); (**2**) Gaussian noise (30 dB); (**3**) Gaussian noise (35 dB); (**4**) amplitude scaling (0.8); (**5**) amplitude scaling (1.2); (**6**) low-pass filtering; (**7**) MP3 compression (64 kbps); (**8**) MP3 compression (128 kbps); (**9**) echo interference; (**10**) resampling; (**11**) requantization; (**12**) without attack.

The average results for NC are shown in Table 2. The similarity between the extracted pictures and original pictures reaches over 0.98 under various attacks, which shows the strong robustness.

According to the experimental results for BER listed in Table 3, the following conclusions can be drawn:

(1) This adaptive algorithm has an excellent robustness against Gaussian noise, resampling, requantization, echo interference, MP3 compression and amplitude scaling, so it is far superior to the algorithms proposed in [1,4,10,18]. This can be seen by comparing the results in column 2 and column 4 that the robustness of this adaptive algorithm is much better than that in [4], mainly because the embedding depth of each audio segment is adaptively controlled by the overall average amplitude.

(2) The BER of this algorithm in resisting the low-pass filter is only 0.01%, which is higher than 0.39% in [1], 21.975% in [10], 28.250% in [18], and 0.12% in [25]. The average BER in case of Gaussian noise with 20dB is 1.92%, which is inferior to the algorithm in [25], so some watermark bits may be lost when resisting strong noise attacks. The 4th level wavelet coefficients will be affected by strong noise so as to reduce the robustness because this algorithm conceals the watermarks by modifying the 4th-level coefficients. As the noise becomes smaller, BER are significantly declined in 30 dB and 35 dB.

Table 2. Average results for normalized correlation (NC) under various attacks.

Attack	The First Picture	The Second Picture	The Third Picture	Average Values
(1)	0.9852	0.9881	0.9672	0.9802
(2)	0.9985	0.9989	0.9971	0.9981
(3)	0.9996	0.9998	0.9991	0.9995
(4)	1	1	1	1
(5)	1	1	1	1
(6)	1	1	1	1
(7)	0.9998	0.9997	0.9996	0.9997
(8)	1	1	1	1
(9)	1	1	1	1
(10)	1	1	1	1
(11)	0.9986	0.9992	0.9993	0.9990

Table 3. Average results for BER (%) under various attacks.

Attack	Proposed	[1]	[4]	[10]	[18]	[25]
(1)	1.92	/	2.27	/	/	1.29
(2)	0.18	/	0.22	/	/	0.31
(3)	0.06	0.78	0.07	/	/	/
(4)	0.01	2.87	0.01	0.50	0.30	0.12
(5)	0.01	17.92	0.01	0.47	0.35	/
(6)	0.01	0.39	0.01	21.97	28.25	0.12
(7)	0.06	1.95	0.08	6.85	0.12	0.00
(8)	0.01	/	0.01	4.97	1.61	0.00
(9)	0.01	/	0.01	/	0.84	/
(10)	0.01	0.00	0.01	6.45	0.12	0.00
(11)	0.12	0.78	0.14	/	0.12	0.00

5. Conclusions

An adaptive and blind audio watermarking algorithm based on chaotic encryption in a hybrid domain is proposed to combat various conventional signal-processing attacks. The watermark picture is encrypted by a chaotic sequence to improve the security of the watermark, and only a user who holds the correct key can extract the watermark in the audio signal. The encrypted binary watermark can be embedded into the high-frequency component of the audio according to the special embedding rules. The embedding depth of each audio segment is controlled by the overall average amplitude, which effectively improves the robustness and imperceptibility of this algorithm. This algorithm does not require the participation of original audio when extracting the watermark, which is very convenient for practical applications. Experimental results confirm the better performance of this proposed algorithm than the related five audio watermarking algorithms. In the process of using audio media, users often perform MP3 compression, resampling and other signal-processing operations on the audio media, just as in several attacks tested in this paper. Sometimes, users also perform some malicious operations to destroy the watermarks concealed in the audio media, such as time scaling or cutting off a piece of audio data, which can seriously damage the watermark and even cause the watermark to be lost. In the future, our research will focus on combating these malicious attacks.

Author Contributions: Q.W. put forward the idea of this paper and completed the preparation of the paper; M.W. instructed Q.W. to complete the design and construction of the experimental scheme; Q.W. and M.W. completed the submission and revision of this manuscript.

Funding: This research was funded by the National Natural Science Foundation of China for Youth (Grant No. 61602263); Research and Innovation Project for Graduate Students of Jiangsu Province, China (Grant No. KYLX_0815).

Acknowledgments: Thanks to Nanjing Metro and NJUPT for providing watermark pictures used for experiments.

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

References

1. Kaur, A.; Dutta, M.K. An optimized high payload audio watermarking algorithm based on LU-factorization. *Multimedia Syst.* **2018**, *24*, 341–353. [CrossRef]
2. Hemis, M.; Boudraa, B.; Megias, D.; Merazi-Meksen, T. Adjustable audio watermarking algorithm based on DWPT and psychoacoustic modeling. *Multimedia Tools Appl.* **2018**, *77*, 11693–11725. [CrossRef]
3. Liu, Z.; Huang, J.; Sun, X.; Qi, C. A security watermark scheme used for digital speech forensics. *Multimedia Tools Appl.* **2017**, *76*, 9297–9317. [CrossRef]
4. Wu, Q.L.; Wu, M. A Novel Robust Audio Watermarking Algorithm by Modifying the Average Amplitude in Transform Domain. *Appl. Sci.* **2018**, *8*, 723. [CrossRef]
5. Liu, S.; Pan, Z.; Song, H. Digital image watermarking method based on DCT and fractal encoding. *IET Image Process.* **2017**, *11*, 815–821. [CrossRef]

6. Han, S.C.; Yang, J.F.; Wang, R.; Jia, G.M. A robust color picture watermarking algorithm against rotation attacks. *Optoelectron. Lett.* **2018**, *14*, 61–66. [CrossRef]
7. Wang, C.Y.; Zhang, Y.P.; Zhou, X. Robust image watermarking algorithm based on ASIFT against geometric attacks. *Appl. Sci.* **2018**, *8*, 410. [CrossRef]
8. Kazemivash, B.; Moghaddam, M.E. A robust digital image watermarking technique using lifting wavelet transform and firefly algorithm. *Multimedia Tools Appl.* **2017**, *76*, 20499–20524. [CrossRef]
9. Khelifi, F.; Brahimi, T.; Han, J.G.; Li, X.L. Secure and privacy-preserving data sharing in the cloud based on lossless image coding. *Signal Process.* **2018**, *148*, 91–101. [CrossRef]
10. Lei, W.N.; Chang, L.C. Robust and high-quality time-domain audio watermarking based on low-frequency amplitude modification. *IEEE Trans. Multimedia* **2006**, *8*, 46–59.
11. Shahriar, M.R.; Cho, S.; Cho, S.; Chong, U. A high-capacity audio watermarking scheme in the time domain based on multiple embedding. *IETE Tech. Rev.* **2013**, *30*, 286–294. [CrossRef]
12. Yang, X.M.; Li, B. A new digital audio watermarking algorithm of non-uniform Discrete Fourier Transform domain based on data mining principals. *Agro Food Ind.* **2017**, *28*, 3074–3079.
13. Natgunanathan, I.; Xiang, Y.; Rong, Y.; Peng, D. Robust patchwork-based watermarking method for stereo audio signals. *Multimedia Tools Appl.* **2014**, *72*, 1387–1410. [CrossRef]
14. Pal, D.; Ghoshal, N. Secured and imperceptible data transmission through digital audio signal with reduced internal noise. *Wirel. Pers. Commun.* **2017**, *3*, 1–14. [CrossRef]
15. Hu, H.T.; Hsu, L.Y. Robust transparent and high capacity audio watermarking in DCT domain. *Signal Process.* **2015**, *109*, 226–235. [CrossRef]
16. Natgunanathan, I.; Xiang, Y.; Hua, G.; Beliakov, G.; Yearwood, J. Patchwork-Based multi-layer audio watermarking. *IEEE Trans. Audio Speech Lang. Process.* **2017**, *25*, 2176–2187. [CrossRef]
17. Qian, Q.; Wang, H.X.; Sun, X.M.; Cui, Y.H.; Wang, H.; Shi, C.H. Speech authentication and content recovery scheme for security communication and storage. *Telecommun. Syst.* **2018**, *67*, 635–649. [CrossRef]
18. Chen, S.T.; Huang, H.N.; Chen, C.J.; Tseng, K.K.; Tu, S.Y. Adaptive audio watermarking via the optimization point of view on the wavelet-based entropy. *Digit. Signal Process.* **2013**, *23*, 971–980. [CrossRef]
19. Wu, Q.L.; Wu, M. Novel Audio information hiding algorithm based on Wavelet Transform. *J. Electron. Inf. Technol.* **2016**, *38*, 834–840.
20. Hu, H.T.; Chang, J.R.; Lin, S.J. Synchronous blind audio watermarking via shape configuration of sorted LWT coefficient magnitudes. *Signal Process.* **2018**, *147*, 190–202. [CrossRef]
21. Li, J.F.; Wang, H.X.; Wu, T.; Sun, X.M.; Qian, Q. Norm ratio-based audio watermarking scheme in DWT domain. *Multimedia Tools Appl.* **2018**, *77*, 14481–14497. [CrossRef]
22. Dhar, P.K.; Shimamure, T. Blind audio watermarking in transform domain based on singular value decomposition and exponential-log operations. *Radioengineering* **2017**, *26*, 552–561. [CrossRef]
23. Hwang, M.J.; Lee, J.; Lee, M.; Kang, H.G. SVD-Based Adaptive QIM Watermarking on Stereo Audio Signals. *IEEE Trans. Multimedia* **2018**, *20*, 45–54. [CrossRef]
24. Liu, Z.H.; Luo, D.; Huang, J.W.; Wang, J.; Qi, C.D. Tamper recovery algorithm for digital speech signal based on DWT and DCT. *Multimedia Tools Appl.* **2017**, *76*, 12481–12504. [CrossRef]
25. Hu, H.T.; Chou, H.H.; Yu, C.; Hsu, L.Y. Incorporation of perceptually adaptive QIM with singular value decomposition for blind audio watermarking. *EURASIP J. Adv. Signal Process.* **2014**, *12*, 1–12. [CrossRef]