FFT-Based Data Hiding on Audio in LWT-Domain Using Spread Spectrum Technique

Gelar Budiman[1, 2, *], Andriyan Bayu Suksmono[1], Donny Danudirdjo[1]
[1]*Graduate School of Electronic Engineering and Informatics, Bandung Institute of Technology, Jl. Ganesha 1, Bandung, Indonesia*
[2]*School of Electrical Engineering, Telkom University, Jl. Telekomunikasi Terusan Buah Batu, Bandung, Indonesia*
gelarbudiman@telkomuniversity.ac.id

***Abstract*—Audio watermarking is a process to hide digital data without being seen or heard by the sense of sight or hearing. Watermaking is applied to insert the copyright on digital media, such as an image file, an audio file or a video file. In this paper, we propose watermarking procedure to embed spread spectrum watermark into frequency domain of adaptive selected subband from host audio. Lifting Wavelet Transform (LWT) is used to decompose the host audio into several subbands, and then Fast Fourier Transform (FFT) transforms selected several subbands with lowest energy. The watermark image is converted into one-dimensional signal, then it is modulated by imperceptible pseudo-noise (PN) code with controlled gain. Next, the frequency domain of audio is added by modulated and imperceptible watermark prior to transforming it to time domain by Inverse FFT (IFFT) obtaining watermarked subbands. Finally, the watermarked subbands are combined with other unused subbands by inverse LWT (ILWT) becoming the perfect version of watermarked audio. The result of this method has good robustness against most attacks from stirmark benchmark experiments, good imperceptibility with Signal to Noise Ratio (SNR) more than 30 dB and payload 172.66 bps.**

***Index Terms*—Audio watermarking; Spread spectrum; Lifting wavelet transform; Fast Fourier transform.**

I. INTRODUCTION

Along with the development of technology and passage of time, information and data exchange in the internet becomes larger and larger causing not only distributed legal data, but also the increase of falsification data illegally. The illegal dissemination of information and data exchange has created problems, such as copyright protection, authentication, and intellectual gain for unauthorized parties in digital form, such as video, image, and audio. It is necessary to develop a useful technology that will provide greater security for the future to protect the copyright from the dissemination of information and data illegally in order to reduce the unauthorized parties in taking advantage. The watermarking method is the solution for this illegal information and data dissemination problem. With the development of this method, it is expected that the copyrighted party will be able to find the perpetrator of the crime of disseminating information and data illegally, e.g., a hacker is one of the perpetrators.

Audio watermarking is a process to hide digital data without being seen or heard by human sense of sight or hearing. Watermarking is applied to mark the copyrights on digital image with varied methods. The goals of watermarking are robustness of watermark combatting any attacks, high watermark imperceptibility in audio host producing high quality watermarked audio with high Signal to Noise power Ratio (SNR) or Objective Different Grade (ODG) and high payload of watermark embedded in host audio. The term "robust" means the strength of the watermarking method to face the signal processing attacks. The detected watermark quality is better when the watermarking method is more robust to the signal processing attacks.

Several papers related to Lifting Wavelet Transform (LWT) in audio watermarking have been published. In [1], Xuesong used LWT for embedding two different watermark images into different subband. In fact, the robustness was still low against the attacks. Dhar in [2] used LWT for decomposing the host audio, and then selected subband was decomposed by QR. QR is one of decomposition algorithms producing **Q** and **R** matrices. In [2], Dhar embed the watermark into **R** matrix. Its imperceptibility is high; nevertheless, the robustness is still not good to combat the attack, especially MP3 compression attack. In [3], Lei proposed hybrid embedding method by first processing the host audio based on LWT - Discrete Cosine Transform (DCT) - Singular Value Decomposition (SVD). She tried DWT-DCT-SVD for embedding also. Those schemes were optimized by differential evolution optimization. The result is was very good in imperceptibility and robustness.

Fast Fourier Transform (FFT) is a transform method, which converts the signal into frequency domain. Several papers published audio watermarking with this transform method. In [4], Dhar used FFT that performed good robustness against resampling and filtering attacks with controlled alpha parameter for adjustable imperceptibility and robustness. In [5], Fallahpour used FFT to generate

Manuscript received 4 August, 2019; accepted 9 March, 2020.
This research was funded by the Ministry of Research, Technology and Higher Education of Indonesia in 2019. This research was performed in cooperation with the Institution.

robust and high capacity watermarked audio. He modified the selected sample in frequency domain to increase the capacity of watermark while its robustness is still high against many attacks.

Audio watermarking in Spread Spectrum (SS) method was introduced first by Kirovski [6]. He proposed Direct-Sequence Spread Spectrum (DS-SS) watermark and embedded it into host audio with controlled amplitude of watermark. Malvar continued the research with improved SS in [7]. He found that SS could be improved by controlling the amplitude of watermark adaptively depending on the average energy of host audio. The performance improvements were about 20 dB in signal to noise ratio. In [8], he proposed SS audio watermarking with synchronization and improved psychoacoustic model. In his paper, Discrete Wavelet Packet Transform (DWPT) is applied to the host audio before the watermark is embedded. Anyway, watermark payload in his method was only 8 bps and the imperceptibility was not explicitly and objectively presented. In [9], Zhang proposed improved SS with perceptual masking. His proposed method obtained imperceptible and robust watermarking method against several attacks, but his watermark capacity reached at maximum 43.07 bps. Frequency domain based audio watermarking with SS watermark was proposed by Xiang [10] also. He used DCT for transforming host audio into frequency domain. The SS-based watermarking is embedded into DCT coefficients by adding watermark with controlled gain. He claimed that his method was not only robust and imperceptible, but has high payload in 84 bps also. In fact, an iterative way to find optimum gain factor for balancing imperceptibility, robustness, and capacity took much time in embedding computation.

Our previous work, published in [11], is a multicarrier-based watermark with controlled amplitude added into time domain host audio. It is robust against several attacks, such as noise, resampling, linear speed change, equalization, echo addition, and MP3 compression with rate more than 64 kbps. However, the payload is still below 50 bps. In [12], we also published audio watermarking in frequency domain with DCT using Fibonacci sequence. The simulation result showed that the payload is high. Nevertheless, the robustness is not good.

In this paper, we propose an audio watermarking method with embedded spread spectrum watermark into frequency domain of adaptive selected audio subband. First, a host audio is decomposed by LWT into several subbands, and then FFT transforms selected several subbands with lower energy than an energy threshold. The watermark image is converted into one-dimensional signal, then it is spread by imperceptible PN code with controlled gain. Next, the frequency domain of audio is added by spread and imperceptible watermark prior to transforming it to time domain by Inverse FFT (IFFT) obtaining watermarked subbands. Finally, the watermarked subbands are combined with other unused subbands by Inverse LWT (ILWT) becoming the perfect version of watermarked audio. In the extraction phase, the watermarked audio is first decomposed by LWT, then several subbands with energy less than threshold are selected. Next, FFT transforms the selected subbands. The frequency domain of selected subbands is multiplied by a same imperceptible pseudo-noise (PN) code with PN code in embedding process. The result of multiplication in each frame is summed into a value. If the number is above or same as 0, the bit watermark extracted is "1", otherwise it is extracted as "0".

The structure of this paper is as follows. Section II describes the fundamental knowledge of the proposed audio watermarking method. Section III presents the watermarking model of the proposed method. Section IV presents the experimental result of the proposed method, and section V presents the conclusions.

II. FUNDAMENTAL KNOWLEDGE

In this section, LWT, FFT, and SS embedding and extraction theoretically is presented. LWT is a decomposition method similar to DWT, but LWT has less complexity than DWT. As described comprehensively in [1], [13], and [14], there are 3 steps that occur in LWT process, such as split, update, and predict. Similar to DWT, one level decomposition of LWT process will produce low subband signal and high subband signal. The output LWT signal length will be a half number of the original signal length. It caused N level decomposition of LWT process will obtain $2N$ outputs. In (1), $x_o(n)$ is an input signal of LWT, $x_i(n)$ is an output signal of LWT, and i is a positive integer index starting from 1 for representing the frequency in each subband as $i \in \{1, 2^N\}$. As an example, if $N = 2$, then $i \in \{1, 4\}$, and the mapping of $i \in \{1, 4\}$ means $i \in \{$LL, LH, HL, HH$\}$. LL means "Low-Low" subband locating at $i = 1$, LH means "Low-High" subband locating at $i = 2$, HL means "High-Low" subband locating at $i = 3$, and HH means "High-High" subband locating at $i = 4$. LWT and invers LWT (ILWT) statements are written in (1) and (2):

$$x_i(n) = LWT(x_o(n)), \tag{1}$$

$$x_o(n) = ILWT(x_i(n)), \tag{2}$$

where n is a discrete sample unit, and the length of $x_i(n)$ uses the following relation with the length of $x_o(n)$

$$L_i = \frac{L_o}{2^N}, \tag{3}$$

where N is a decomposition level, L_i is the length of $x_i(n)$, and L_o is the length of $x_o(n)$ in sample.

From LWT output, all subbands of $x_i(n)$ are selected for the next process, that is FFT. FFT is a fast version of Discrete Fourier Transform (DFT). The output of FFT is the same as the output of DFT. The difference between them is about their processing speed. FFT is much faster than DFT, as described in detail in the old publication about FFT and DFT in [15]. N_p point DFT and Inverse DFT (IDFT) equations are described as follows:

$$X(k) = \sum_{n=0}^{N_p-1} x(n) e^{\frac{-2\pi nk}{N_p}}, \tag{4}$$

$$x(n) = \frac{1}{N_p} \sum_{k=0}^{N_p-1} X(k) e^{\frac{2\pi nk}{N_p}}. \quad (5)$$

FFT and IFFT statements to be used in this paper are written as following equations:

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

$$x(n) = IFFT(X(k)), \quad (7)$$

where n is discrete time in sample unit, k is frequency in sample unit, $x(n)$ is a time domain signal, and $X(k)$ is a frequency domain signal. The embedding process is applied on $X(k)$ on the left side of the spectrum with the following equation

$$\mathbf{X_w} = \mathbf{X} + \alpha w \mathbf{p_m}, \quad (8)$$

where $\mathbf{X_w}$ is a vector of watermarked audio, $\mathbf{X}$ is a vector of FFT domain audio, α is gain factor of watermark to be embedded, w is watermark, which $w \in \{-1, 1\}$, and $\mathbf{p_m}$ is a filtered random code, which is generated by pseudo noise (PN) code according to Gram-Schmidt procedure as described in [16] and filtered by perceptual masking or psychoacoustic filter as in (9)

$$\mathbf{p_m} = \mathbf{p} * \mathbf{h}, \quad (9)$$

where $\mathbf{p}$ is a random code, which is generated by pseudo noise (PN) code according to Gram-Schmidt procedure, * is convolution operation, and $\mathbf{h}$ is a vector of psychoacoustic filter coefficients, which has coefficients as described in [11] and [17] as in (9). Human hearing frequencies are between 20 Hz to 22050 Hz, but the critical band of human hearing is mostly sensitive in 2000 Hz to 4000 Hz. So, the filter will be mostly covered in the critical band with the 4th order Impulse Infinite Response (IIR) on Butterworth type

$$H(z) = \frac{0.07 - 0.147z^{-1} + 0.128z^{-2} - 0.053z^{-3} + 0.009z^{-4}}{1 - 0.15z^{-1} - 0.76z^{-2}}. \quad (10)$$

To extract w from (8), correlation procedure is computed as follows

$$\hat{w} = sign\left(\mathbf{X_w}^T \mathbf{p_m}\right), \quad (11)$$

where $\hat{w}$ is extracted watermark, $\hat{w} \in \{-1, 1\}$, $sign(A)$ will decide "+1" if $A \geq 0$, otherwise it will decide "-1". Length of $\mathbf{X}$, $\mathbf{X_w}$, and $\mathbf{p_m}$ is $L_i/2$ because only a half of the output FFT coefficients are embedded due to FFT properties. According to this condition, we can formulate the length of embedded host audio as a function of watermark length, segment length, and decomposition level as follows

$$L = 2^{N+1} L_w L_s, \quad (12)$$

where L_w is watermark length in bits, L_s is segment length in sample/bit, L is host audio length needed for embedding in sample, and N is decomposition level of LWT with range 1–5. The host audio is segmented firstly before all processes mentioned above are applied. The detailed steps are presented in the next section.

III. WATERMARKING MODEL

The audio watermarking system consists of two processes: embedding process (inserting watermark into audio) and extraction process (obtaining the watermark from watermarked audio). We use black and white image as a watermark. After embedding process, the quality of watermarked audio is measured by SNR and ODG formulas. Then, watermarked audio is distorted by some signal processing attacks. Finally, watermark is extracted and the robustness performance is measured with Bit Error Rate (BER) formula.

A. Embedding Process of Audio Watermarking

The embedding process contains inputs, such as audio as a host, image logo as watermark, PN code as orthogonal code for spreading the watermark bits, and key index of PN code, which must be the same as in extraction process also. Figure 1 shows the embedding process of audio watermarking. These are several steps of embedding process with adaptive subband selection and spread spectrum framework of watermark as displayed in Fig. 1.

1. Read the binary image as $w(m, n)$ and reshape it into 1 dimension with the size 1xM. Value "0" is stated black color and changed to "-1" due to Not Return to Zero (NRZ) data type as input of spreading, and value "1" is white color. This 1-dimension watermark is assumed as $w(n)$.
2. Modulate NRZ watermark data by PN code with certain key. If $p(k)$ is the PN code, watermark "1" will be $p(k)$ and watermark "-1" will be $-p(k)$. Index k is used due to frequency domain as embedding domain. This step produces $w\mathbf{p_m}$. Length of PN code is L_s.
3. The filtered modulated data are multiplied by gain α before added by the processed host audio. This 2nd and 3rd step results are assumed as $\alpha w\mathbf{p_m}$ as watermark information in the right side of (7) before added into host audio.
4. Convert the host audio from stereo to mono. The mono audio length is then adjusted as watermark length (L) as in (12).
5. Transform $x(n)$ by LWT obtaining several subbands depend on the decomposition level used as in (1).
6. All subbands are going to the next process; each subband is transformed to frequency domain by N_p-point FFT by (6). This output FFT is assumed as $X(k)$. The result will be one-dimensional FFT coefficients. Relation between N_p and L_s from step 2 is described in (13) as follows

$$N_p = 2L_s. \quad (13)$$

7. The frequency domain signal on a half of output FFT coefficients is added by output of 3rd step as in (8) obtaining $\mathbf{X_w}$. $\mathbf{X_w}$ length is L_s or a half of N_p.

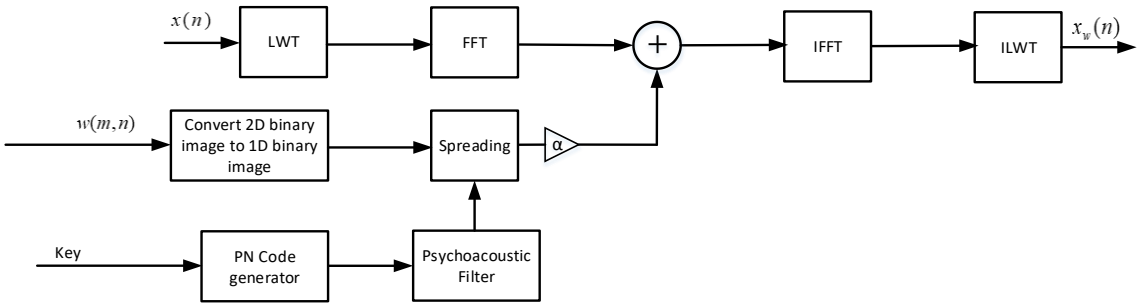

Fig. 1. Embedding process.

Thus, before IFFT processing, the remaining of **X** coefficients or right side of **X** coefficients is modified into mirror and conjugate version of $\mathbf{X_w}$ with symmetrical axis $k = N_p/2$ as in the following equation

$$\mathbf{X_{tw}} = \left[\mathbf{X_w}; reff\left(\mathbf{X_w}^*\right)\right], \tag{14}$$

where $\mathbf{X_{tw}}$ with length N_p is the input of IFFT on the next step, *reff*(*A*) is reflected or mirror version of *A*, and *A** is a conjugate of *A*.

8. Apply IFFT to $\mathbf{X_{tw}}$ obtaining $\mathbf{x_{tw}}$ by (7).

9. Apply ILWT to all subbands of $\mathbf{x_{tw}}$ obtaining $\mathbf{x_w}$ in vector symbol or $x_w(n)$ in sample unit. This is a watermarked audio signal, which is ready for distribution and is secured with embedded watermark.

The watermarked audio now is produced and we can calculate the audio imperceptibility in objective calculation like Signal to Noise Power Ratio (SNR) and Objective Different Grade (ODG), and we can also do subjective measurement by Mean Opinion Score (MOS) procedure [18]. SNR formula is computed as follows

$$SNR = 10\log_{10}\frac{\sum_{n=0}^{L-1} x^2(n)}{\sum_{n=0}^{L-1}\left(x(n)-x_w(n)\right)^2}. \tag{15}$$

In addition, from this watermarked audio, we can also compute payload of watermark with the following formula

$$L_p = \frac{L_w F_s}{L}, \tag{16}$$

where L is watermarked audio length in sample, F_s is sampling rate of audio, and L_p is watermark payload. If we replace L in (16) from (12), then (16) will be changed to the following equation

$$L_p = \frac{F_s}{L_s 2^{N+1}}. \tag{17}$$

From (17), it is clear that watermark payload is affected by sample rate, decomposition level, and segment length. Watermark length does not affect watermark payload.

B. Extraction Process of Audio Watermarking

Extraction process is a process to take the watermark from host audio. The watermark robustness is known by the result of extracted watermark with bit error rate (BER) calculation. Here are steps of extraction process as displayed in Fig. 2.

1. Read watermarked audio as $x_w(n)$.

2. Transform $x_w(n)$ by LWT obtaining several subbands depending on the decomposition level used.

3. All subbands are selected for the next process, where each subband is transformed by N_p point FFT assumed as $x_{sw}(n)$.

4. The selected subbands are going to the next process, each subband is transformed to frequency domain by FFT. This signal is assumed as $X_{tw}(k)$ or $\mathbf{X_{tw}}$ in vector.

5. Next step is to extract the watermark from $\mathbf{X_{tw}}$ by multiplying it with the $\mathbf{p_m}$ and detecting the sign of the result as in (11). If it obtains the result more than or the same as 0, then the binary extracted watermark is "+1", otherwise the binary extracted watermark is "-1". We now have 1D watermark image or $w_t(n)$ after converting "-1" to "0" or Return to Zero (RZ) number. Several extracted watermarks from several subbands are averaged and rounded.

6. Convert back the result of step 5 from 1D to 2D watermark image. Thus, we have $w_t(m,n)$.

7. After this step, we can calculate robustness parameter, i.e., bit error rate (BER) as in the equation below

$$BER = \frac{L_e}{L_w}, \tag{18}$$

where L_e is number of error extracted watermark in bits and L_w is length of watermark in bits. The overall of extraction process is displayed in Fig. 2.

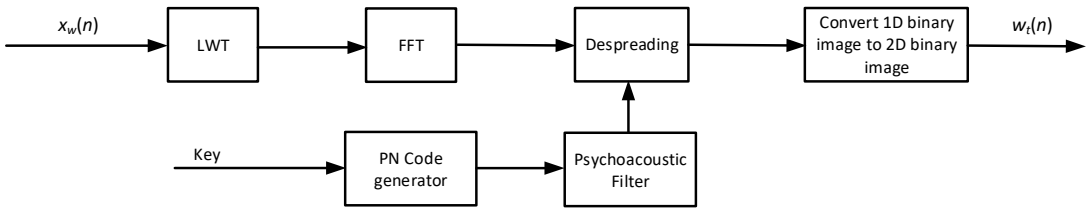

Fig. 2. Detection process.

IV. EXPERIMENTAL RESULTS

In this section, we report the experimental results of our method. Gain factor α is a valuable parameter due to its contribution to affect the balance between imperceptibility and robustness. Segment length (L_s) and decomposition level of LWT (N) have even more contribution to imperceptibility, robustness, and payload. Thus, balancing imperceptibility, robustness, and payload by changing α, L_s, and N is an interesting effort to be reported in this section.

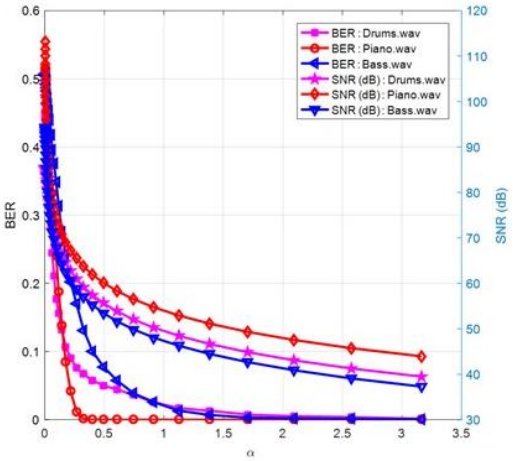

Fig. 3. Robustness and Imperceptibility vs α Without Attack.

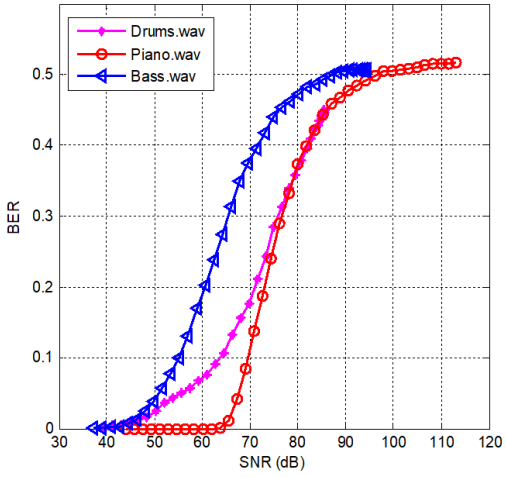

Fig. 4. BER vs SNR without Attack.

In the first experiment, the parameters L_s and N are set to 32 sample/bit and 2 sample, respectively. Thus, by (17) the payload of watermark is 172.26 bps. The bit number of watermark in this experiment is 1600 bit. Host audio files used in this experiment are drums.wav, piano.wav, and bass.wav. The variable parameter is α, which is set in range from 0.001 to 3.16. The experiment is applied without any attacks. The experiment result is displayed in Fig. 3 showing that α affects robustness and imperceptibility. Figure 3 shows not only robustness, but also imperceptibility on the right axis. The robustness of drums.wav and bass.wav is relatively less than piano.wav. To reach BER = 0, SNR of bass.wav and drums.wav are about from 40 dB to 45 dB, but SNR of piano.wav is more than 60 dB. In order to simplify the graphics in Fig. 3, it is simple to combine the graphic into only BER vs SNR and get rid of α as displayed in Fig. 4. It is clear that embedding watermark in piano.wav is much more robust than embedding watermark in drums.wav and bass.wav due to audio characteristics. The interference signal coming from audio in drums.wav and bass.wav is much higher than piano.wav.

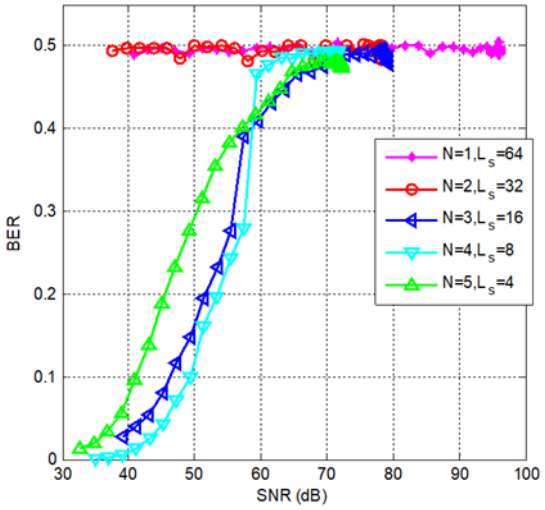

Fig. 5. BER vs SNR on Bass.wav with MP3 64 kbps Attack.

Fig. 6. Original binary watermark image.

TABLE I. ADJUSTED WATERMARKING PARAMETER VALUES OF EACH HOST AUDIO.

Host Audio	N	L_s	α	SNR (dB)
Rock.wav	3	16	7	31.24
Drama.wav	3	16	7	30.20
Piano.wav	3	16	10	33.77
Bass.wav	4	8	10	34.99
Drums.wav	3	16	9	30.21

TABLE II. ROBUSTNESS DEGRADATION DUE TO ADDITIVE NOISE ATTACK.

SNR (dB)	BER	Extracted Image	SNR (dB)	BER	Extracted Image
40	0.015		18	0.120	
25	0.024		15	0.200	
20	0.087		10	0.320	

The second experiment uses bass.wav as a host audio. Parameter α is set in the range from 0.001 to 10. There are 5 simulations with different parameters, such as N = 1 with L_s = 64, N = 2 with L_s = 32, N = 3 with L_s = 16, N = 4 with L_s = 8, and N = 5 with L_s = 4. Thus, the watermark payload in each scenario will be the same, i.e., 172.26 bps. The experiment is applied at MP3 compression attack with compression rate 64 kbps. The result of the experiment is displayed in Fig. 4. With the same payload, but different value of experiment parameters, we get much different performance. At the segment length 4 sample/bit, 8 sample/bit, and 16 sample/bit with decomposition level 3, 4, and 5, respectively, we obtain good robustness with BER < 10 % or SNR > 30 dB.

TABLE III. STIRMARK BENCHMARK RESULTS.

Attack	Parameter	Rock	Drama	Piano	Bass	Drums
LPF	9 k	0.161	0.091	0.001	0.008	0.031
Requantization	8 bit	0.069	0.015	0.000	0.003	0.013
AdditiveWhite	10 dB	0.333	0.324	0.361	0.406	0.318
AdditiveWhite	20 dB	0.142	0.087	0.153	0.243	0.076
AdditiveWhite	30 dB	0.068	0.017	0.000	0.023	0.009
Resampling	22.05 k	0.239	0.215	0.048	0.014	0.038
Resampling	11.02 k	0.233	0.191	0.051	0.013	0.036
Resampling	16 k	0.133	0.054	0.000	0.011	0.021
Resampling	24 k	0.133	0.053	0.000	0.009	0.024
LinearSpeedChange	0.990	0.081	0.015	0.000	0.019	0.014
LinearSpeedChange	0.950	0.063	0.009	0.000	0.002	0.012
LinearSpeedChange	0.900	0.049	0.006	0.000	0.002	0.009
Equalizer		0.107	0.021	0.000	0.001	0.017
Echo		0.116	0.067	0.021	0.028	0.038
MP3Compression	64 k	0.076	0.016	0.000	0.004	0.013
MP3Compression	96 k	0.074	0.019	0.000	0.003	0.012
MP3Compression	128 k	0.074	0.016	0.000	0.004	0.013
MP3Compression	192 k	0.072	0.018	0.000	0.003	0.013
AddBrumm	0.001	0.071	0.017	0.000	0.002	0.013
AddSinus	0.100	0.071	0.017	0.000	0.002	0.013
AddNoise	0.001	0.071	0.017	0.000	0.002	0.013
AddFFTNoise	0.100	0.071	0.017	0.000	0.002	0.013
Denoise		0.157	0.148	0.000	0.005	0.024
LSBZero		0.071	0.017	0.000	0.002	0.013
Echo	10	0.019	0.006	0.000	0.075	0.002
Amplify	50	0.071	0.017	0.000	0.002	0.013
Normalizer	28000	0.071	0.017	0.000	0.002	0.013
BassBoost	-10	0.071	0.017	0.000	0.003	0.013
RC-HighPass		0.074	0.017	0.000	0.003	0.012
RC-LowPass		0.073	0.014	0.000	0.002	0.013
FFT-HLPassQuick		0.065	0.013	0.000	0.003	0.011
Stat1		0.198	0.113	0.006	0.016	0.034
Stat2		0.113	0.032	0.000	0.004	0.018
FFTStat1		0.124	0.077	0.007	0.021	0.030
Smooth1		0.234	0.183	0.006	0.014	0.038
Smooth2		0.121	0.025	0.001	0.006	0.016
Invert		0.071	0.017	0.000	0.002	0.013
FFTInvert		0.071	0.017	0.000	0.002	0.013
ZeroCross		0.084	0.037	0.000	0.003	0.196
ZeroLength		0.071	0.018	0.000	0.002	0.018
ZeroRemove		0.071	0.017	0.000	0.002	0.013
Exchange		0.156	0.071	0.000	0.009	0.024

The most robust parameter value is $N = 4$ and $L_s = 8$, which reaches the lowest BER at the highest SNR compared to other parameters. The purpose of this experiment is to understand how much do the parameter values of N, L_s, and α in different host audio affect audio watermarking performance consisting of imperceptibility and robustness. This understanding will lead a way to know the correct parameter value for balancing imperceptibility and

robustness on each host audio empirically.

Adjusted α parameter, N and L_s in each host audio will be used for final experiment using stirmark benchmark. Using the same experiment as displayed in Fig. 4 for other 4 host audio files, such as drama.wav, piano.wav, drums.wav, and rock.wav empirically, we obtain α, N, and L_s as displayed in Table I. The watermarking imperceptibility of each audio is still far beyond IFPI standard, i.e., SNR minimum must be 20 dB as described in [18].

Using parameters from Table I, final experiment is applied by stirmark benchmark as standard tool for audio watermarking robustness evaluation as described in [19]. The binary image watermark is ijet.png with resolution of 40x40 pixels. The payload of each experiment is the same, i.e., 172.26 bps with SNR more than 30 dB.

From Fig. 5, it is a fair decision to choose BER maximum at 10 % or minimum SNR 30 dB as a realistic robustness and imperceptibility for selecting minimum value of α, because 10 % of BER is still in an acceptable robustness and 30 dB SNR of watermarked audio is imperceptible. The empirical proof for this acceptable robustness value is displayed in Table II. Extracted watermark image with ascending BER or descending robustness is presented. An original binary watermark image with resolution of 40x40 pixels is displayed in Fig. 6. That watermark image is embedded in drama.wav with parameter $N = 3$, $L_s = 16$ and α = 7, then its watermarked audio is attacked by additive Gaussian noise with descending SNR from 40 dB. Thus, the result of watermark extraction is a watermark image with descending robustness when SNR of additive noise attack is descending or noise power is ascending. We can see that BER value of the extracted watermark image below 10 % is acceptable because we can still understand the extracted watermark image content, although it is not perfectly reconstructed.

Table III displays the Stirmark benchmark. There are 5 columns inside the table from left to right, such as attack name, attack parameter, and BER for each host audio file in five last columns. More detailed description about the attacks is presented in [19] and [20]. From Table III, we can see that the proposed audio watermarking method obtains good robustness against most attacks. Several results with BER > 20 % happen to rock.wav and drama.wav when they are attacked by smooth1, resampling, and additive noise in certain attack parameter. The audio signal of rock.wav and drama.wav is congested in most of the audio frequency, that is the reason why they are not too robust for several attacks. The watermark interferent or host signal in both files is higher than in the other file. Nevertheless, from Table III, we can observe that our proposed method is robust. There are 48 attacks with the average BER > 10 % of the total 255 experiments. It is assumed that an experiment is robust if BER < 10 %, then the success rate of experiments by stirmark benchmark is (255 - 48)/255, or 81.18 %. Compared to our previous method in [21], where the success rate was (96 - 11)/96 or 88.54 %, the success rate of this method is lower. However, this method obtains payload with 172.26 bps, which is much higher than the payload in [21] with only 21.43 bps. Overall, the number of stirmark experiments in this paper is 255 experiments as displayed in Table III. This experiment number is much higher than the attacks in [21] with only 96 attacks.

TABLE IV. PERFORMANCE COMPARISON.

Ref.	Robustness/BER (%)				Imperceptibility/SNR (dB)	Capacity (bps)
	MP3 64 kbps	MP3 128 kbps	Requantization 8 bit	Resampling 22.05 kHz		
[22]	16–23	0–4	NA	0–6	27.44–29.33	172.26
[23]	0	NA	0	0	22.64	83
[24]	0.31	0.18	NA	NA	NA	96
Proposed	0–7.60	0–7.40	0–6.90	1.40–23.90	30.20–34.99	172.26

Our method, in this paper, has the highest capacity compared to the previous method, i.e., 172.26 bps with excellent robustness as described in the previous paragraph. Compared to the previous method, as displayed in Table IV, our proposed method also obtains higher imperceptibility than the previous work. NA means not available or no reporting. The robustness of our method is competitive with the previous method. Even though our method is not better, but the robustness is still acceptable.

V. CONCLUSIONS

We introduced spread spectrum audio watermarking in the LWT-FFT domain. SS modulated watermark is embedded in all subbands after LWT-FFT process. In the extraction, extracted watermark is calculated in each subband, averaged, and rounded. The simulation results shows that this method has good imperceptibility with SNR > 30 dB and the highest payload compared to previous work, that is 172.26 bps. The proposed method is also robust according to Stirmark benchmark experiments obtaining average BER lower than 10 % from total 255 experiments.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] C. Xuesong, C. Haiman, and W. Fenglei, "A dual digital audio watermarking algorithm based on LWT", in *Proc. of International Conference on Measurement, Information and Control* (MIC), 2012, pp. 721–725. DOI: 10.1109/MIC.2012.6273393.

[2] P. K. Dhar, "A blind audio watermarking method based on lifting wavelet transform and QR decomposition", in *Proc. of 8th International Conference on Electrical and Computer Engineering*, 2014, pp. 136–139. DOI: 10.1109/ICECE.2014.7027012.

[3] B. Lei, I. Y. Soon, and E. Tan, "Robust SVD-based audio watermarking scheme with differential evolution optimization", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 11, pp. 2368–2378, 2013. DOI: 10.1109/IIHMSP.2011.85.

[4] P. K. Dhar and I. Echizen, "Robust FFT based watermarking scheme for copyright protection of digital audio data", in *Proc. of 7th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IIHMSP, 2011, no. 2, pp. 181–184. DOI: 10.1109/IIHMSP.2011.85.

[5] M. Fallahpour and D. Megías, "Audio watermarking based on Fibonacci numbers", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 8, pp. 1273–1282, 2015. DOI: 10.1109/MSN.2014.58.

[6] D. Kirovski and H. S. Malvar, "Spread-spectrum watermarking of

audio signals", *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 1020–1033, 2003. DOI: 10.1109/TSP.2003.809384.

[7] H. S. Malvar and D. A. F. Florêncio, "Improved spread spectrum: A new modulation technique for robust watermarking", *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 898–905, 2003. DOI: 10.1109/TSP.2003.809385.

[8] X. He and M. S. Scordilis, "Efficiently synchronized spread-spectrum audio watermarking with improved psychoacoustic model", *Research Letters in Signal Processing*, pp. 1–5, 2008. DOI: 10.1155/2008/251868.

[9] P. Zhang, S. Z. Xu, and H. Z. Yang, "Robust audio watermarking based on extended improved spread spectrum with perceptual masking", *International Journal of Fuzzy Systems*, vol. 14, no. 2, pp. 289–295, 2012. DOI: 10.30000/IJFS.201206.0013.

[10] Y. Xiang, I. Natgunanathan, Y. Rong, and S. Guo, "Spread spectrum based high embedding capacity watermarking method for audio signals", *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 12, pp. 2228–2237, 2015. DOI: 10.1109/TASLP.2015.2476755.

[11] G. Budiman, A. B. Suksmono, D. Danudirdjo, K. Usman *et al*., "A modified multicarrier modulation binary data embedding in audio file", *International Journal on Electrical Engineering and Informatics*, vol. 8, no. 4, pp. 762–773, 2016. DOI: 10.15676/ijeei.2016.8.4.5.

[12] G. Budiman, A. B. Suksmono, and D. Danudirdjo, "Fibonacci sequence - based FFT and DCT performance comparison in audio watermarking", *International Journal of Engineering & Technology*, vol. 8, no. 19, pp. 209–214, 2019. DOI: 10.14419/ijet.v8i1.9.26401.

[13] R. W. Goodman, *Discrete Fourier and Wavelet Transforms: An Introduction through Linear Algebra with Applications to Signal Processing*. World Scientific, 2016. DOI: 10.1142/9835.

[14] P. K. Dhar and T. Shimamura, "Audio watermarking based on LWT and SD", in *Advances in Audio Watermarking Based on Matrix Decomposition*. Cham, Springer International Publishing, 2019, pp. 43–51. DOI: 10.1007/978-3-030-15726-5_5.

[15] D. F. G. Coelho, R. J. Cintra, N. Rajapaksha, G. J. Mendis, A. Madanayake, and V. S. Dimitrov, "DFT computation using Gauss-Eisenstein basis: FFT algorithms and VLSI architectures", *IEEE Transactions on Computers*, vol. 66, no. 8, pp. 1442–1448, Aug. 2017. DOI: 10.1109/TC.2017.2677427.

[16] H. Anton and Ch. Rorres, *Elementary Linear Algebra: Applications Version*, 11th ed. Wiley, 2013. DOI: 10.1007/978-0-387-09421-2_10.

[17] A. Atriek and A. Kaur, "Perceptually transparent watermarking of audio signals", in Proc. of *2016 6th International Conference - Cloud System and Big Data Engineering* (*Confluence*), 2016, pp. 458–462. DOI: 10.1109/CONFLUENCE.2016.7508163.

[18] A. S. Patil and G. Sundari, "An embedding of secret message in audio signal", in *Proc. of 2018 3rd International Conference for Convergence in Technology* (I2CT), 2018, pp. 1–3. DOI: 10.1109/I2CT.2018.8529549.

[19] A. Nadeau and G. Sharma, "An audio watermark designed for efficient and robust resynchronization after analog playback", *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 6, pp. 1393–1405, 2017. DOI: 10.1109/TIFS.2017.2661724.

[20] A. Lang, J. Dittmann, R. Spring, C. Vielhauer *et al*., "Audio watermark attacks : From single to profile attacks", in *Proc. of the MM&Sec '05: 7th workshop on Multimedia and Security*, ACM, 2005, pp. 39–50. DOI: 10.1145/1073170.1073179.

[21] G. Budiman, A. B. Suksmono, D. Danudirdjo, and S. Pawellang, "QIM-based audio watermarking with combined techniques of SWT-DST-QR-CPT using SS-based synchronization", in *Proc. of 2018 6th International Conference on Information and Communication Technology* (ICoICT), 2018, pp. 286–292. DOI: 10.1109/ICoICT.2018.8528727.

[22] G. Budiman, A. B. Suksmono, and D. Danudirdjo, "CPT-based data hiding in selected subband using combined transform and decomposition method", in *Proc. of International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, 2018, pp. 86–92. DOI: 10.1109/ICCEREC.2018.8712001.

[23] A. A. Attari and A. A. B. Shirazi, "Robust and transparent audio watermarking based on spread spectrum in wavelet domain", in *Proc. of 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology* (JEEIT), 2019, pp. 366–370. DOI: 10.1109/JEEIT.2019.8717415.

[24] Y. Hong and J. Kim, "Autocorrelation modulation-based audio blind watermarking robust against high efficiency advanced audio coding", *Applied Sciences*, vol. 9, no. 14, 2019. DOI: 10.3390/app9142780.