Image Hiding on Audio Subband Based On Centroid in Frequency Domain

GELAR BUDIMAN, IRMA SAFITRI, RIZKY DAMARJATI SIREGAR

Fakultas Teknik Elektro Universitas Telkom
Email: gelarbudiman@telkomuniversity.ac.id

Received 14 Agustus 2018 **|** *Revised* 6 September 2018 **|** *Accepted* 7 Januari 2019

ABSTRAK

Audio watermarking adalah mekanisme penyembunyian data pada audio. Metode penyembunyian data yang digunakan dalam penulisan ini adalah Lifting Wavelet Transform (LWT), Fast Fourier Transform (FFT), Centroid dan Quantization Index Modulation (QIM). Langkah pertama adalah host audio tersegmentasi menjadi beberapa frame. Kemudian sub-band terpilih diubah oleh FFT dengan mengubah domain sub-band dari waktu ke frekuensi. Proses centroid digunakan untuk menemukan titik pusat frekuensi untuk lokasi penyisipan untuk mendapatkan output yang lebih stabil. Proses penyematan dilakukan dengan QIM. Kinerja watermarking oleh parameter yang disesuaikan memperoleh nilai imperceptibility dengan Signal to Noise Ratio (SNR) > 21 dB, Mean Opinion Score (MOS)> 3.8 dengan kapasitas = 86.13 bps. Selain itu, untuk sebagian besar file audio terwatermark yang diserang, metode ini tahan terhadap beberapa serangan seperti Low Pass Filter (LPF) dengan f_{co}> 6 kHz, Band Pass Filter (BPF) dengan f_{co} 50 Hz - 6 kHz, Linear Speed Change (LSC) dan MP4 Compression dengan Bit Error Rate (BER) kurang dari 20%.

Kata kunci: *FFT, subband, LWT, Centroid, Audio Watermarking, QIM*

ABSTRACT

Audio watermarking is a mechanism for hiding data on audio. Data hiding methods used in this paper are Lifting Wavelet Transform (LWT), Fast Fourier Transform (FFT), Centroid and Quantization Index Modulation (QIM). The first step is to segment host audio into several frames, then the selected sub-band is changed by the FFT by changing the sub-band domain from time to frequency. The centroid process is used to find the center of frequency for the insertion location to get a more stable output. The embedding process is done by QIM. The watermarking performance by adjusted parameters obtains the imperceptibility value with Signal to Noise Ratio (SNR)> 21 dB, Mean Opinion Score (MOS)> 3.8 with a capacity = 86.13 bps. In addition, for most of attacked watermarked audio files, this method is resistant to several attacks such as Low Pass Filter (LPF) with f_{co}> 6 kHz, Band Pass Filter (BPF) with f_{co} 50 Hz - 6 kHz, Linear Speed Change (LSC) and MP4 Compression with Bit Error Rate (BER) less than 20%.

Keywords: *FFT, subband, LWT, Centroid, Audio Watermarking, QIM*

1. INTRODUCTION

With the development of information and communication technology and internet globalization, everyone can access some contents with limitless freedom especially audio. Those developments of internet and information technology could improve data transfer productivity. However, the nature of internet itself cannot be handled entirely when it comes to audio piracy. These unresponsible parties can get audio content freely, modifying it and using it for their own advantages that would harm the data's original owner. Thus, we need a method in audio content that can protect the copyrights, namely audio watermarking.

Digital watermarking is a method of embedding data information into digital multimedia content. The digital multimedia content can contain image, video or audio, while the data information can contain identity or unique data containing texts or images **(Hartung & Kutter, 1999)**. There are parameters that define the best results in watermarking, such as **(Singh & Chadha, 2013)** : 1. Imperceptibility, a watermark cannot be listened by human ear, it can only be detected by specific signal processing at machine computation. Parameters that can be counted are Signal to Noise Ratio (SNR) and Mean Opinion Score (MOS), 2. Robustness, survival of extracted watermark comparing to original watermark when watermarked audio is attacked by several attacks, such as common signal processing, geometric signal attack, or compression attack. Bit Error Rate (BER) is the value parameter representing robustness, 3. Payload, the amount of watermark that is embedded into the audio host, known as the value of C with unit bit per second (bps).

Embedding watermark in frequency domain of audio with any method was published by several authors. In **(Budiman, Suksmono, & Danudirdjo, 2017)**, authors present a design of audio watermarking system based on Fast Fourier Transform (FFT), with Lifting Wavelet Transform (LWT) and Spread Spectrum (SS) combined, but they did not describe the system robustness against MP4 compression. In **(Fan & Wang, 2008)** with Discrete Cosine Transform (DCT) and Centroid combined, the authors get the perfect satisfying extraction results. However, during some attacks, it appears that the MP3 compression (48kbps) attack shows the lowest value of extraction caused unaccepted robustness. In **(Budiman, Suksmono, & Danudirdjo, 2016)**, authors compared between FFT and DCT performance by Fibonacci embedding method, the watermark robustness in DCT can reach perfect robustness with all frame length and FFT can get BER = 0 with frame length more than 256, but authors didn't describe the robustness against any attacks. In **(Fallahpour & Megías, 2015)**, authors presented a high-capacity audio watermarking by modifying several of FFT spectrum magnitudes with Fibonacci characteristics. They showed that the method obtained high payload up to 3 kbps, with good imperceptibility and provide good robustness against common audio signal processing, one of which is MP3 compression, but MP3 compression rate is minimum 64 kbps.
Quantization Index Modulation (QIM) is popular embedding method which was first introduced in **(Chen & Wornell, 2001)**. In the development of audio watermarking research, QIM was developed by combining it with other transform or decomposition method, such as wavelet decomposition **(Hu, Chen, & Hsu, 2014)** or **(Novamizanti, Budiman, & Wibowo, 2018)**, transformation to frequency domain **(Lei, Soon, & Tan, 2013)**, SVD **(Agradriya, Perdana, Safitri, & Novamizanti, 2017)** or QR decomposition **(Dhar, 2014)**. In recent years, researchers found that embedding watermark can also be executed in centroid location of host audio, especially in the frequency domain of audio. In **(Hongxia & Mingquan, 2010)** authors proposed embedding watermark by calculating audio centroid in time domain and embedding it into hybrid domain, but authors stated that their research was for fragile audio watermarking. In **(Hongxia & Mingquan, 2010)**,

authors used DCT for transforming audio to frequency domain and calculate the centroid in frequency domain before embedding watermark into it by QIM. They described only the robustness against noise 58 dB, Low Pass Filter (LPF) 19.8 kHz, resampling (11 kHz and 22 kHz), echo, and MP3 with minimum rate 48 kbps. In **(Zhang, Liu, & Huang, 2012)**, authors used LWT and DCT before calculating centroid and embedding with QIM, anyway, authors did not describe the robustness against attack completely, as an example, MP3 attack was only carried out in one type of rate without any explanation of the rate value.

In this paper, we propose an audio watermarking system based on a centroid location in frequency domain by QIM method. We select this method due to high robustness of a signal in frequency domain and more stable value of signal in centroid location which will increase robustness also. For the embedding process, the first step host audio is segmented into several frames and get the signal into high subband based on high frequency and low subband based on low frequency by using LWT. The LWT algorithm will select which subband will be embedded. Second step, FFT is used to change the host signal from time domain to frequency domain in which the signal will be more robust. Third step, the centroid process is used to find the central point of the frequency for the insertion location resulting more stable output. Forth step, the watermark data can be embeded by using Quantization Index Modulation. Fifth step, after the watermark is embedded, Inverse FFT (IFFT) and Inverse LWT (ILWT) are required to get the watermarked audio into time domain to calculate the SNR and Objective Different Grade (ODG). Extraction process mostly same with embedding process, first step is to frame the watermarked audio and to transform using LWT to get the subband which used for embedding process before. Second step, the FFT will process a domain-changing procedure. Third step, calculate the centroid of each frame. Forth step, the watermark is extracted by using QIM. Finally, in the fifth step, the extracted watermark data is compared to original watermark data for calculating BER as watermarking robustness. The purpose of this method combination for audio watermarking is to get an audio watermarking performance with high imperceptibility and robustness against any attacks. The combination of LWT-FFT before centroid calculation is to select the signal with high power and robust domain with high capacity of watermark to be embedded. Frequency domain is a robust domain for information hiding, while QIM is a watermarking method which is suitable for hiding data in high power signal and QIM also has good imperceptibility. Centroid is chosen as a method for calculating the location of a signal to be embedded in frequency domain because centroid is a statistic calculation obtaining robust value similar with averaging calculation.

1.1. Lifting Wavelet Transform (LWT)

LWT is usually used to decrease the processing time and memory requirement. It has several advantages in comparison with conventional wavelet, (a) the LWT process calculation is more efficient because LWT complexity is lower than DWT complexity (b) It needs less memory requirements than conventional wavelet, (c) LWT is not difficult to build a non-linear wavelet decomposition, (d) it has localization features in frequency which reduce the weakness of the conventional wavelet transform **(Dhar, 2014)**. The main principle of LWT is to build a new wavelet with several advantages than conventional wavelet. These are schemes of the LWT process for the audio domain **(Sweldens, 1997)**:

1. *Split/Decomposition*, is the division of data into two parts; *odds* (x_o) and *even* (x_e). The original data $x(n)$ is divided into odd and even samples with the following formula:

$$x_e(n) = x(2n) \tag{1}$$

$$x_o(n) = x(2n + 1) \tag{2}$$

2. *Predict* (P), is a step of using a function that approximates the data set. The difference between the approaching data and the actual data is by replacing the odd elements of the data set. The remaining element becomes the input for the next step in the transformation after the data is divided into the odd part (x_o) and even part (x_e) is carried out by the calculation process within wavelet function (high pass filter) denoted by d_n, with $x_e(n)$ used in predicting $x_o(n)$ as follows:

$$d_n = x_o(n) - P[x_e(n)] \tag{3}$$

3. *Update* (U), is a step to replace even samples with average values. The result will be inputed as the next step input on the wavelet transform. The odd element is also rewritten in the original data set forming the filter. The calculation of values by scaling function (low pass filter) is indicated by $c(n)$. Here is the equation:

$$c_n = x_e(n) + U[d_n] \tag{4}$$

1.2 Fast Fourier Transform (FFT)

The Fast Fourier Transform is an enhancement algorithm of Discrete Fourier Transform (DFT), in which FFT can calculate discrete Fourier algorithms with relatively low complexity and fast calculation times. For the formula change the signal from time domain to frequency domain is as follows **(Neyman, Pradnyana, & Sitohang, 2014)**:

$$X\,(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \tag{5}$$

$$W_N = e^{-j\frac{2\pi}{N}} \tag{6}$$

where : *X(k)* is the domain transformation value, *x(n)* is the digital media block value, *N* is the amount of the data that will be altered to be a frequency domain, *n* is the sample in time domain and *k* is the sample in frequency domain.

As for the formula of Inverse-FFT change the signal back from frequency domain to time domain is as follows **(Neyman et al., 2014)**:

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) e^{j2\pi\frac{kn}{N}} \tag{7}$$

1.3 Centroid

The peak spectrum is named Sub-band Spectrum Cencored (SSC). The peak spectrum has relatively little influence when the watermark is embedded. Centroid represents the center of energy distribution of each audio frame. The spectrum centroid of each frame calculated as **(Chu & Champagne, 2008)**:

$$SC_n = \frac{(\sum_{k=1}^{n} kA_n(k))}{\sum_{k=1}^{n} A_n(k)} \quad (8)$$

where : SC_n = spectrum centroid, k = index $A_n[k]$ = the amplitude of signal, and *n* = the number of sample.

1.4 Quantization Index Modulation (QIM)

Quantization Index Modulation is one of methods that is most used to insert the watermark data into host audio. QIM can be applied in the time domain or frequency domain. The formula of QIM embedding is shown below **(Hu et al., 2014)**:

$$F'(0) = \begin{cases} A_k, if \text{ w} = 0 \; and \arg\min|F\ (0) - A_k| \\ B_k, if \; w = 1 \; and \arg\min|F\ (0) - B_k| \end{cases} \quad (9)$$

$$A_k = \left(2k + \tfrac{1}{2}\right)\Delta;\ B_k = \left(2k - \tfrac{1}{2}\right)\Delta \quad (10)$$

where :
$k = 0, \pm 1, \pm 2, \ldots$
$F(0)$ = the amplitude of original audio signal
$F'(0)$ = the amplitude of quantized
w = original watermark (bit)
k = quantization index
$nbit$ = number of quantization bits

The formula for extraction the bit of watermark is:

$$\widetilde{w} = mod\left(ceil\left(\frac{F'(0)}{\Delta}\right), 2\right) \quad (11)$$

$$\Delta = \frac{1}{2^{(nbit-1)}} \quad (12)$$

where : $\widetilde{w}$ = extracted watermark (bit)

2. METHODOLOGY

The purpose of this paper is to design and analyze the performance of audio watermarking using the LWT-FFT method with Centroid-based location determination and insertion techniques with QIM. The initial step is to design an audio watermarking block diagram. The block diagram consists of embedding and extraction process. The final step is calculating the performance of audio watermarking. Several performance parameters in audio watermarking contain SNR and ODG as the imperceptibility parameters, BER as a robustness parameter, MOS as a subjective imperceptibility parameter, and C as a watermark capacity parameter. To do this audio watermarking research, we use research methodology as following, research problem identification which we already describe in first four paragraphs of Section 1, data preparation which we describe in the beginning of Section 3, embedding and extraction process design which we describe in this section, and the experiment which we describe in Section 3, especially in Subsection 3.1.

2.1. Embedding Process

Embedding process is a procedure to insert the watermark data into the host audio. There are several steps in embedding process as shown in Figure 1. Steps of embedding process are described below :

1. Audio host is going through framing process where a whole audio is divided into some audio frames. The framing process will also limit the audio duration. Use the following equation to determine the initial sample and final sample of each frame:

$$L_h = 2^N \times L_w \times N_f \tag{13}$$

$$X = \frac{L_h}{N_f} \tag{14}$$

Where:

L_h = minimum host length required

L_w = watermark length

N = 1, 2, 3, …, 5 (decomposition level)

N_f = 128, 256,…, 2048 (frame length)

X = total of frame

2. Since the audio frames are still in time domain, the decomposition of the original image will produce a four-band data such as coefficient matrix approach i.e. Low-Low (LL), Low-High (LH), High-Low (HL), and High-High (HH). Decomposition of audio will only produce low-pass coefficients (L) and high pass coefficients (H). The lifting scheme is proposed to reduce the calculation time, for LWT simplify the problem by directly analyzing problems in the domain of integers so that LWT count more effectively and only requires a small memory space. Output of this process is called coefficient of LWT *x(n)*, consist of *low frequency* [$X_L(n)$] and *high frequency* [$X_H(n)$].
3. $x_L(n)$ will be transformed into frequency domain by Fast Fourier Transform (FFT) for centroid process that can be done in the frequency domain. Output is called *X(k).*
4. *X(k)* will form a square matrix. Centroid method is used to find out the center point amplitude of the signal, resulting the output of *X(c)*.
5. Read the binary image as *w(m,n)* and reshape it into 1 dimension by pre-processing process within the audio dimension with the size 1 × *M*. Value "0" is stated black color spreading, and value "1" is white color. This 1 dimension watermark is assumed as *w(n).*
6. Embed *X(c)* matrix and *w(n)* with QIM process by using Equation (9) and (10). Output of this process is called $X_w(c)$.
7. Inserting modified magnitude of watermarked audio to defiling $X_w(c)$ to become $X_w(k)$, in the location of centroid of frequency domain.
8. Apply Inverse-FFT process to convert $X_w(k)$ (freq. domain) become $x_{Lw}(n)$ in time domain.
9. Apply Inverse-LWT process to merge the preferred subband that has been processed with other subband and obtain an audio signal that has been watermarked. An audio signal is called watermarked audio [$x_w(n)$]. Then, calculate the SNR and ODG value to measure the watermarked audio quality.

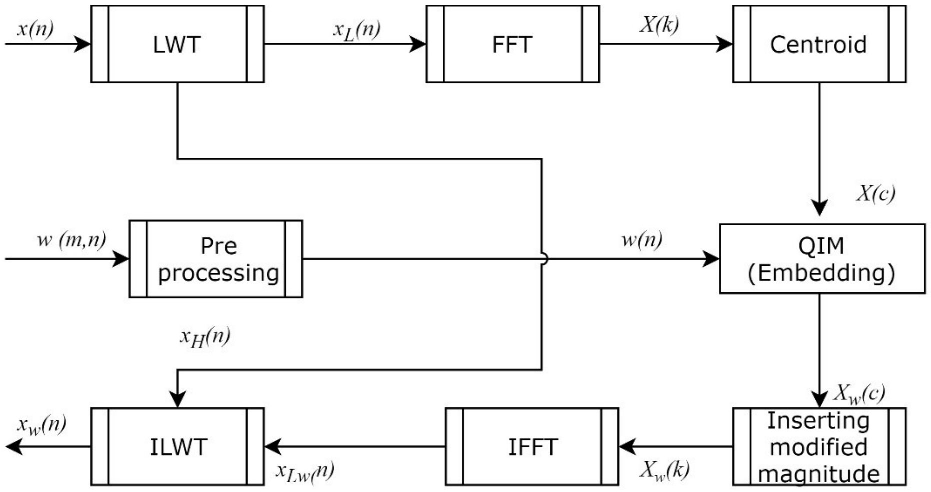


Figure 1. Embedding Process

2.2. Extraction Process

To get the watermark data again from the audio watermarked is what the extraction process for, the extraction process mostly same with embedding process. For more detail, the flowchart and extraction process shown in Figure 2. Steps of extraction process are described below.

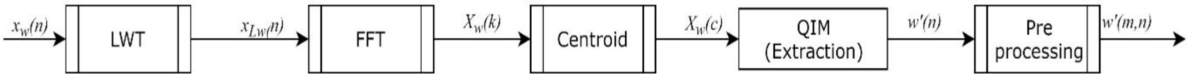


Figure 2. Extraction Process

1. Decompose watermarked audio $x_w(n)$ by using N-level LWT into several subband, as an example, 2-level LWT obtains LL, LH, HL, and HH. Select the $x_{Lw}(n)$ as the low frequency sub-band for next process.
2. Convert audio host in spatial domain as $x_{Lw}(n)$ to $X_w(k)$ in frequency domain with FFT process.
3. $X_w(k)$ will form a square matrix. Apply centroid method to find out the center point amplitude of the signal for detected location of the watermark which has been embeded. The output will be marked as $X_w(c)$.
4. After detecting location of the watermark by centroid process, the watermark extraction process is performed from the component value by QIM method. It will produce the output of *w'(n).*
5. Perform conversion process in the post-processing block from one dimension (1D) to two dimension (2D). After that, convert each bits into pixel [signed as *w'(m,n)*] .
6. Calculate the Bit Error Rate (BER) value.

3. RESULT AND DISCUSSION

In this chapter we describe about the result and analysis from the process in the previous chapter. In the experiment, we use five host audio files with *.wav file format which the duration of each file is about one minute The host audio that is used includes host.wav, piano.wav, guitar.wav, bass.wav and drums.wav. The watermark image size is 20x140 pixels from file elkomika.png as shown in Figure 3.

Figure 3. Watermark Image

3.1 System Parameters Testing

In this subsection, we disscuss about finding the best parameter before the attack and the optimization process. There are 4 parameters that are used, they are level of decomposition (N), length of frame (*Nframe*), number of quantization bits (*Nbit*) and threshold. Table 1 below shows the best parameter for attacking process that had been tested with host.wav as the type of default host audio.

Tabel 1. Input of Parameters Pre-Attack and Pre-Optimization

Decomposition Level (*N*)	**Length of frame (*N_f*)**	**Number of quantization bits (*nbit*)**	**Threshold (*thr*)**
4	2048	3	0.9

The output of audio watermarking process with the input parameters as initial parameters above are SNR = 37.92 dB, BER = 0, Capacity = 1.34 bit/s. By these parameters, we get the extracted watermark exactly the same as original watermark without attack. Then, using those parameters, we attack the watermarked audio by several types of attacks. There are 8 types of attacks which are used in the robustness test of this watermarking audio system. The types of attacks that is used such as LPF, Band Pass Filter (BPF), noise, resampling, time scale modification (TSM), linear speed change (LSC), MP3 and MP4 compression. In this experiment, 5 different audio types are used, such as host.wav, piano.wav, guitar.wav, drums.wav, and bass.wav.

Table 2. The robustness test result with initial parameters

Attack	**Criteria**	**BER**				
		host	**piano**	**gitar**	**drums**	**bass**
LPF	3kHz	0.37	0.39	0.37	0.37	0.39
	6kHz	0.37	0.35	0.37	0.37	0.37
	9kHz	0.37	0.36	0.36	0.35	0.38
BPF	100Hz-6kHZ	0.37	0.38	0.38	0.38	0.37
	50Hz-6kHz	0.37	0.36	0.37	0.38	0.37
	25Hz-6kHz	0.37	0.38	0.37	0.38	0.37
Noise	0 dB	0.37	0.37	0.37	0.38	0.37
	10 dB	0.38	0.38	0.37	0.38	0.37
	20 dB	0.36	0.36	0.38	0.38	0.38
Resampling	22.05kHz	0.37	0.36	0.37	0.37	0.38
	11.025kHz	0.36	0.32	0.36	0.37	0.37
	16kHz	0.31	0.25	0.32	0.35	0.37
TSM	1%	0.37	0.37	0.38	0.36	0.37
	2%	0.38	0.37	0.38	0.37	0.37
	4%	0.37	0.37	0.38	0.36	0.37
	1%	0.38	0.36	0.37	0.38	0.37

Attack	Criteria	BER				
		host	piano	gitar	drums	bass
Linear Speed Change	5%	0.38	0.36	0.37	0.38	0.37
	10%	0.38	0.35	0.35	0.37	0.37
MP3 Compression	32kHz	0.36	0.36	0.36	0.38	0.37
	64kHz	0.38	0.33	0.38	0.37	0.37
	128kHz	0.28	0.2	0.3	0.34	0.35
	192kHz	0.08	0.08	0.08	0.23	0.27
MP4 Compression	32kHz	0.38	0.35	0.37	0.38	0.36
	64kHz	0.38	0.37	0.38	0.38	0.36
	128kHz	0.38	0.35	0.37	0.38	0.36
	192kHz	0.38	0.35	0.37	0.38	0.36
Average		**0.36**	**0.34**	**0.36**	**0.37**	**0.37**

From Table 2, it is displayed that the average BER of extracted result of each watermarked audio is about 0.34 to 0.37 which means bad quality watermarks. The system shows that the watermark data cannot survive to the various attacks (Low Pass Filter, Band Pass Filter, Noise, Resampling, Time Scale Modification, Linear Speed Change, MP3 and MP4 Compression). Next, we select 5 host audio files with an attack in each file for parameter optimization. This parameter optimization is performed in order to get better robustness than unoptimized one for 5 host audio files with a selected attack in highlight cell displayed in Table 2.

3.2 Optimized Parameter

After the 5 types of host audio attacked, we choose 5 samples of host audio and attacks that will be optimized. The type of attack and host audio that will be optimized based on the BER value which is still possible to do optimization (BER<0.4). The samples are Resampling 16k, MP2 Compression 64k, MP3 Compression 128k, MP4 Compression 32k and BPF cut-off 100-6k. The results of 5 optimized parameters are shown in Table 3.

Table 3. Comparison Before and After Optimization

Attack Type	Before Optimization			After Optimization		
	SNR	**BER**	**Capacity**	**SNR**	**BER**	**Capacity**
Resampling 16 kHz	37.9	0.31	1.34	28.13	0	26.92
MP3 Compression 64 kHz	49.4	0.33	1.34	31.25	0	53.83
MP3 Compression 128 kHz	42.5	0.3	1.34	23.90	0	26.92
MP4 Compression 32 kHz	54.6	0.38	1.34	21.23	0.13	53.83
BPF cut-off 100 Hz-6kHz	63.2	0.37	1.34	30.44	0.25	86.13

The robustness after optimization is increasing, it means that the BER is decreasing. Anyway, if the robustness is increasing, then the other performances will decrease. As an example, for BPF attack, after optimization, BER decreases to 0.25 from 0.37, as the consequence, the SNR decreases from 63.2 dB to 30.44 dB. This is happening because there is a trade off between BER and SNR. Based on Table 3 and 4, we choose the parameter resulted of BPF attack with cut-off 100-6k as the optimized parameter for all attacks, because the average BER is the lowest. Thus, the selected optimized parameters can be used for next experiment to measure the robustness from all attacks. The best optimized parameters are shown in Table 5.

Table 4. Average BER Values of All Attacks Post-Optimization

Adjusted Parameter	1	2	3	4	5
Audio Host.	**host.wav**	**piano.wav**	**gitar.wav**	**drums.wav**	**bass.wav**
Average BER of all attack	0.28	0.23	0.26	0.193	0.191

Table 5. Optimum Input Parameter For All Attacks

Decomposition Level (N)	Length of frame (*N_f*)	Number of quantization bits (*nbit*)	Threshold (*thr*)
3	64	1	0.009

3.3 Robustness Test Results from Optimized Parameter

The input parameters in Table 5 was already attacked with all types of attacks. The average BER value of bass.wav before optimization is 0.37 and after optimization is 0.191. It means that after optimization the BER value decrease more than 40%. Thus, in Table 6 the result of watermark extraction is shown.

Table 6. The result of Optimization For All Attacks

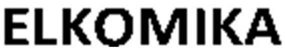
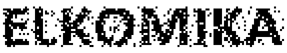
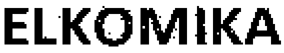
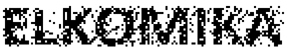
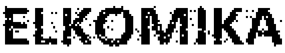
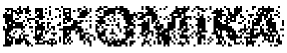
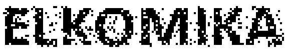
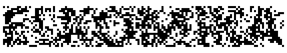
Attack	Criteria	BER				
		host	**piano**	**gitar**	**drums**	**bass**
LPF	3k	0.41	0.35	0.36	0.22	0.33
	6k	0.05	0.13	0.09	0.21	0.23
	9k	0.02	0.12	0.04	0.18	0.17
BPF	100-6k	0.38	0.33	0.39	0.26	0.25
	50-6k	0.06	0.13	0.09	0.21	0.22
	25-6k	0.05	0.13	0.09	0.24	0.24
Noise	0 dB	0.31	0.38	0.36	0.38	0.38
	10 dB	0.26	0.33	0.24	0.35	0.33
	20 dB	0.18	0.38	0.17	0.25	0.33
Resampling	22.05k	0.06	0.20	0.11	0.26	0.25
	11.025k	0.09	0.19	0.11	0.25	0.27
	16k	0.03	0.07	0.03	0.14	0.10
TSM	1%	0.14	0.14	0.13	0.18	0.13
	2%	0.22	0.22	0.22	0.24	0.22
	4%	0.32	0.22	0.20	0.43	0.36
Linear Speed Change	1%	0.00	0.00	0.00	0.03	0.00
	5%	0.00	0.00	0.00	0.02	0.00
	10%	0.00	0.00	0.00	0.02	0.00
MP3 Compression	32k	0.21	0.20	0.15	0.29	0.25
	64k	0.12	0.07	0.05	0.29	0.15
	128k	0.01	0.00	0.02	0.29	0.03
	192k	0.00	0.00	0.00	0.29	0.00
MP4 Compression	32k	0.09	0.17	0.07	0.23	0.20
	64k	0.09	0.17	0.07	0.23	0.20
	128k	0.09	0.17	0.07	0.23	0.20
	192k	0.09	0.17	0.07	0.23	0.20
Average		**0.13**	**0.16**	**0.12**	**0.23**	**0.19**

Comparing to Table 2, Table 6 describes that the overall robustness is much better. The system with adjusted parameter is moslty more robust to various attacks in some hosts with the average BER decreased with the range 0.12 to 0.23. The most robust watermarked audio against the attacks is gitar.wav with the lowest average BER, 0.12. And the weakest watermarked audio against the attacks is bass.wav with the highest average BER, 0.23.

3.4 Watermark Degradation Quality

The BER value of the watermark resistance test shows the quality of the watermark image extracted from the watermarked audio that is attacked. The higher the BER value, the worse the quality of the watermark image extracted. But there is a maximum BER value limit on watermark images that are still acceptable to humans because visually human can still understand the contents of the watermark image even if it is damaged. The maximum limit of the BER value from the watermark image depends greatly on the resolution of the watermark image inserted in the audio. In the audio watermarking with the above method, the elkomika.png image with a resolution of 20x140 is inserted where the original display can be seen in table 4.8 at BER = 0. When the watermark image is damaged and obtained BER = 0.01 to 0.15, it can be seen in Table 7 that the image can still be understood by humans, but when the watermark image has a level of damage with BER = 0.23, humans do not understand the contents of the picture. This shows that if the watermark resistance has BER below 0.2 at 20x140 image resolution according to elkomika.png image, then the extracted watermark image can still be received, or in other words the watermark is still resistant to any attacks if the watermark quality is still in the BER range or less than 20%.

Tabel 7. Quality Degradation of Watermark Image with Various BER

BER	Watermark Image	BER	Watermark Image
0		0.12	
0.01		0.15	
0.05		0.23	
0.1		0.3	

3.5 Watermarked Audio Quality

To measure the quality of audio that has been inserted with watermark, we perform subjective and objective measurement. Subjective measurement is signed as Mean Opinion Score (MOS). The objective measurement is known as Signal to Noise Ratio (SNR). The MOS value is rated by 30 respondens by listening to 5 types of the original host audio and 5 types of watermarked audio. SNR are measured by program with following formula. The audio quality in SNR and MOS is displayed in Table 8.

$$SNR = 10\ log_{10} \frac{\sum_{i=1}^{L} x^2(n)}{\sum_{i=1}^{L} |x(n) - \hat{x}(n)|^2} \tag{15}$$

where $x(n)$ = original audio, $\hat{x}(n)$ = watermarked audio.

Table 8. MOS and SNR for 5 type of host audio

Host Audio	MOS	SNR (dB)
host.wav	3.87	28.13
piano.wav	3.98	31.26
guitar.wav	4.14	23.90
drums.wav	3.97	21.24
bass.wav	3.98	30.44

From Table 8, the highest quality objectively of watermarked audio is piano.wav with SNR=31.26 dB, but the highest quality subjectively is guitar.wav with MOS=4.14. The value of MOS depends on human capability in hearing the audio, so the low difference of subjective and objective measurement as shown in above table, still makes sense.

3.6 Audio Watermarking Performance Comparison

In order to understand how well this method performs, we compare this method with the previous method in **(Fan & Wang, 2008)** and **(Hongxia & Mingquan, 2010)**. Two previous methods above were also using centroid as watermark location calculation, but they presented different technique of host audio pre-processing. Table 9 displays performance comparison consisting of imperceptibility, robustness and capacity parameter performances. NA means not available. In Table 9, our method has biggest watermark capacity on 86.13 bps with accepted imperceptibility. Anyway, our method obtains lower robustness and lower imperceptibility than the previous method. High watermark capacity in our method pays the low robustness and low imperceptibility. Nevertheless, the robustness and imperceptibility performance in our method are still in accepted subjective range as we already describe in section 3.4 and 3.5.

Table 9. Performance Comparison With Previous Method

Author	SNR (dB)	Payload (bps)	BER						
				Resampling			MP3		
			LPF 9k	44.1k-16k-44.1k	44.1k-22k-44.1k	44.1k-11k-44.1k	32k	64k	128k
(Fan & Wang, 2008)	31.82-50	43	NA	NA	0.09	0.09	NA	0.09	0.02
(Hongxia & Mingquan, 2010)	42-46	NA	NA	0.01-0.02	0-0.001	NA	0	0	0
Our method	21.24-31.26	86.13	0.02-0.18	0.03-0.14	0.06-0.26	0.09-0.26	0.15-0.2	0.05-0.29	0-0.29

5. CONCLUSION

In this paper we combined Lifting Wavelet Transform, Fast Fourier Transform, Centroid calculation and QIM for the embedding method. The proposed system has good robustness against several attacks at most of host audio files with the BER value is less than 20% as accepted robustness. Several attacks on which the system with most host audio files are robust, such as LPF with cut off 9k, resampling with rate 16k, TSM 1%, LSC for all criteria, MP3 compression with rate more than 64kbps and MP4 compression with rate more than 32kbps. The imperceptibility is also good for all type of host audio with range SNR 21.24 dB to 31.26 dB. From survey, MOS is also in good range, between 3.87 to 4.14. The capacity or payload of watermark to be embeded in the audio with optimized parameter is high, that is 86.13 bps.

ACKNOWLEDGEMENT

This research is supported by Research and Community Service of Universitas Telkom.

REFERENCES

Agradriya, B. A. F., Perdana, F. K., Safitri, I., & Novamizanti, L. (2017). Watermarking Technique Based On Arnold Transform. In *2017 2nd International Conference on Automation, Cognitive Science, Optics, Micro Electro-Mechanical System, and Information Technology* (pp. 17–21).

Budiman, G., Suksmono, A. B., & Danudirdjo, D. (2016). Fibonacci Sequence – based FFT and DCT performance comparison in Audio Watermarking. In *The International Conference on Science, Engineering, Built Environment, and Social Science (ICSEBS)* (pp. 2–8).

Budiman, G., Suksmono, A. B., & Danudirdjo, D. (2017). FFT-Based Audio Watermarking in Adaptive Subband with Spread Spectrum Framework. In *2017 2nd Advanced Research in Electrical and Electronic Engineering Technology (ARIEET)* (pp. 3–8).

Chen, B., & Wornell, G. W. (2001). Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, *47*(4), 1423–1443. https://doi.org/10.1109/18.923725

Chu, W., & Champagne, B. (2008). A Noise-Robust FFT-Based Auditory Spectrum With Application in Audio Classification. In *International Conference on Signal Processing Proceedings, ICSP* (Vol. 16, pp. 2729–2733). https://doi.org/10.1109/ICOSP.2008.4697712

Dhar, P. K. (2014). A Blind Audio Watermarking Method Based on Lifting Wavelet Transform and QR Decomposition. *8th International Conference on Electrical and Computer Engineering*, 136–139.

Fallahpour, M., & Megías, D. (2015). Audio Watermarking Based on Fibonacci Numbers. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *23*(8), 1273–1282. Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=7103318&url=http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7103318

Fan, M., & Wang, H. (2008). Centroid-based Robust Audio Watermarking Scheme. In

International Conference on Audio, Language and Image Processing (pp. 476–479). https://doi.org/10.1109/ICALIP.2008.4590170

Hartung, F., & Kutter, M. (1999). Multimedia watermarking techniques. In *Proceedings of the IEEE* (Vol. 87, pp. 1079–1107). IEEE. https://doi.org/10.1109/5.771066

Hongxia, W., & Mingquan, F. A. N. (2010). Centroid-based semi-fragile audio watermarking in hybrid domain, *53*(3), 619–633. https://doi.org/10.1007/s11432-010-0058-0

Hu, H., Chen, S., & Hsu, L. (2014). Incorporation of Perceptually Energy-Compensated QIM into DWT-DCT Based Blind Audio Watermarking. In *International Conference on Intelligent Information Hiding and Multimedia Signal Processing* (Vol. 1, pp. 0–4). https://doi.org/10.1109/IIH-MSP.2014.191

Lei, B., Soon, I. Y., & Tan, E. (2013). Robust SVD-Based Audio Watermarking Scheme With Differential Evolution Optimization. *IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING*, *21*(11), 2368–2378.

Neyman, S. N., Pradnyana, I. N. P., & Sitohang, B. (2014). A New Copyright Protection for Vector Map using FFT-based Watermarking. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *12*(2), 367. https://doi.org/10.12928/TELKOMNIKA.v12i2.1975

Novamizanti, L., Budiman, G., & Wibowo, B. A. (2018). Optimasi Sistem Audio watermarking Menggunakan Stationary Wavelet Transform Dan Statistical Mean Manipulation. *ELKOMIKA*, *6*(2), 165–179.

Singh, P., & Chadha, R. (2013). A Survey of Digital Watermarking Techniques, Applications and Attacks. *International Journal of Engineering and Innovative …*, *2*(9), 165–175. https://doi.org/10.1109/INDIN.2005.1560462

Sweldens, W. I. M. (1997). The Lifting Scheme: A Construction of Second Generation Wavelet. *SIAM Journal Mathematical Analysis*, *29*(2), 1–35.

Zhang, Q., Liu, Z., & Huang, Y. (2012). Adaptive Audio Watermarking Algorithm Based on Sub-band Feature. *Journal of Information & Computational Science*, *2*(February), 305–314.