International Journal of Engineering & Technology, 8 (1.9) (2019) 209-214

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Fibonacci Sequence – based FFT and DCT Performance Comparison in Audio Watermarking

Gelar Budiman [1], Andriyan B. Suksmono[2], Donny Danudirdjo[3]

1School of Electrical Engineering, Telkom University, Indonesia
,2,3Graduate School of Electrical and Informatics Engineering, Bandung Technology Institute, Bandung, Indonesia
**Corresponding author E-mail: gelarbudiman@telkomuniversity.ac.id*

Abstract

Audio watermarking is a manner to hide watermark into the audio for copyright protection. Recently, there are many techniques based on audio watermarking. Frequency domain based audio watermarking is one of audio watermarking technique that has good watermark robustness against many attacks, good watermarked audio quality, and also high watermark payload. In this paper, FFT and DCT performance will be compared as transform technique for data hiding in audio. Host audio is first transformed into frequency domain in frame-based by FFT or DCT, then watermark is embedded into the frequency domain signal by Fibonacci sequence rule. Different than DCT which can embed watermark on full frame, in FFT, only a half of frame that can be embedded by watermark due to FFT properties. After embedding, the frequency domain signal is transformed to time domain by IFFT or IDCT to get watermarked audio. The simulation result of frequency based audio watermarking comparing FFT and DCT transform method shows that watermark payload for perfect robustness at no attack condition could reach up to 70 bps for FFT and 500 bps for DCT. With good watermarked audio quality due to ODG > -1 and SNR > 30 dB. Due to this performance, DCT is a recommended transform method for audio watermarking technique to obtain high imperceptibility, strong robustness and high capacity than FFT.

Keywords: *Audio watermarking, copyright, frequency domain, embedding, FFT, DCT, fidelity, Fibonacci*

1. Introduction

Audio watermarking is one of information hiding type in audio file as host signal. Information hiding in audio file has good challenges due to the limited perceptual of human audibility (Bender, Gruhl, Morimoto, & Lu, 1996) called human auditory system (HAS). This fact caused many researches in audio processing using HAS characteristics, such as audio watermarking and audio compression.

The performance of audio watermarking depends on several criterias. First criteria is imperceptibility or fidelity in which the watermarked audio perceptually same as host audio. The performance parameter for this criteria consists of objective quality that calculated by formula or model and subjective quality that is taken from survey to several persons. The objective quality parameter could be PEAQ, SNR, LLR, etc as described in (ITU-R, 1998). In this paper, PEAQ and SNR will be used for performance analysis. Second criteria is watermark robustness against audio attack in which the robustness is expected to zero error or BER 0. Third criteria is high payload. This criteria is not priority criteria as fidelity and robustness because payload will follow the duration of host audio. Normally, host audio is music/audio that has duration is about 3 to 6 minutes. And that duration is enough for watermark embedding with high resolution.

Dymarski in (Dymarski & Markiewicz, 2014) used another techniques to embed watermark data in frequency domain. He used log-spectrum for embedding data. But he added dirty paper codes and LDPC for improving the robustness. Iynkaran (Natgunanathan et al., 2012) proposed a patchwork method on audio watermarking using discrete cosine transform (DCT) in the embedding process. In (Xiang, Member, Natgunanathan, Guo, & Member, 2014) Y. Xiang and Iynkaran has added synchronization bit that is useful to provide audio watermarking robustness against attacks desynchronization attack such as, pitch, time scaling, and jitter.

Pranab and Tetsuya (Dhar & Shimamura, 2015) use the merger of three methods of transformation in the process of insertion of data in the audio. Such method is DWT, DCT, and SVD. Its embedding method is using the quantization of the SVD results. Yiqing Lin and Waleed (Lin & Abdulla, 2015) published a book about audio watermarking which explains in detail about some of the basic methods of audio watermarking, among others : LSB method, phase coding, spread spectrum, echo hiding, DWT, and histogram techniques. And she concludes with a discussion of audio watermarking method proposed for performance with excellent durability, capacity, and better imperceptibility. In the proposed audio watermarking, Yiqing Lin and Waleed used blocks in the process of watermark data embedding utilizing the psychoacoustic models and FFT. Embedding is done in the frequency domain by first selecting signal in the time domain and the frequency domain using Gammatone filterbank as well as selecting the watermark area at the time domain by applying a certain threshold. In order to handle synchronization, bit insertion is applied, so that the watermark is resistant to desynchronization.

Fallahpour (Fallahpour & Megías, 2015) did a combination of OFDM method to insert data utilizing the Fibonacci sequence of numbers on a subcarrier signal after converted to the frequency domain with FFT. Embedding is utilizing the FFT techniques and changes in host audio signal into a frequency domain and then the distribution of the frame, and embed watermark with a modified spectrum in selected samples using Fibonacci numbers. By utilizing the Fibonacci sequence, Mehdi easily and adaptively modifiedy the sample frequency. And simulation testing proved that the insertion of the Fibonacci numbers closest to the sample to produce FFT spectrum watermarking technique that is resistant to attack. And the result is a significant increase in both capacity and imperceptibility while maintaining the quality of public, it is also robust to various attacks on audio watermarking.

This paper describes the process for evaluating watermarking technology comparing FFT and DCT performance as transform method for embedding watermark. Previous work has been published by Fallahpour (Fallahpour & Megías, 2015) as described in previous paragraph. Modification from (Fallahpour & Megías, 2015) is replacing FFT by DCT and finding out the performance difference between FFT and DCT. There are also slightly modifications on framing or segmentation of host audio. This paper proposes framing location before FFT or DCT process. In (Fallahpour & Megías, 2015) the segmentation was done after FFT process. The proposed embedding will be done at every range of found two Fibonacci number in which its magnitude is increased. The embedding will be stopped when there are no watermark bits to embed. Comparison performance is needed to find out the better transform method than FFT for audio watermarking application. In the previous research which written by Mehdi Fallahpour, Mehdi (Fallahpour & Megías, 2015) described that FFT produced robust watermark against MP3 attacks, and also produced high capacity audio watermarking. In this paper, we tried to replace FFT by DCT and want to know whether DCT is better or not than FFT.

This paper consists of 5 sections, such as : introduction, basic theory containing DCT, FFT and Fibonacci sequence, watermarking model containing embedding and extraction steps, simulation result containing fidelity, robustness and watermark payload and conclusion. The significant contribution of this audio watermarking method is to gain the better performance of audio watermarking, either the watermarked audio imperceptibility, higher robustness or higher capacity of watermark to be embed in audio, thus the overall performance should be better than the previous research by Mehdi Fallahpour.

2. Basic Theory

Proposed audio watermarking will compare DCT and FFT as transform method before embedding is executed. DCT and FFT has same purpose for signal processing, to convert signal from discrete time domain to discrete frequency domain. Anyway, DCT has several differences than FFT. First, DCT length for embedding is twice longer than FFT. Second, DCT value is real. It causes simpler embedding than FFT. In FFT embedding must concern about the imaginary value of FFT outputs. In DCT there is nothing worry about imaginary value.

FFT is faster version of discrete fourier transform (DFT), but their output are same at all. DFT input is discrete time signal at N point as described in this equation (Hayes, 1999):

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

IDFT is inverse of DFT process (Hayes, 1999):

$$x(n)=\frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{\frac{j2\pi kn}{N}},\ \ 0\le n\le N-1 \tag{2}$$

By fast algorithm using butterfly scheme, DFT is decomposed into smaller sequences becoming two alternatives of FFT : the decimation in time FFT and decimation in frequency FFT. After decimation in time process, we obtain FFT formula (Hayes, 1999) :

$$X(k)=\sum_{n=0}^{N/2-1} x(2n)e^{\frac{-j2\pi kn}{N/2}}+e^{\frac{-j2\pi k}{N}}\sum_{n=0}^{N/2-1} x(2n+1)e^{\frac{-j2\pi kn}{N/2}} \tag{3}$$

The FFT output is complex number because of complex exponential factor inside DFT that is calculated. In DCT, there is no complex exponential factor, but only cosine factor as equation below (Wu & Shin, 1995):

$$X(k)=l(k)\sum_{n=0}^{N-1} x(n)\cos\left(\frac{\pi(2n+1)k}{N}\right),$$

where $0\le k\le N-1$, and (4)

$$l(k)=\begin{cases}\sqrt{2/N}, & \text{for}\ \ k=0\\ \sqrt{1/N}, & \text{for}\ \ k\neq 0\end{cases}$$

Fibonacci numbers was found by Leonardi Fibonacci in the middle ages. He wrote Fibonacci sequence in Liber Abaci as mathematics book. Fibonacci numbers has been used in many applications and also represents many natural phenomenas. The equation representing Fibonacci sequence is in the equation below (Fredric T. Howard, 2004) (Fallahpour & Megías, 2015) :

$$F(n)=\begin{cases}0, & \text{for } n<0\\ 1, & \text{for } n=1\\ F(n-1)+F(n-2), & \text{for } n>1\end{cases} \tag{5}$$

3. Watermarking Model

Watermark payload of proposed audio watermarking is N bits watermark data in which 1 bit is embedded per frame as displayed in figure 1. s(i) is host audio which is segmented and every bit of watermark d(m) is embedded in each frame of segmented host audio. Frame length and host audio type will affect the watermarking performance. Block diagram for audio watermarking process is displayed in figure 2. And watermark extraction is displayed in figure 3.

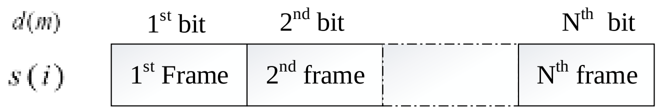

Fig 1: Watermark scheme for embedding

The embedding process is executed in frequency domain, which modifiesy the magnitude of host audio depending bit “0” or “1”. This embedding process is different from the previous research about multicarrier audio watermarking which there is no modification on host audio. The embedding is executed by adding the watermark after multicarrier modulation to the host audio (Budiman, Suksmono, & Shin, 2015). This basic technique of embedding is similar to the spread spectrum audio watermarking as Kirovski proposed in (Kirovski & Malvar, 2003).

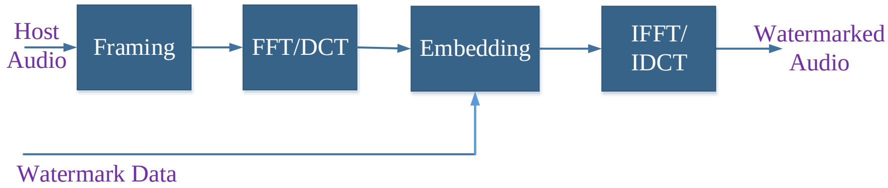

Fig 2: Watermark block diagram for embedding

Fig 3: Watermark block diagram for extraction

As displayed in figure 4a, the steps of embedding in this proposed audio watermarking technique is described below.

1. Host audio is selected. The duration of host audio is less than 1 minute. After embedding process the host audio will be modified becoming watermarked audio. The duration of watermarked audio will depend on the length of watermark.
2. Host audio is segmented in frame-based. Assume the frame length is *N* sample/frame. And segmented host audio is $s(i)$.
3. Each frame from segmented host audio is transformed by *N* point FFT or *N* point DCT. Output of FFT or DCT is assumed as $S(k)$. Both $s(i)$ and $S(k)$ are in discrete form. The first one is in time domain, and the second one is in frequency domain.
4. Select the frequency range for embedding. Considering HAS, the selected frequency is in high frequency. But for robustness performance, especially against low pass filter, the selected frequency will depend on host audio type.
5. Find magnitude coefficients value at Fibonacci sample number. This means to determine what is the value of $|S(1)|, |S(2)|, |S(3)|, |S(5)|, |S(8)|, \ldots\ldots$ started from the beginning of selected frequency. The maximum limit of Fibonacci number must less than *N*/2 for FFT. But for DCT the maximum limit of Fibonacci number must less than *N*.
6. Check the magnitude value between two adjacent Fibonacci sample number. If its value is between the magnitude of first and second Fibonacci sample magnitude, then execute embedding. If not, embedding is not executed. As example, if $|S(6)| > |S(5)|$ and $|S(6)| < |S(8)|$, then embedding is executed, else than that, it is not executed. The embedding will skip until the algorithm find the increase magnitude between 2 adjacent Fibonacci numbers.
7. Assume the modified magnitude for embedding is $|S(i)|$, the magnitude from previous Fibonacci number is $|S(n)|$, and the magnitude from next Fibonacci number is $|S(n+1)|$, then embedding equation is obtained using equation below (Fallahpour & Megías, 2015) (Fallahpour & Megías, 2014).

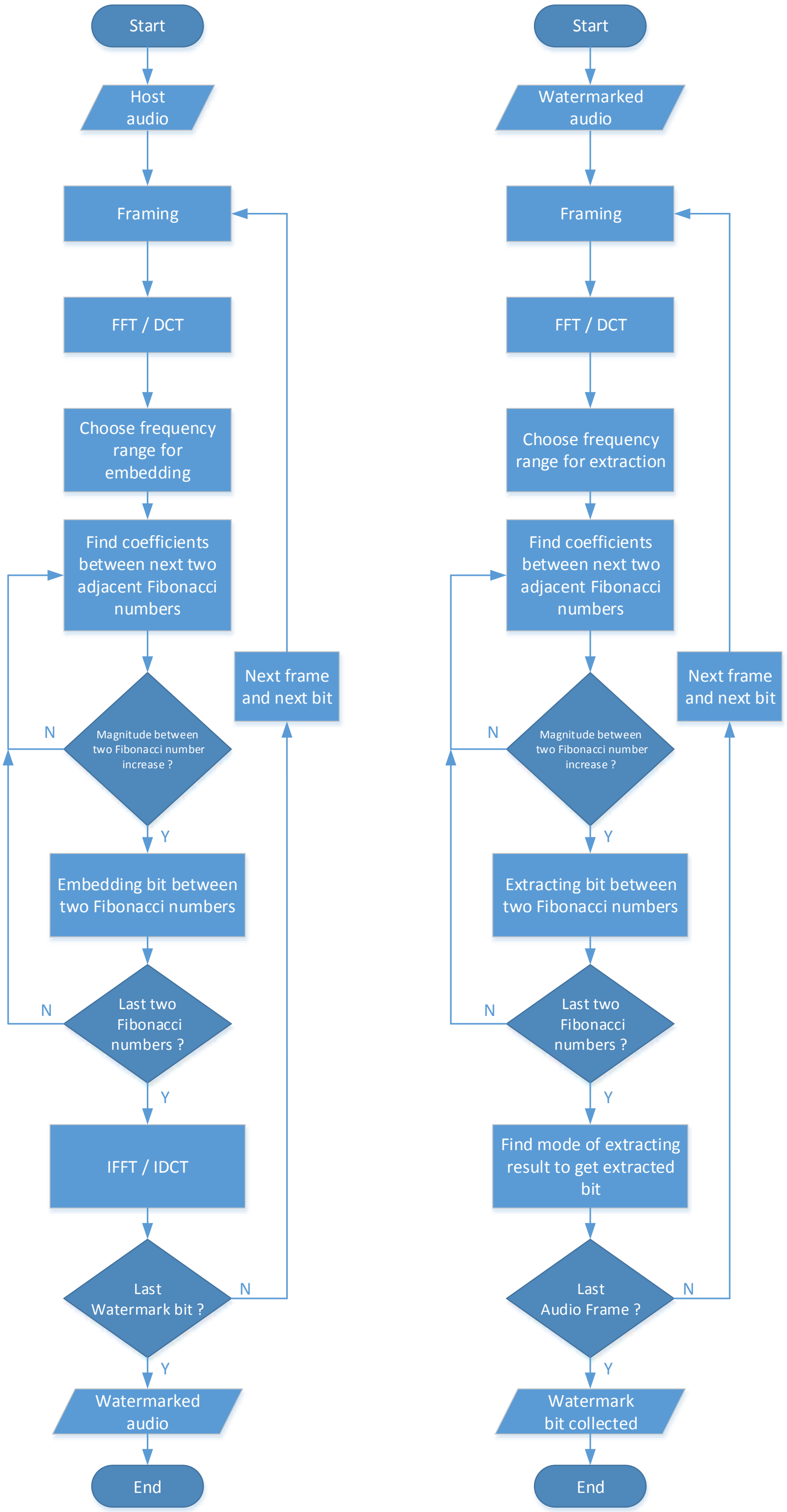

Fig 4a: Watermark embedding procedure, 4b Watermark extraction procedure

$$S(i) = \begin{cases} S(n), & \text{if } n \bmod 2 = d(m) \\ S(n+1), & \text{if } n \bmod 2 \neq d(m) \end{cases} \quad (5)$$

8. If Fibonacci number beyonds the last sample at current frame, then IFFT or IDCT is executed. If not, the process is back to step 5 – 7.

9. If previous watermark bit is the last bit for embedding then embedding is finished, watermarked audio will be generated, if not, step 2 – 8 is executed again.

Assume that the duration of watermarked audio is the same as watermarked bit number. If there are L watermark bit, then there will be minimum L frame or NL samples. Then the duration will be NL/F_s s. F_s is sample rate of watermarked or host audio. “Minimum” means that there is possibility to find frame that can not be embedded by watermark due to inexistence of increased magnitude between two adjacent Fibonacci number. The step to extract watermark bit is described below and displayed in figure 4b.

The steps to extract the watermark is described below :

1. Watermarked audio is selected.
2. Watermarked audio is segmented in frame-based. Assume the frame length is N sample/frame. And segmented watermarked audio is $\hat{s}(i)$.
3. Each frame from segmented host audio is transformed by N point FFT or N point DCT. Output of FFT or DCT is assumed as $\hat{S}(k)$.
4. Select the frequency range for extraction. The frequency range must be same as the frequency range in embedding side.
5. Find magnitude coefficients value at Fibonacci sample number.
6. Check the magnitude value between two adjacent Fibonacci sample number. If magnitude of next Fibonacci number is more than the previous one then do extraction in step 7. If not, go to next Fibonacci number, do step 5 and 6.
7. Extracting equation is obtained using equation below (Fallahpour & Megías, 2014) (Fallahpour & Megías, 2015).

$$d(m) = \begin{cases} 0, & \text{if } n \bmod 2 = 0 \\ 1, & \text{if } n \bmod 2 = 1 \end{cases} \tag{6}$$

8. If Fibonacci number beyonds the last sample at current frame, then collect the extracted bits and calculate mode of collected bits to get most frequent bits between “0” and “1”. If not, the process is back to step 5 – 8.
9. If the processed frame is last frame then the extracting process is finished and the extracted watermark from step 8 is collected, if not, go to next frame and do step 2 - 9.

4. Simulation Result

There are two important input parameters in proposed audio watermarking method, such as : frame length and bit depth of wav file as extension file saved before extraction stage. These two parameters affect the performance of audio watermarking, such as : imperceptibility (SNR and ODG), payload (watermark bit rate) and robustness (BER). To see how these parameters impact the performance, the simulation is applied at no attack condition. There are 13 audio files with frame rate 44,1 kHz, which are embedded each by 128 bits of watermark. The watermarked audio is saved to wav file by different bit depth, that is : 8, 16 and 32 bits/sample. The frame length is also changed : 32, 64, 128, 256, 512, 1024, and 2048 samples per frame. For DCT there is additional frame length for simulation, that is 16 sample per frame. As described in the previous section, the frame length is the same as N point FFT and DCT. There is no zero padding for transforming host audio to frequency domain. Since the embedding is not always applied in every frame, the rate of embedded watermark is difficult to estimate. Thus, the duration of watermarked audio depends on the characteristic of host audio.

Table 1 displays watermark imperceptibility as simulation result using FFT and DCT as transform method with vary frame length and bit depth. All simulation results in those tables are obtained by averaging the result of ODG and SNR from 13 testing audio. ODG has range -4 to 1 and SNR unit is in dB. The watermarked audio quality in fact depends on the host audio characteristics, thus the simulation result displays ODG and SNR with almost similar value for different frame length and bit depth. Comparing FFT and DCT, the ODG and SNR differences are also almost similar. But there is tendency that higher frame length will be lower SNR and ODG. This is because the possibility to find increased magnitude in Fibonacci number in higher frame length is higher than in lower frame length.

Table 1: FFT and DCT-based watermarked audio quality performance (SNR in dB)

Frame Length (sample/ frame)	Audio Quality											
	8 bit/sample				16 bit/sample				32 bit/sample			
	FFT		DCT		FFT		DCT		FFT		DCT	
	ODG	SNR	ODG	SNR	ODG	SNR	ODG	SNR	ODG	SNR	ODG	SNR
32	-0.5	40.4	-0.4	36.3	-0.2	46.1	-0.4	39.9	-0.2	45.5	-0.3	40.0
64	-0.5	38.5	-0.5	35.7	-0.3	45.5	-0.4	41.0	-0.3	45.7	-0.4	40.6
128	-0.4	35.7	-0.6	35.9	-0.2	40.4	-0.4	41.4	-0.2	41.1	-0.4	41.2
256	-0.6	37.3	-0.6	34.6	-0.2	43.5	-0.3	39.3	-0.2	44.0	-0.4	39.2
512	-0.5	38.9	-0.6	37.4	-0.2	46.0	-0.3	43.7	-0.2	46.5	-0.5	44.2
1024	-0.8	36.0	-0.7	36.5	-0.5	42.3	-0.5	43.3	-0.5	42.7	-0.6	43.2

Table 2 displays the watermark robustness for FFT and DCT respectively. BER (bit error rate) is displayed in % and comparing BER with bit depth 8, 16 and 32 for different frame length of FFT and DCT from 32 until 2048. But in table 4 the frame length is from 16 to 2048 due to DCT ability for embedding watermark eventhough the frame length is 16 sample per frame. As displayed in table 2, we can see that overall DCT robustness is better than FFT. FFT can only have perfect robustness or no error frame length above 512 samples per frame. But DCT has perfect robustness in all frame length at bit depth 32. This result shows that this proposed method is very sensitive to the bit depth of saved watermarked audio in wav file.

Table 2: FFT and DCT-based watermark robustness performance

Frame Length (sample/frame)	**BER (%)**					
	FFT			**DCT**		
	8	**16**	**32**	**8**	**16**	**32**
32	51.38	40.75	7.45	50.84	38.52	0
64	50.66	40.69	18.81	47.9	34.62	0
128	49.28	38.34	4.15	51.56	39.72	0
256	50.6	40.81	12.08	49.94	33.77	0
512	49.4	34.62	0	48.38	20.61	0
1024	47.36	28.06	0	45.73	12.38	0
2048	47.12	14.36	0	46.03	2.34	0

Table 3 displays the watermark payload for FFT and DCT respectively. Bit rate means the payload of watermark per second or bit per second. The payload is quantified not in integer because the embedding depends on the host audio characteristic, thus it is difficult to determine the payload of watermark by this proposed method. The payload is only able to calculate after simulation is done. Overall, the watermark payload for DCT as transform method is better in almost twice than FFT as transform method. As described in basic theory section, DCT has superiority than FFT in embedding capacity, that is twice than FFT embedding capacity. It can be seen also that bit depth doesn't affect the payload, but frame length does affect the payload. As the frame length increase, the payload will decrease. This payload is averaging result of 13 watermarked audio payload in different frame length and bit depth. Since the proposed audio watermarking is robust when DCT is used and the frame length can be from 16 to 2048 samples per frame, then the highest watermark payload can reach 500.27 bps. For next development, the above method should have robustness testing against signal processing attacks. It should be done to know how big the signal processing attacks affect the audio watermarking robustness.

Table 3: FFT and DCT-based watermark payload

Frame Length (sample/frame)	**Bit Rate (bps)**					
	FFT			DCT		
	8	**16**	**32**	**8**	**16**	**32**
32	191.49	191.49	191.49	500.27	500.27	500.27
64	195.3	195.3	195.3	386.43	386.43	386.43
128	171.1	171.1	171.1	284.05	284.05	284.05
256	120.44	120.44	120.44	160.17	160.17	160.17
512	70.6	70.6	70.6	85.66	85.66	85.66
1024	40.2	40.2	40.2	43.57	43.57	43.57
2048	21.28	21.28	21.28	21.92	21.92	21.92

5. Conclusion

The simulation result of frequency based audio watermarking comparing FFT and DCT transform method shows that watermark payload for perfect robustness at no attack condition could reach up to 70 bps for FFT and 500 bps for DCT. With good watermarked audio quality due to ODG > -1 and SNR > 30 dB, perfect robustness at no attack condition is obtained at only bit depth 32 for DCT and FFT, frame length more than 256 for FFT and all frame length scheme for DCT. Due to this performance, DCT is a recommended transform method for audio watermarking technique to obtain high imperceptibility, strong robustness and high capacity than FFT.

Acknowledgement

This research was partially supported by ITB WCU grant 2016 and RISET ITB 2016 and the publication was supported by Telkom University.

References

[1] Bender, W., Gruhl, D., Morimoto, N., & Lu, a. (1996). Techniques for data hiding. IBM Systems Journal, 35(3.4), 313–336. http://doi.org/10.1147/sj.353.0313

[2] Budiman, G., Suksmono, A. B., & Shin, D. H. (2015). A Multicarrier Modulation Audio Watermarking System. In International Conference on Electrical Engineering and Informatics 2015 (pp. 164–171). Legian, Bali: IEEE.

[3] Dhar, P. K., & Shimamura, T. (2015). Advances In Audio Watermarking Based on SVD. New York, London: Springer. http://doi.org/10.1007/9783319148007

[4] Dymarski, P., & Markiewicz, R. (2014). Robust Audio Watermarks in Frequency Domain. Journal of Telecommunications and Information Technology, 2, 1–10.

[5] Fallahpour, M., & Megías, D. (2014). Robust Audio watermarking based on Fibonacci numbers. 10th International Conference on Mobile Ad-Hoc and Sensor Networks Robust. http://doi.org/10.1109/MSN.2014.58

[6] 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

[7] Fredric T. Howard. (2004). Applications of Fibonacci Numbers. In F. T. Howard (Ed.), Proceedings of The Tenth International Research Conference on Fibonacci Numbers and Their Applications (pp. 1–328). North Carolina: Springer-Science+Business Media, B.V. http://doi.org/10.1007/s13398-014-0173-7.2

[8] Hayes, M. H. (1999). Schaum ' s Outline of Theory and Problems of Digital Signal Processing. McGraw-Hill.

[9] ITU-R, R. RECOMMENDATION ITU-R BS.1387-1 Method for objective measurements of perceived audio quality, BS.1387-1 International Telecommunications Union-Recommendation , Geneva (1998). Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Method+for+objective+measurements+of+perceived+audio+quality+1#0\nhttp://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:1387,+Method+for+objective+measurements+of+perceived+audio+qualit

[10] Kirovski, D., & Malvar, H. S. (2003). Spread-Spectrum Watermarking of Audio Signals. IEEE TRANSACTIONS ON SIGNAL PROCESSING, 51(4), 1020–1033.

[11] Lin, Y., & Abdulla, W. H. (2015). Audio Watermark, A Comprehensive Foundation Using Matlab. Springer. Auckland: Springer. http://doi.org/10.1007/978-3-319-07974-5_1

[12] Natgunanathan, I., Xiang, Y., Rong, Y., Member, S., Zhou, W., & Member, S. (2012). Robust Patchwork-Based Embedding and Decoding Scheme for Digital Audio Watermarking. IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, 20(8), 2232–2239. http://doi.org/10.1109/TASL.2012.2199111

[13] Wu, J.-L., & Shin, J. (1995). Discrete cosine transform in error control coding. IEEE Transactions on Communications, 43(5), 1857–1861. article. http://doi.org/10.1109/26.387423

[14] Xiang, Y., Member, S., Natgunanathan, I., Guo, S., & Member, S. (2014). Patchwork-Based Audio Watermarking Method Robust to Desynchronization Attacks. IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, 22(9), 1413–1423. http://doi.org/10.1109/TASLP.2014.2328175