

ANALISIS PERFORMANSI TEKNIK GABUNGAN *DISCRETE WAVELET TRANSFORM*, *DISCRETE SINE TRANSFORM*, DAN *QR DECOMPOSITION* PADA *AUDIO WATERMARKING STEREO* DENGAN *COMPRESSIVE SAMPLING*

PERFORMANCE ANALYZE FOR *DISCRETE WAVELET TRANSFORM*, *DISCRETE SINE TRANSFORM*, AND *QR DECOMPOSITION* IN *STEREO AUDIO WATERMARKING WITH COMPRESSIVE SAMPLING*

Syifa Maliah Rachmawati¹, Gelar Budiman, S.T., M.T.², Nur Ibrahim, S.T., M.T.³

^{1,2,3}Electrical Engineering Faculty of Telkom University, Bandung, Jawa Barat 40257

¹syifamaliah26@gmail.com, ²gelarbudiman@telkomuniversity.ac.id, ³nuribrahim.nib@gmail.com

ABSTRAK

Teknik *watermarking* perlu dikembangkan seiring kemajuan teknologi dan informasi. *Audio watermarking* memiliki kriteria utama meliputi *imperceptibility* dan *robustness*. Oleh karena itu, sangat penting mengetahui metode *watermarking* yang terbaik yang tahan terhadap berbagai serangan pemrosesan sinyal diiringi dengan kualitas audio yang baik ketika diperdengarkan kepada manusia. Penelitian ini merancang *audio watermarking* dengan gabungan metode *Discrete Wavelet Transformation*, *Discrete Sine Transformation*, dan *QR Decomposition*. *Compressive Sampling* digunakan untuk mengkompresi data *watermark* sehingga saat disisipkan kapasitas pada sinyal *host* meningkat. Teknik penyisipan yang digunakan adalah *Quantization Index Modulation*. Sistem *audio watermarking* yang dilakukan dalam penelitian ini menghasilkan kualitas audio dengan rata-rata SNR sebesar 26.96 dB, rata-rata BER 0.18, rata-rata ODG -1.1, dan MOS 4.2 terhadap seluruh *genre* audio yang diujikan.

Kata Kunci: *Discrete Wavelet Transformation*, *Discrete Sine Transformation*, *QR Decomposition*, *Compressive Sampling*, *Audio Watermarking*

ABSTRACT

Watermarking techniques need to be developed along with advances in technology and information. Audio watermarking has the main criteria of imperceptibility and robustness. Therefore, it is important to know the best watermarking method that is robust to various signal processing attacks accompanied by good audio quality when played to humans. This research designs audio watermarking with a combination of Discrete Wavelet Transformation, Discrete Sine Transformation, and QR Decomposition methods. Compressive sampling is used to compress the watermark data so that during the embedding process, the capacity of the host signal increases. The insertion technique used is Quantization Index Modulation. The audio watermarking system performed in this study produces audio quality with an average SNR of 26.96 dB, average BER 0.18, average ODG -1.13, and MOS 4.2 against all audio genres tested.

Keywords: *Discrete Wavelet Transform*, *Discrete Sine Transform*, *QR Decomposition*, *Compressive Sampling*, *Audio Watermarking*

1. Pendahuluan

Audio watermarking merupakan teknik penyisipan data digital pada sebuah sinyal audio dengan tujuan otentikasi data. Data digital yang disebut *watermark* disisipkan ke dalam *file* multimedia berupa audio, yang dapat digunakan untuk melacak dan memverifikasi keaslian dan hak cipta dari *file* yang bersangkutan. Sinyal audio yang menjadi tempat disisipinya *watermark* ini disebut dengan *host*.

Beberapa aspek yang harus dipenuhi dalam sebuah *audio watermarking* [1],[2] diantaranya : (i) *Imperceptibility* : *watermark* sebaiknya tidak mempengaruhi kualitas sinyal *host*. Hal ini dapat dievaluasi dengan penilaian objektif dengan mengukur *Signal to Noise Ratio* (SNR). (ii) *Robustness* : *watermark* harus tahan terhadap berbagai serangan yang disengaja atau tidak disengaja seperti noise aditif, filtering, *resampling*, *requantization*, *cropping*, kompresi MP3, dan sebagainya. (iii) *Security*: *watermark* harus tahan terhadap ekstraksi ilegal. (iv) *Capacity* : jumlah bit yang dapat tertanam harus lebih dari 20 bit per detik.

Dilaksanakannya penelitian ini merujuk pada beberapa referensi yang telah dilakukan dan menghasilkan audio yang *imperceptible* dan *robust* terhadap serangan pemrosesan sinyal. Penelitian [3] menggunakan metode *Discrete Wavelet Transformation* (DWT), penyisipan *watermark* dilakukan beberapa kali pada *level* dekomposisi berbeda. Penelitian [4] dengan memanfaatkan DWT yang dapat menganalisis multiresolusi dalam domain waktu dan domain frekuensi serta DCT yang memberi pemadatan pada energi sinyal sehingga memberi ruang untuk menyisipkan *watermark*. Penelitian [1] menggunakan metode DWT-DST yang menggunakan transformasi frekuensi ganda dipastikan lebih *imperceptible* dengan rata-rata SNR 27 dB serta tahan terhadap serangan daripada DWT-DCT. Untuk menambah kapasitas *watermark* pada sinyal *host*, dapat digunakan metode *QR decomposition*

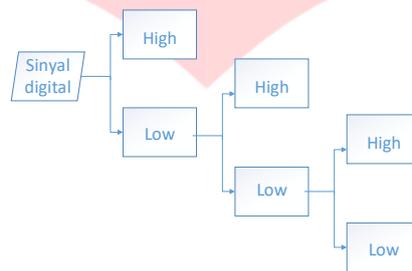
[5]. Metode lainnya adalah CS [6], dimanfaatkan dalam memulihkan sinyal yang bersifat *sparse* secara efisien dari data sampel yang terkompresi. Penelitian [7] mengusulkan metode penyisipan watermark dengan QIM, hasilnya menunjukkan kualitas audio dengan SNR di atas 20 dB dan kinerja yang baik terhadap berbagai serangan.

Penelitian ini mengembangkan sebuah metode baru yang merupakan kombinasi dari *Discrete Wavelet Transformation*, *Discrete Sine Transformation*, dan *QR decomposition* pada *audio watermarking* dengan *Compressive Sampling*. Teknik penyisipan yang digunakan adalah *Quantization Index Modulation*. Dipilihnya metode tersebut didasarkan pada kelebihan dan kekurangan dari tiap-tiap metode pada penelitian sebelumnya.

2. Dasar Teori

2.1 Discrete Wavelet Transformation

Dekomposisi dan rekonstruksi DWT melibatkan analisis multiresolusi. Selama proses dekomposisi, sinyal input didekomposisi menjadi dua bagian: bagian frekuensi tinggi (*detail coefficients*) dan bagian frekuensi rendah (*approximation coefficient*). Kemudian bagian frekuensi rendah didekomposisi kembali ke bagian frekuensi tinggi dan rendah. Proses ini diulang sampai tingkat dekomposisi yang diinginkan.



Gambar 1. Tiga level dekomposisi DWT

Satu tingkat dekomposisi secara matematis dapat dinyatakan sebagai berikut [5]:

$$Y_{high}[k] = \sum_n x[n] \cdot g[2k - n] \quad (1)$$

$$Y_{low}[k] = \sum_n x[n] \cdot g[2k - n]$$

$Y_{high}[k]$ dan $Y_{low}[k]$ masing-masing merupakan *output* dari *high-pass* dan *low-pass filter* setelah *sub-sampling*. Sinyal asli dikembalikan lagi semirip mungkin dengan sebelumnya menggunakan *inverse DWT*.

2.2 Discrete Sine Transformation

DST menjelaskan sebuah fungsi atau sinyal dari segi jumlah sinyal sinusoid dengan frekuensi dan amplituda yang berbeda [8]. Pada *audio watermarking*, DST digunakan untuk pemadatan energi. Fungsi DST dapat didefinisikan sebagai berikut [9]:

$$y(k) = \sum_{n=0}^{N-1} x(n) \sin \left[\frac{\pi kn}{N+1} \right], \quad k = 0, \dots, N-1 \quad (2)$$

Untuk mengembalikan sinyal diskrit ke domain waktu dari domain frekuensi, digunakan *inverse DST* atau disebut IDST (*Inverse Discrete Sine Transform*). Persamaan IDST dapat didefinisikan sebagai berikut:

$$x(n) = \frac{2}{N+1} \sum_{k=0}^{N-1} y(k) \sin \left[\frac{\pi kn}{N+1} \right], \quad n = 0, \dots, N-1 \quad (3)$$

2.3 QR Decomposition

Pada dasarnya dekomposisi QR digunakan untuk membagi matriks menjadi dua matriks, yaitu matriks orthogonal dan matriks segitiga. Misalkan A adalah matriks nyata $r \times r$. Matriks tersebut dapat di dekomposisi menggunakan QR seperti berikut [10]:

$$A = [a_1 | a_2 | \dots | a_n] \quad (4)$$

$$Q = [e_1 | e_2 | \dots | e_n]$$

$$R = \begin{bmatrix} a_1 \cdot e_1 & a_2 \cdot e_1 & a_3 \cdot e_1 \\ 0 & a_2 \cdot e_2 & a_3 \cdot e_2 \\ 0 & 0 & a_3 \cdot e_3 \end{bmatrix}$$

Dimana,

$$\begin{aligned} u_1 &= a_1, & e_1 &= \frac{u_1}{\|u_1\|} \\ u_2 &= a_2 - (a_2 \cdot e_1)e_1, & e_2 &= \frac{u_2}{\|u_2\|} \\ u_{k+1} &= a_{k+1} - (a_{k+1} \cdot e_1)e_1 - \dots - (a_{k+1} \cdot e_k)e_k \\ & & e_{k+1} &= \frac{u_{k+1}}{\|u_{k+1}\|} \end{aligned}$$

Q merupakan matriks ortogonal $r \times r$ ($Q^T \times Q = I$) dan R adalah $r \times r$ matriks segitiga atas. Untuk merekonstruksi menjadi matriks asli, digunakan inverse QR sebagai berikut:

$$A = Q \times R \quad (5)$$

2.4 Compressive Sampling

Compressive Sampling diterapkan untuk mencari solusi ketika suatu sinyal akan dibentuk dari cuplikan tak lengkapnya. Biasanya pada sebuah proses kompresi data sederhana dilakukan dengan cara memilih beberapa koefisien dominan, CS tidak perlu pencuplikan koefisien dominan karena bisa menggunakan koefisien mana saja asalkan batas minimum terpenuhi [11]. Memulihkan sinyal yang bersifat *sparse* dari data yang terkompresi merupakan bagian dalam CS. Karena ada banyak solusi yang tak terbatas, maka menemukan solusi yang terbaik dari beberapa pengukuran menjadi masalah dalam optimasi.

Secara matematis, sebuah sinyal dikatakan "*K-sparse*" jika $\|x\|_0 \leq K \ll n$, dimana $\|x\|_0$ merupakan *quasi-norm* yang dapat dihitung dengan $\|x\|_0 = \#\{j: x_j \neq 0\}$.

Pada CS, setiap pengukuran diperoleh dengan proyeksi sinyal aktual ke fungsi uji a_i [6]:

$$y_i = \langle x, a_i \rangle = a_i^T x, \quad 1 \leq i \leq m \ll n \quad (6)$$

atau jika ditulis dalam bentuk matriks menjadi:

$$y = Ax \quad (7)$$

dimana matriks penginderaan $A: R^n \rightarrow R^m$ mengurangi dimensi menjadi m baris, terdiri dari vektor $a_1^T, a_2^T, \dots, a_m^T$ dan rangkaian pengukuran pada vektor $y \in R^m$.

Peningkatan kinerja kompresi dapat terjadi karena dua hal: (1) tingkat kompresi tinggi karena koefisien yang diambil hanya sedikit, dan (2) sejumlah kecil koefisien cukup untuk merekonstruksi sinyal secara baik.

Proses rekonstruksi yang dilakukan memanfaatkan l_1 norm yang dapat didefinisikan sebagai berikut:

$$\hat{x} = \operatorname{argmin} \|x\|_1 \quad (8)$$

2.5 Quantization Index Modulation

Pada teknik penyisipan dengan QIM, tiap bit *watermark* $w(k)$ disisipkan pada koefisien matriks R_k yang dipilih dari setiap blok sesuai dengan persamaan berikut:

$$\begin{aligned} F'(0) &= \begin{cases} A_k, & w = 1 \text{ dan } \operatorname{argmin} |F(0) - A_k| \\ B_k, & w = 0 \text{ dan } \operatorname{argmin} |F(0) - B_k| \end{cases} \\ A_k &= \left(2k + \frac{1}{2}\right) \Delta, & B_k &= \left(2k - \frac{1}{2}\right) \Delta, & k &= 0, \pm 1, \pm 2, \dots \\ \Delta &= \frac{1}{2^{\text{jumlah bit}-1}} \end{aligned} \quad (9)$$

Dimana, Δ adalah selisih perubahan amplituda, $F(0)$ adalah host audio, $F'(0)$ merupakan audio ter-*watermark*, w adalah bit *watermark* yang disisipkan dan k merupakan indeks.

Proses ekstraksi dengan QIM dapat menggunakan persamaan berikut:

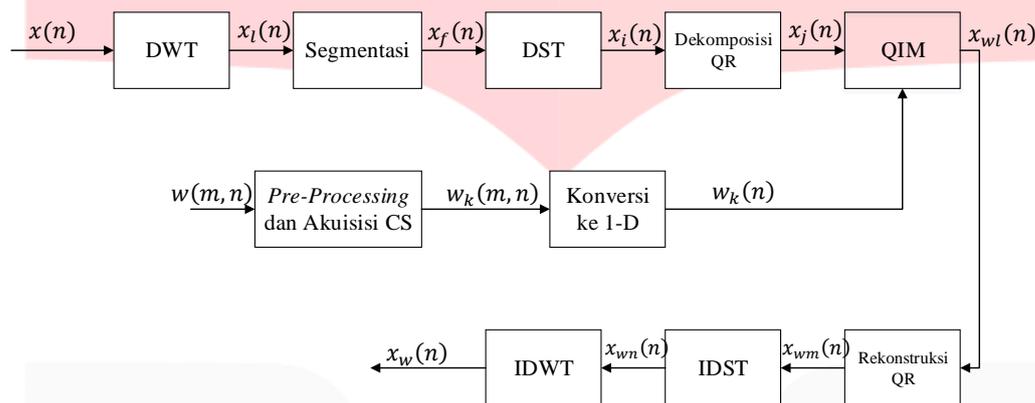
$$\hat{v}(k) = \operatorname{mod} \left(\operatorname{ceil} \left(\frac{F'(0)}{\Delta} \right), 2 \right) \quad (10)$$

Dimana, $\hat{v}(k)$ merupakan bit *watermark* hasil ekstraksi.

3. Perancangan Sistem

Perancangan sistem pada penelitian ini dibagi menjadi dua proses meliputi proses penyisipan dan proses ekstraksi. *Watermark* dikompresi kemudian disisipkan pada audio yang telah ditransformasi DWT-DST-dekomposisi QR. Setelah dilakukan proses penyisipan, dilakukan IQR-IDST-IDWT untuk mengembalikan sinyal seperti semula. Hasil dari proses tersebut adalah *watermarked audio*. *Watermarked audio* dilakukan uji kualitas dengan menghitung SNR, ODG, dan dilakukan survei MOS dengan membandingkan dengan audio asli. Untuk mengetahui ketahanan audio, diberikan beberapa serangan pemrosesan sinyal pada *watermarked audio*. Selanjutnya, dilakukan *watermark extracting* untuk dihitung nilai BER.

3.1 Proses Watermark



Gambar 2. Proses *watermark*

Langkah-langkah proses penyisipan sebuah informasi digital berupa gambar dapat dipaparkan sebagai berikut:

- Langkah 1 : Membaca data *watermark* $w(m, n)$.
- Langkah 2 : Lakukan *pre-processing* dan kompresi $w(m, n)$ dengan teknik *Compressive Sampling* pada persamaan 7 dan menghasilkan $w_k(m, n)$ sebagai keluaran proses ini.
- Langkah 3 : Konversi $w_k(m, n)$ menjadi matriks 1-D dan menghasilkan $w_k(n)$.
- Langkah 4 : Membaca *host audio* $x(n)$ lalu sesuaikan panjangnya sesuai dengan *watermark* (LL):

$$LL = 2^N \times \text{length } wt \times N\text{frame} \quad (11)$$

Dimana,

N = level dekomposisi dari DWT

$\text{length } wt$ = panjang *watermark*

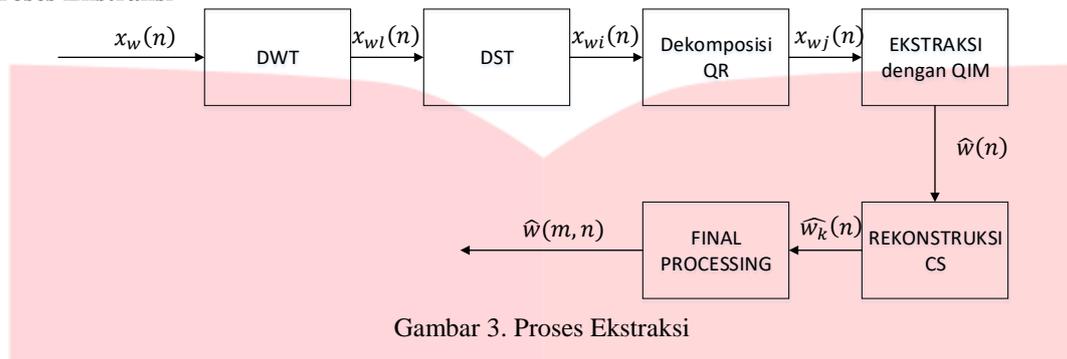
LL = panjang minimum *host audio*

- Langkah 5 : Dilakukan proses DWT pada sinyal *host* menggunakan persamaan 1 untuk mendapatkan komponen frekuensi tinggi dan rendah. Jumlah subband yang dihasilkan dalam proses ini adalah 2^N . Level dekomposisi yang digunakan adalah 4-level dekomposisi. Kemudian dipilih subband frekuensi rendah $x_l(n)$ sebagai masukan pada langkah berikutnya.
- Langkah 6 : Dilakukan proses segmentasi sehingga memberikan keluaran $x_f(n)$.
- Langkah 7 : $x_f(n)$ ditransformasi DST menggunakan persamaan 2. Keluaran dari proses ini adalah $x_i(n)$.
- Langkah 8 : Hasil transformasi DST disusun ulang menjadi matriks 2-D $r \times r$ yang diberi nama A_k , kemudian didekomposisi menjadi matriks orthogonal Q_k dan matriks segitiga atas R_k menggunakan *QR Decomposition*. Untuk mendapatkan matriks Q dan R gunakan persamaan 4.
- Langkah 9 : Tiap bit dari *watermark* $w_k(n)$ disisipkan pada sebuah koefisien yang telah dipilih dari matriks R_k dari tiap *frame* pada *host audio* sesuai prinsip QIM dengan persamaan 9 untuk menghasilkan $x_{wl}(n)$ sebagai keluaran.
- Langkah 10 : Rekonstruksi QR dilakukan pada hasil *embedding* untuk mendapatkan matriks A_k , yaitu:

$$A = Q \times R \quad (12)$$
 Keluaran dari proses ini dikonversi menjadi matriks 1-D dan dapat diasumsikan sebagai $x_{wm}(n)$.
- Langkah 11 : Untuk mengembalikan sinyal diskrit ke domain waktu dari domain frekuensi, diterapkan *inverse* DST yang dapat dihitung dengan persamaan 3. Keluaran dari proses ini adalah $x_{wn}(n)$.
- Langkah 12 : Merekonstruksi sinyal audio yang diberi *watermark* dengan menggunakan *inverse* DWT dengan menggabungkan dengan subband yang tidak digunakan.

Langkah 5 sampai 12 dilakukan dua kali karena menggunakan audio stereo. Didapatkan sinyal audio yang telah ter-*watermark* $x_w(n)$ dan dapat dihitung SNR dan ODG.

3.2 Proses Ekstraksi



Gambar 3. Proses Ekstraksi

Proses ekstraksi merupakan proses pengambilan *watermark* setelah diberi serangan pemrosesan sinyal untuk dihitung nilai BER. Adapun tahapan proses ekstraksi, diantaranya:

- Langkah 1 : Mendeteksi file audio yang telah terserang sebagai $\hat{x}_w(n)$.
- Langkah 2 : Transformasi menggunakan DWT untuk mendapatkan subband frekuensi rendah dan tinggi sebanyak 4-level dekomposisi. Pilih subband dengan frekuensi rendah $x_{wl}(n)$ untuk proses selanjutnya.
- Langkah 3 : $x_{wl}(n)$ dilakukan proses segmentasi kemudian ditransformasi menggunakan DST sesuai persamaan 2. keluaran dari proses ini diasumsikan sebagai $x_{wi}(n)$.
- Langkah 4 : Hasil transformasi DST disusun ulang menjadi matriks 2-D $r \times r$ yang diberi nama A_w , kemudian didekomposisi menjadi matriks orthogonal Q_w dan matriks segitiga atas R_w menggunakan *QR Decomposition*. Untuk mendapatkan matriks Q dan R dapat menggunakan persamaan 4.
- Langkah 5 : Tiap bit *watermark* diekstraksi dari tiap matriks R_k sesuai dengan posisi yang disimpan saat proses penyisipan. Proses ekstraksi ini dilakukan dengan metode QIM sebagai berikut:

$$\hat{v}(k) = \text{mod} \left(\text{ceil} \left(\frac{F'(0)}{\Delta} \right), 2 \right) \quad (13)$$

Keluaran dari proses ini diasumsikan sebagai $\hat{w}(n)$.

- Langkah 6 : Konversi menjadi matriks 2-D kemudian lakukan CS rekonstruksi menggunakan l_1 norm dengan persamaan 8 dan hasilnya diasumsikan sebagai $\hat{w}_k(n)$.
- Langkah 7 : Bit *watermark* yang diekstrak disusun ke bentuk matriks awal untuk mendapatkan *watermark* hasil ekstraksi $\hat{w}(m, n)$. Langkah 2 sampai 6 dilakukan dua kali karena audio yang digunakan adalah stereo. *Watermark* hasil ekstraksi nilainya di rata-ratakan kemudian dibulatkan.
- Langkah 8 : Hitung nilai BER.

4. Analisis

4.1 Analisis Kinerja Compressive Sampling terhadap Watermark

Pengujian dilakukan dengan pengubahan sisi *watermark* dan rasio kompresi bit untuk menghasilkan rasio kompresi dan *elapsed time*.



Gambar 4. Hasil ekstraksi sisi 8



Gambar 5. Hasil ekstraksi sisi 16



Gambar 6. Hasil ekstraksi sisi 32

Saat proses pengujian CS jika rasio kompresi bit semakin besar maka waktu yang dibutuhkan untuk melakukan sekali proses menjadi lebih lama. Penelitian ini menggunakan sisi 16 karena kualitas gambar cukup baik dan pada rasio kompresi bit yang sama dengan sisi 32 waktu yang dibutuhkan untuk sekali proses lebih cepat.

4.2 Analisis Parameter Awal

Penelitian ini menguji pengaruh masing-masing parameter terhadap nilai ODG, SNR, BER, dan C. Pengujian dilakukan pada audio host.wav dan menghasilkan parameter awal terbaik, sesuai Tabel 1.

Tabel 1. Parameter awal terbaik

| N | N Frame | nbit | thr | posisiw | typew | ODG | SNR | BER | C |
|---|---------|------|-----|---------|-------|--------|---------|-----|----------|
| 2 | 128 | 6 | 0.9 | 1 | 1 | 0.1279 | 50.1593 | 0 | 114.8438 |

4.3 Analisis Parameter Terhadap Serangan

Pada bagian ini akan dilakukan uji ketahanan parameter awal menggunakan 5 tipe audio dengan diberikan serangan pemrosesan sinyal, yaitu *Low Pass Filter*, *Band Pass Filter*, *noise*, *pitch shifting*, *resampling*, *equalizer*, *echo*, *Time Scale Modification*, dan kompresi MP3.

Tabel 2. Hasil pengujian parameter awal terhadap serangan

| Serangan | Parameter | BER | | | | |
|---------------------|-----------|----------|-----------|-----------|-----------|----------|
| | | Host.wav | Piano.wav | Gitar.wav | Drums.wav | Bass.wav |
| LPF | 3k | 0 | 0.2031 | 0.043 | 0.1016 | 0.0273 |
| | 6k | 0 | 0.0352 | 0 | 0.0664 | 0.0273 |
| | 9k | 0 | 0.0195 | 0 | 0.0664 | 0.0156 |
| BPF | 100-6k | 0.3906 | 0.4492 | 0.4297 | 0.2266 | 0.4023 |
| | 50-6k | 0.4688 | 0.4609 | 0.4609 | 0.2852 | 0.5195 |
| | 25-6k | 0.2344 | 0.3594 | 0.4453 | 0.2383 | 0.4102 |
| Noise | 0 dB | 0.4766 | 0.4063 | 0.4141 | 0.4766 | 0.4648 |
| | 10 dB | 0.4883 | 0.457 | 0.4531 | 0.4648 | 0.4414 |
| | 20 dB | 0.4922 | 0.4102 | 0.4063 | 0.418 | 0.4688 |
| Resampling | 22.05k | 0.0313 | 0.2539 | 0 | 0.0859 | 0.0313 |
| | 11.025k | 0.0078 | 0.207 | 0 | 0.0664 | 0.0273 |
| | 16k | 0 | 0.0234 | 0 | 0.0664 | 0 |
| TSM | 1% | 0.2773 | 0.3242 | 0.3008 | 0.2656 | 0.3242 |
| | 2% | 0.3555 | 0.3047 | 0.1289 | 0.3672 | 0.4023 |
| | 4% | 0.4023 | 0.418 | 0.3555 | 0.3906 | 0.3789 |
| Linear Speed Change | 1% | 0.0273 | 0.207 | 0.0352 | 0.043 | 0 |
| | 5% | 0.0273 | 0.1719 | 0.0352 | 0.043 | 0 |
| | 10% | 0.0234 | 0.1523 | 0.0352 | 0.0469 | 0 |
| Pitch Shifting | 1% | 0.4531 | 0.3984 | 0.4883 | 0.4063 | 0.418 |
| | 2% | 0.4336 | 0.4023 | 0.4297 | 0.4297 | 0.5 |
| | 4% | 0.4336 | 0.4492 | 0.4844 | 0.4336 | 0.4414 |
| Equalizer | | 0.418 | 0.4492 | 0.4297 | 0.1992 | 0.4102 |
| Echo | | 0.4219 | 0.4883 | 0.4453 | 0.4141 | 0.4609 |
| Kompresi MP3 | 32k | 0.4414 | 0.4102 | 0.4609 | 0.2578 | 0.4219 |
| | 64k | 0.4336 | 0.457 | 0.4414 | 0.1406 | 0.4688 |
| | 128k | 0.4141 | 0.4688 | 0.4492 | 0.1133 | 0.4219 |
| | 192k | 0.3867 | 0.4648 | 0.4336 | 0.1602 | 0.4258 |

Berdasarkan Tabel 2, parameter awal menghasilkan BER yang belum baik karena masih banyak didapatkan BER bernilai 0.4. Oleh karena itu, dilakukan optimasi untuk mendapatkan parameter terbaik yang menghasilkan nilai BER lebih kecil. Optimasi akan dilakukan pada audio piano.wav terhadap serangan kompresi MP3 32 kbps.

4.4 Optimasi Terhadap Serangan

Optimasi dilakukan untuk mendapatkan parameter terbaik dari metode ini agar memenuhi aspek *imperceptible* dan *robustness*. Optimasi dilakukan pada suatu sampel acak, dimana optimasi penelitian ini dilakukan pada audio piano.wav terhadap serangan Kompresi MP3 32 kbps. Hasil optimasi ditunjukkan pada Tabel 3.

Tabel 3 Parameter terbaik hasil optimasi

| N | N Frame | nbit | thr | posisiw | typew | ODG | SNR | BER | C |
|---|---------|------|-----|---------|-------|--------|---------|-----|---------|
| 4 | 32 | 1 | 0.9 | 0 | 1 | -1.195 | 32.1147 | 0 | 114.844 |

4.5 Ketahanan Terhadap Serangan

Dilakukan pengujian pada parameter terbaik hasil optimasi untuk menunjukkan ketahanan sistem *audio watermarking* tersebut terhadap serangan pada audio piano.wav. Hasil pengujian dapat dilihat pada Tabel 4.

Tabel 4. Nilai BER dan *watermark* hasil ekstraksi berdasarkan parameter hasil optimasi

| Type Serangan | Parameter | BER | Ekstraksi Watermark | Type Serangan | Parameter | BER | Ekstraksi Watermark |
|----------------|-----------|--------|---------------------|---------------------|-----------|-------|---------------------|
| LPF | 3k | 0 | | Noise | 0 dB | 0.418 | |
| | 6k | 0 | | | 10 dB | 0.390 | |
| | 9k | 0 | | | 20 dB | 0.382 | |
| Pitch Shifting | 1% | 0.3594 | | Linear Speed Change | 1% | 0 | |
| | 2% | 0.3281 | | | 5% | 0 | |
| | 4% | 0.3047 | | | 10% | 0 | |
| TSM | 1% | 0.2344 | | Echo | | 0.453 | |
| | 2% | 0.0352 | | Kompresi MP3 | 32k | 0 | |
| | 4% | 0.168 | | | 64k | 0 | |
| BPF | 100-6k | 0.3828 | | | 128k | 0 | |
| | 50-6k | 0.4102 | | 192k | 0 | | |
| | 25-6k | 0.3047 | | Resampling | 22.05k | 0 | |
| Equalizer | 0.140 | | | | 11.025k | 0 | |
| | | | | | 16k | 0 | |

4.6 Kualitas Audio Hasil Optimasi

Kualitas audio merupakan hal penting dalam sebuah sistem *audio watermarking*. Data *watermarking* tidak boleh terdengar oleh indera pendengaran manusia. Parameter yang mendasari adalah nilai ODG. Parameter lain dalam penilaian kualitas audio adalah SNR dengan membandingkan sinyal asli dengan sinyal *terwatermark*. Berikut merupakan hasil uji dari kualitas *watermarked audio* berdasarkan parameter hasil optimasi.

Tabel 5. Uji kualitas berdasarkan parameter hasil optimasi

| Host | SNR | | ODG | |
|-----------|-----------------------|----------------------|-----------------------|----------------------|
| | Dengan CS, Param-2 | Tanpa CS, Param-2 | Dengan CS, Param-2 | Tanpa CS, Param-2 |
| Host.wav | 20.9104 | 16.4045 | -0.0783 | -0.0982 |
| Piano.wav | 32.091 | 32.8808 | -1.6161 | -2.1167 |
| Gitar.wav | 18.7149 | 17.2563 | -0.2791 | -0.3199 |
| Drums.wav | 25.5572 | 28.4526 | -3.7993 | -3.5222 |
| Bass.wav | 37.5604 | 38.0392 | 0.0773 | 0.0905 |
| Rata-rata | 26.96678 | 26.60668 | -1.1391 | -1.1933 |

Tabel 5 menunjukkan kualitas audio dengan dan tanpa *Compressive Sampling* berdasarkan parameter terbaik. Hasilnya menunjukkan bahwa penggunaan metode CS lebih baik meskipun pada nilai rata-rata tidak terjadi perubahan yang signifikan dalam kualitas. Namun, jika ditinjau satu persatu dari setiap audio, penggunaan metode CS yang menghasilkan audio dengan nilai SNR di bawah standar IFPI hanya pada 1 audio, yaitu gitar. Sementara tanpa penggunaan metode CS terdapat 2 audio yang menghasilkan nilai SNR di bawah standar, yaitu gitar dan host. Meski demikian keduanya tetap menghasilkan nilai ODG di atas -1 yang artinya *watermark* terasa sedikit tapi tidak mengganggu. Sehingga dapat disimpulkan bahwa penggunaan metode *Compressive Sampling* lebih baik.

4.6 Mean Opinion Score (MOS)

MOS digunakan untuk evaluasi *audio imperceptibility* secara subyektif. Setiap pendengar dilengkapi dengan sinyal audio asli dan ter-*watermark* dan diminta untuk melaporkan degradasi antara kedua sinyal dengan menggunakan nilai subyektif. Penilaian dilakukan kepada 30 orang responden. Hasil rata-rata nilai MOS pada audio dengan parameter B kepada 30 responden adalah sebagai berikut:

Tabel 6. Rata-rata MOS

| Jenis Audio | Rata-rata MOS |
|-------------|---------------|
| Host.wav | 4.3 |
| Piano.wav | 4.267 |
| Gitar.wav | 4.167 |
| Drums.wav | 4.23 |
| Bass.wav | 4.067 |
| Rata-rata | 4.2062 |

Nilai MOS berkisar antara 1 dimana *watermark* sangat mengganggu dan audio tidak terdengar sampai 5 dimana *watermark* tidak terdengar. Dilihat dari Tabel 4.14 seluruh audio memiliki nilai rata-rata MOS di atas 4. Hal tersebut menunjukkan kualitas audio yang dihasilkan meski *watermark* sedikit terasa tetapi tidak mengganggu.

5. Kesimpulan

Implementasi sistem *audio watermarking* pada *file* audio dengan format .wav menggunakan metode yang diusulkan menghasilkan rata-rata SNR 26.96 dB, BER 0.18, ODG -1.13, dan MOS 4.2 terhadap seluruh *genre* audio yang diujikan. Metode tersebut tahan terhadap serangan *low pass filter*, *resampling*, dan *linear speed change*. Nilai SNR terbaik dihasilkan oleh audio bass.wav dengan SNR 37.56 dB dan ODG 0.07.

6. Referensi

- [1] M. Hemis and B. Boudraa, "Digital watermarking in audio for copyright protection," *Proc. - ICACISIS 2014 2014 Int. Conf. Adv. Comput. Sci. Inf. Syst.*, pp. 189–193, 2014.
- [2] J. Bajpai and A. Kaur, "A Literature Survey – Various Audio Watermarking Techniques and their challenges," *2016 6th Int. Conf. - Cloud Syst. Big Data Eng.*, pp. 451–457, 2016.
- [3] K. Datta and I. Sengupta, "A Redundant Audio Watermarking Technique Using Discrete Wavelet Transformation," *Commun. Softw. Networks, 2010. ICCSN '10. Second Int. Conf.*, pp. 0–4, 2010.
- [4] S. M. Deokar and B. Dhaigude, "Blind Audio Watermarking Based On Discrete Wavelet and Cosine Transform," *2015 Int. Conf. Ind. Instrum. Control*, pp. 264–268, 2015.

- [5] A. Kaur, M. K. Dutta, K. M. Soni, and N. Taneja, "A High Payload Audio Watermarking Algorithm Robust against Mp3 Compression," *2014 Seventh Int. Conf. Contemp. Comput.*, pp. 531–535, 2014.
- [6] H. Haider, J. A. Shah, and U. Ali, "Comparative analysis of sparse signal recovery algorithms based on minimization norms," *2014 World Congr. Sustain. Technol. WCST 2014*, pp. 72–76, 2015.
- [7] J. Wang, M. Lai, K. Liang, and P. Chang, "Adaptive Wavelet Quantization Index Modulation Technique for Audio Watermarking," *Int. Comput. Symp.*, no. 1, 2006.
- [8] B. N. Madhukar and S. Jain, "A Duality Theorem for the Discrete Sine Transform (DST)," *Proc. 2015 Int. Conf. Appl. Theor. Comput. Commun. Technol. iCATccT 2015*, pp. 156–160, 2016.
- [9] H. Yassine, B. Bachir, and K. Aziz, "A Secure and High Robust Audio Watermarking System for Copyright Protection," *Int. J. Comput. Appl.*, vol. 53, no. 17, pp. 33–39, 2012.
- [10] M. Hemis, B. Boudraa, and T. Merazi-Meksen, "Optimized audio watermarking scheme with swarm intelligence," *2015 First Int. Conf. New Technol. Inf. Commun.*, pp. 1–6, 2015.
- [11] A. B. Suksmono, *Memahami Penginderaan Kompresif dengan MATLAB*. 2008.