

Audio Watermarking Stereo Dengan Sinkronisasi Berbasis Hybrid SWT-FFT-SVD Dan QIM

Rifdah Fausiah Ashari¹, Gelar Budiman², R. Yunendah Nur Fuadah³

^{1,2,3}Fakultas Teknik Elektro, Universitas Telkom, Bandung, Jawa Barat 40257

rfausiah@gmail.com¹, gelarbudiman@telkomuniversity.ac.id², yunendah@gmail.com³

Abstrak – *Audio watermarking stereo merupakan suatu metode penyisipan data rahasia (watermark) ke dalam file audio stereo yang terdiri dari 2 channel. Proses watermarking akan dilakukan secara dual method yakni pada host audio dan pada data watermark. Pada makalah ini, akan dirancang suatu sistem audio watermarking stereo menggunakan metode SWT-FFT-SVD dan QIM dengan sinkronisasi. Pertama, sinyal host audio didekomposisi kedalam beberapa sub-band menggunakan metode SWT. Keluaran dari waktu ke domain frekuensi. Selanjutnya, sub-band yang telah dipilih didekomposisi kedalam bentuk matriks menggunakan metode SVD. Penambahan bit-bit sinkronisasi pada data watermark dilakukan sebelum proses penyisipan (embedding). Proses penyisipan data watermark ke dalam host audio dilakukan menggunakan metode QIM. Hasil yang diharapkan dalam makalah ini didapatkan tingkat imperceptibility, ketahanan sinyal (robustness) terhadap berbagai serangan, dan kapasitas payload watermark yang baik.*

Kata kunci: *Audio Watermarking stereo, Stationary Wavelet Transform (SWT), Fast Fourier Transform (FFT), Singular Value Decomposition (SVD), Sinkronisasi, Quantization Index Modulation (QIM)*

1. Pendahuluan

Perkembangan teknologi informasi digital dan jaringan internet telah memberikan kemudahan dan kebebasan kepada para penggunanya, khususnya dalam mempublikasikan karya yang diciptakan, yakni musik, lagu, dan file audio lainnya. Berkembangnya teknologi dan internet tersebut ternyata berbanding lurus dengan dampak negatif yang ditimbulkan. Banyaknya kasus pelanggaran hak cipta yang terjadi telah mencerminkan bahwa masih lemahnya sistem perlindungan atas karya intelektual seseorang. Oleh karena itu, diperlukan suatu metode yang dapat digunakan untuk melindungi hak cipta dari pemilik karya tersebut. Salah satunya adalah dengan menggunakan *audio watermarking*.

Audio watermarking merupakan suatu metode penyisipan data rahasia (*watermark*) ke dalam suatu host/informasi digital audio. Proses penyisipan data *watermark* dilakukan dengan berbagai cara agar tidak merusak file audio serta tidak dapat dirasakan oleh indera manusia. Umumnya, skema *audio watermarking* yang ideal harus memenuhi persyaratan berikut [1][2][3][4]:

1. Imperceptibility

Data *watermark* yang disisipkan tidak boleh diketahui oleh indera manusia dan tidak boleh merusak host audio. Kualitas audio yang diberi watermark dapat diukur secara subjektif atau objektif.

2. Robustness

Jika terdapat orang yang tidak bertanggung jawab ingin menghancurkan data *watermark*, maka sistem *audio watermarking* harus mampu menahan berbagai serangan seperti *noise*, filtering, dll.

3. Kapasitas / Payload

Jumlah data *watermark* yang dapat disisipkan ke dalam host audio. Sebisa mungkin, setelah proses ekstraksi, ukuran data *watermark* mendekati ukuran data *watermark* asli.

Dalam beberapa tahun terakhir, beberapa peneliti telah mengembangkan skema sistem *audio watermarking* sesuai dengan perkembangan teknologi, diantaranya penggabungan beberapa metode [8], modulasi *multicarrier* [25], dll. Pada penelitian [5] mengusulkan sebuah algoritma untuk mengekstrak segmen audio untuk *watermarking* menggunakan SWT dan *spread spectrum*. Hasil eksperimen dari metode ini cukup kuat dan bisa bertahan melawan beberapa serangan, namun hanya bisa menahan hingga 50%. Pada penelitian [6] menemukan bahwa jika pita frekuensi diperbesar, kapasitas dan distorsi akan meningkat dan sebaliknya ketahanan sinyal akan turun. Sedangkan, jika ukuran *frame* ditingkatkan, ketahanan akan meningkat dan kapasitas akan menurun. Hasil simulasi menggunakan metode FFT berdasarkan angka Fibonacci adalah *average error* untuk setiap sampel adalah 25%, dan memiliki kapasitas tinggi. Sedangkan pada penelitian [24] data *watermark* akan disisipkan kedalam *file* audio menggunakan transformasi dalam domain frekuensi yaitu FFT dan DCT. Pada metode DCT, *watermark* dapat disisipkan pada semua *frame*, sedangkan pada FFT, *watermark* dapat disisipkan hanya pada setengah *frame* saja. Hasil dari perbandingan metode ini didapatkan nilai ODG > -1 and SNR > 30 dB dan *watermark payload* 70 bps pada FFT dan 500 bps pada DCT saat kondisi audio belum diberi serangan. Pada penelitian selanjutnya menunjukkan perbandingan antara beberapa metode. Metode yang menggunakan teknik SVD memiliki nilai BER dan Corr yang lebih baik dibandingkan dengan metode tanpa SVD. Penelitian ini juga menunjukkan bahwa metode *hybrid* memiliki ketahanan yang lebih baik terhadap beberapa serangan gabungan [7]. Pada [8] mengusulkan skema *audio watermarking* berdasarkan metode hibrida (DCT-SVD, DWT-SVD, DWT-DCT-SVD, LWT-DCT-SVD). Penelitian ini menunjukkan bahwa sinyal audio asli akan terpengaruh jika nilai tingkat kuantisasi meningkat yaitu pada makalah ini menggunakan nilai kuantisasi antara 0,01 sampai 1,0. Pada penelitian [9] menyatakan bahwa sebagian besar skema dengan metode QIM memiliki tingkat ketahanan yang kurang baik. Metode yang mereka usulkan yakni penggabungan SVD-QIM memiliki kinerja yang kurang baik jika diberi serangan *resampling* AWGN 16KHz.

Dalam penelitian ini, kami mengusulkan transformasi gabungan SWT-FFT-SVD dengan sinkronisasi dan QIM. Transformasi *wavelet* [10] [11][12] adalah salah satu teknik transformasi yang dapat digunakan untuk mengurangi noise, memiliki kemampuan multi-resolusi yang baik dan dapat memberikan representasi sinyal audio yang lebih baik dalam domain frekuensi-waktu. Saat ini, untuk teknik transformasi, DWT dan LWT merupakan metode transformasi yang paling banyak digunakan. Metode transformasi yang digunakan pada penelitian ini untuk implementasi sistem *audio watermarking* adalah SWT. *Stationary Wavelet Transform* (SWT) adalah algoritma transformasi *wavelet* yang dirancang untuk mengatasi *time-variant* yaitu nilai input akan berbeda dengan nilai output karena adanya pergeseran pada sinyal sehingga dapat memberikan variasi sinyal yang cukup besar pada output pada metode DWT. SWT adalah tahap pertama pada proses *watermarking* dengan men-dekomposisi sinyal file audio dan *watermark* menjadi beberapa sub-band. Dalam domain frekuensi, metode FFT diterapkan untuk mengubah sinyal dalam domain waktu kedalam sinyal domain frekuensi. Seperti yang kita bandingkan dengan domain waktu, domain frekuensi lebih kompleks dan kuat terhadap berbagai serangan [13]. SVD diterapkan untuk dekomposisi sinyal kedalam tiga buah matriks. Selanjutnya, penambahan bit-bit sinkronisasi diperlukan untuk mencari posisi awal data *watermark* saat proses ekstraksi. Pada [14] [15] [16], bit sinkronisasi disisipkan ke dalam domain waktu dan memiliki hasil pendeteksian yang lebih cepat, sementara jika disisipkan ke dalam domain frekuensi dapat mencapai ketahanan yang tinggi. QIM adalah salah satu metode kuantisasi untuk menyisipkan informasi. QIM melakukan proses pemetaan nilai amplitudo sampel sinyal dengan membulatkan ke bilangan terdekat (0 dan 1) [17] atau memasukkan bit *watermark* ke sinyal host sehingga sinyal host mengalami perubahan amplitudo. QIM merupakan metode yang sangat sederhana dan dapat memberikan ketahanan yang baik, kapasitas, dan rate-distortion [17] [18].

Makalah ini diuraikan dengan susunan sebagai berikut: Bab 1 menjelaskan pendahuluan dan latar belakang penelitian, pada bab 2 mendeskripsikan metode yang digunakan dalam skema

audio watermarking, pemodelan sistem, dan serangan yang digunakan, bab 3 menjelaskan hasil dan analisis performansi dari skema sistem audio watermarking, dan kesimpulan penelitian pada bab 4.

2. Metode Penelitian

2.1. Dasar Teori

2.1.1. Stationary Wavelet Transform

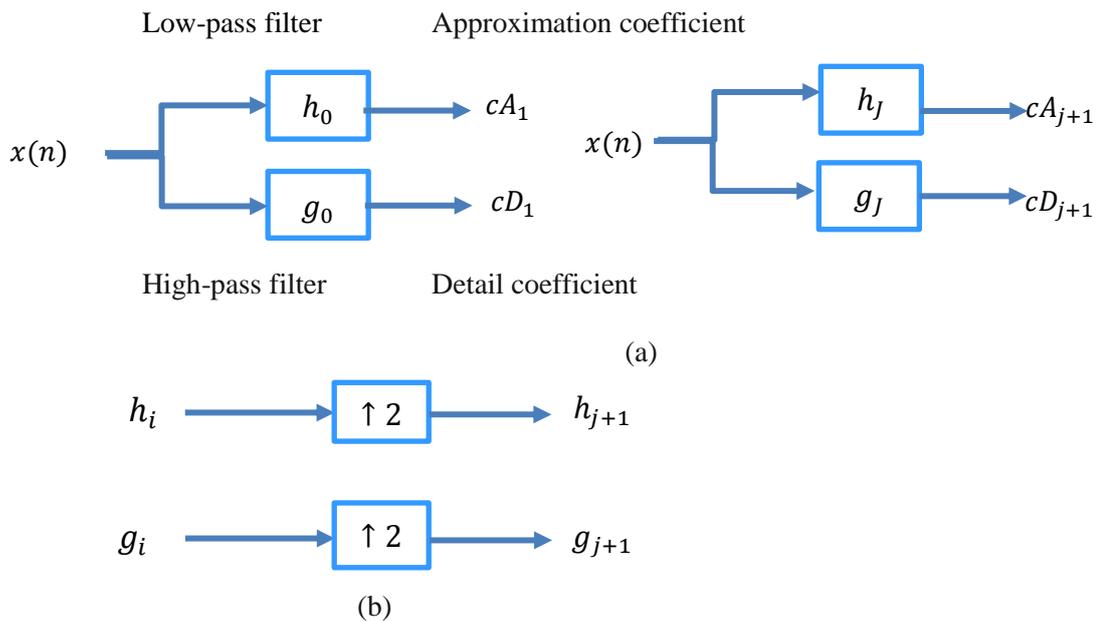
Stationary Wavelet Transform (SWT) adalah salah satu jenis algoritma transformasi wavelet yang menyerupai algoritma DWT. Algoritma SWT merupakan *undecimated transform* yang menggunakan skema *time-invariant* dan redundansi. Output dari masing-masing level memiliki panjang yang sama dengan input sehingga tidak menggunakan *downsampling* pada tahap dekomposisi [10].

Dekomposisi SWT membagi sinyal menjadi 2 koefisien yaitu *approximate* dan *detail coefficients* dengan menggunakan LPF untuk *approximate* dan HPF untuk *detail*, seperti yang ditunjukkan pada Gambar.1. Persamaan untuk mendapatkan *approximate coefficient* SWT dapat dihitung dengan menggunakan rumus [5]:

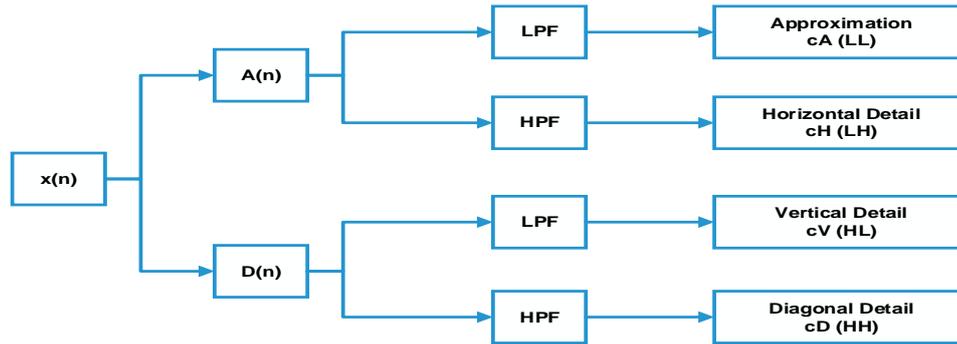
$$A_{j+1,k} = (\int(x), \frac{1}{2^{\frac{j+1}{2}}} \varphi(\frac{x-k}{2^{j+1}})) = \sum_{l=-\infty}^{+\infty} h(l)A_{j,k+2^j l} \tag{2.1}$$

Dan untuk *detail coefficient* SWT didapatkan dengan menggunakan rumus[5] :

$$D_{j+1,k} = \sum_{l=-\infty}^{+\infty} h(l)D_{j,k+2^j l} \tag{2.2}$$



Gambar 1. Dekomposisi 1-level SWT (a) Proses dekomposisi (b) Komputasi filter [5]



Gambar 2. Dekomposisi 2-level SWT

Seperti ditunjukkan pada Gambar 2, proses dekomposisi dilakukan menggunakan dekomposisi 2-level SWT. File audio asli dibagi menjadi $A(n)$ dan $D(n)$. $A(n)$ adalah komponen dengan frekuensi rendah dan $D(n)$ adalah komponen frekuensi tinggi. Rumus untuk dekomposisi 2-level SWT adalah sebagai berikut [20]:

$$\begin{aligned}
 A_{j,k_1,k_2} &= \sum_{n_1} \sum_{n_2} h0^{\uparrow 2j}(n_1 - 2k_1) h0^{\uparrow 2j}(n_2 - 2k_2) A_{j-1,n_1,n_2} \\
 D^1_{j,k_1,k_2} &= \sum_{n_1} \sum_{n_2} h0^{\uparrow 2j}(n_1 - 2k_1) g0^{\uparrow 2j}(n_2 - 2k_2) A_{j-1,n_1,n_2} \\
 D^2_{j,k_1,k_2} &= \sum_{n_1} \sum_{n_2} g0^{\uparrow 2j}(n_1 - 2k_1) h0^{\uparrow 2j}(n_2 - 2k_2) A_{j-1,n_1,n_2} \\
 D^3_{j,k_1,k_2} &= \sum_{n_1} \sum_{n_2} g0^{\uparrow 2j}(n_1 - 2k_1) g0^{\uparrow 2j}(n_2 - 2k_2) A_{j-1,n_1,n_2} \quad (2.3)
 \end{aligned}$$

Dimana $A_{j,k_1,k_2}, D^1_{j,k_1,k_2}, D^2_{j,k_1,k_2}, D^3_{j,k_1,k_2}$ adalah komponen (LL), komponen (LH), komponen (HL) dan komponen (HH). Sedangkan algoritma rekonstruksi (IDSWT) diperoleh dengan rumus [20]:

$$\begin{aligned}
 A_{j-1,n_1,n_2} &= \frac{1}{4} \sum_{i=0}^3 \left\{ \sum_{k_1} \sum_{k_2} h_1(n_1 - 2k_1 - i) h_1(n_2 - 2k_2 - i) A_{j,k_1,k_2} \right. \\
 &\quad + \sum_{k_1} \sum_{k_2} h_1(n_1 - 2k_1 - i) g_1(n_2 - 2k_2 - i) D^1_{j,k_1,k_2} \\
 &\quad + \sum_{k_1} \sum_{k_2} g_1(n_1 - 2k_1 - i) h_1(n_2 - 2k_2 - i) D^2_{j,k_1,k_2} \\
 &\quad \left. + \sum_{k_1} \sum_{k_2} g_1(n_1 - 2k_1 - i) g_1(n_2 - 2k_2 - i) D^3_{j,k_1,k_2} \right\} \quad (2.4)
 \end{aligned}$$

2.1.2. Fast Fourier Transform

Fast Fourier Transform merupakan salah satu transformasi fourier yang mengubah dari domain waktu ke domain frekuensi. FFT merupakan suatu algoritma perhitungan dari *Discrete Fourier Transform* (DFT) yang memiliki durasi komputasi yang lebih cepat. Transformasi Fourier Diskrit (DFT) didefinisikan menggunakan formula [23]:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{-j2\pi kn/N} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn} = \quad n = 0, 1, \dots, N - 1 \quad (2.5)$$

$$W_N = e^{-j2\pi/N}$$

$$(2.6)$$

Invers dari DFT dapat dilihat dari persamaan [21]:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{-j2\pi kn/N} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn} \quad n = 0, 1, \dots, N-1 \quad (2.7)$$

Lalu, FFT didefinisikan menggunakan persamaan [23]:

$$X(k) = \sum_{n=even} x(n) W_N^{nk} + \sum_{n=odd} x(n) W_N^{nk} \quad (2.8)$$

$$X(k) = \sum_{l=0}^{\frac{N}{2}-1} x(2l) W_N^{2lk} + \sum_{l=0}^{\frac{N}{2}-1} x(2l+1) W_N^{(2l+1)k} \quad (2.9)$$

Dimana $X(k)$ adalah sinyal FFT, $x(n)$ adalah sinyal audio asli dalam domain waktu, dan N adalah panjang sinyal.

2.1.3. Singular Value Decomposition

Singular Value Decomposition (SVD) adalah salah satu analisis numerik yang menggunakan matriks untuk menguraikan sinyal atau masukan menjadi tiga matriks sebagai berikut [22]:

$$A = U * S * V^T, \quad (2.10)$$

dimana A adalah input dan akan didekomposisi menjadi vektor singular kiri dan kanan (matriks U, V) dan S adalah matriks diagonal yang memiliki nilai eigen dari input sebagai elemen diagonal [8]. Matriks U dan V adalah matriks $M \times M$ dan $N \times N$ yang kolomnya mengandung vektor eigen yang dinormalisasi yang diturunkan dari matriks AA^T dan $A^T A$ sedangkan S adalah matriks diagonal berukuran $M \times N$ yang pada bagian diagonalnya mengandung nilai singular (σ) dari matriks X , di mana $\sigma = \lambda$ dan λ adalah nilai eigen dari matriks AA^T dan $A^T A$.

2.1.4. Quantization Index Modulation

Kuantisasi adalah proses pemetaan nilai amplitudo pada sampel sinyal dengan melakukan pembulatan ke bilangan terdekat (0 dan 1) atau dengan kata lain kuantisasi merupakan proses penyisipan bit *watermark* ke dalam sinyal host sedemikian rupa sehingga sinyal host mengalami perubahan amplitudo. Dalam metode QIM, terdapat parameter Δ yang merupakan selisih perubahan amplitudo setelah dikuantisasi. Nilai Δ dipengaruhi oleh jumlah bit kuantisasi yang digunakan.

Persamaan metode QIM untuk proses penyisipan dan proses ekstraksi adalah sebagai berikut:

1. Proses penyisipan:

$$F'(0) = \begin{cases} A_k, & \text{jika } w = 1 \text{ dan } \arg \min |F(0) - A_k| \\ B_k, & \text{jika } w = 0 \text{ dan } \arg \min |F(0) - B_k| \end{cases} \quad (2.13)$$

$$\Delta = \frac{1}{2^{n-1}}, A_k = \left(2k + \frac{1}{2}\right) \Delta, B_k = \left(2k - \frac{1}{2}\right) \Delta, k = 0, \pm 1, \pm 2, \dots \quad (2.14)$$

2. Proses ekstraksi:

$$v(k) = \text{mod} \left[\left(\frac{F'(0)}{\Delta} \right), 2 \right] \quad (2.15)$$

Keterangan:

$F(0)$ = *sample host* sebelum dikuantisasi

$F'(0)$ = *sample host* setelah disisipkan

Δ = delta

k = indeks kuantisasi

n = jumlah bit kuantisasi

2.1.4. Sinkronisasi

Bit sinkronisasi digunakan untuk menemukan posisi *watermark* pada proses ekstraksi. Bit sinkronisasi dapat disisipkan ke domain waktu atau domain frekuensi. Keuntungan jika bit sinkronisasi disisipkan kedalam domain waktu adalah biaya rendah dalam mencari kode sinkronisasi tetapi memiliki ketahanan yang rendah. Ketika bit sinkronisasi dimasukkan ke dalam domain frekuensi, ketahanan akan bagus namun biaya pencarian akan besar secara bersamaan [14].

Dalam makalah ini, algoritma *PN sequence* digunakan untuk bit sinkronisasi karena memiliki autokorelasi yang tinggi [26]. *Pseudorandom Noise code* (PN code) merupakan suatu kode berupa bit *random* yang bersifat periodik yang ditunjukkan dengan [19] :

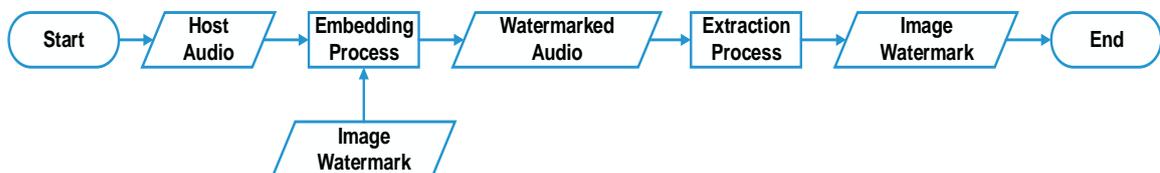
$$p(n) = p(0), p(1), p(2), \dots, p(L_p - 1) \tag{2.16}$$

dimana L_p adalah panjang $p(n)$ dan setiap elemen $p(n)$ memiliki nilai "0" atau "1" secara acak. Untuk meningkatkan ketahanan pada *watermark*, dilakukan pengacakan kode *PN sequence* dengan $w_o(n)$ (bit *watermark* asli) menggunakan operasi XOR. Bit-bit acak tersebut ($w(n)$) akan disisipkan ke sinyal audio.

$$w(n) = p(\text{mod}(n, L_p)) \oplus w_o(n) \tag{2.17}$$

2.2. Pemodelan Sistem

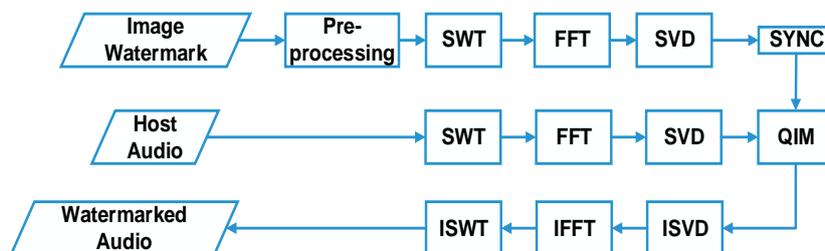
Perancangan sistem *audio watermarking* terbagi menjadi dua proses, yaitu proses penyisipan (*embedding*) dan proses ekstraksi *watermark*. Umumnya skema desain sistem *audio watermarking* ditunjukkan pada gambar 3.



Gambar 3. Sistem Audio Watermarking

2.2.1. Proses Penyisipan (*Embedding*)

Embedding adalah proses menyisipkan data *watermark* kedalam file audio. Langkah-langkah dalam proses penyisipan adalah sebagai berikut:



Gambar 4. Skema Proses Penyisipan

(Pada sisi file host audio)

Langkah 1 : Dekomposisi host audio dengan menggunakan N -level dekomposisi. Dalam penelitian ini digunakan dekomposisi pada level 2 yang mendapatkan *sub-band* LL, LH, HL, dan HL menggunakan SWT. Output dari proses ini disebut koefisien SWT ($x(n)$).

Langkah 2 : Ubah domain $x(n)$ ke $x(k)$ dengan proses FFT dengan menggunakan rumus 2.8 dan 2.9. $x(k)$ adalah sinyal audio dalam domain frekuensi.

Langkah 3 : Dekomposisi *sub-band* yang telah dipilih dari keluaran FFT (sinyal $x(k)$) menggunakan metode SVD menjadi 3 buah matriks yaitu matriks U , V , dan S_H . Proses dekomposisi SVD ini dilakukan dengan persamaan :

$$G = U * S_H * V^T \quad (2.18)$$

dimana matriks U , V adalah vektor singular kiri dan kanan (matriks ortogonal). Setelah dilakukan dekomposisi dengan SVD, matriks S_H akan diproses dalam proses QIM sedangkan matriks U dan matriks V akan diteruskan ke *Invers* SVD (ISVD).

(Pada sisi data watermark)

Langkah 4 : Ubah dimensi data *watermark* dari format 2 dimensi menjadi 1 dimensi sesuai dengan dimensi file audio dengan menggunakan proses *pre-processing*. Setelah itu, ubah setiap piksel pada gambar menjadi bit ($w(n)$).

Langkah 5 : Lakukan proses SWT pada $w(n)$ dengan menggunakan dekomposisi level-2 sehingga didapatkan *sub-band* LL, LH, HL, dan HL.

Langkah 6 : Ubah domain $w(n)$ ke $w(k)$ dengan proses FFT dengan menggunakan rumus 2.8 dan 2.9.

Langkah 7 : Dekomposisi *sub-band* yang telah dipilih dari keluaran FFT (sinyal $w(k)$) menggunakan metode SVD menjadi 3 buah matriks yaitu matriks U , V , dan S_w . Proses dekomposisi SVD ini dilakukan dengan persamaan :

$$H = U * S_w * V^T \quad (2.19)$$

Langkah 8 : Setelah proses dekomposisi dengan SVD, matriks S_w akan diteruskan kedalam proses QIM sedangkan matriks U dan matriks V akan diteruskan ke *Invers* SVD (ISVD). Sebelum itu, tambahkan bit sinkronisasi ke dalam nilai matriks S_w menggunakan persamaan 2.14 dan 2.15

(Proses penyisipan)

Langkah 9 : Lakukan proses penyisipan matriks S_H dan S_w dengan metode QIM menggunakan persamaan 2.11 dan 2.12. Keluaran dari proses QIM ini menghasilkan koefisien S_{wa} .

$$S_{wa} = S_H + S_w \quad (2.20)$$

Lalu, S_{wa} akan diteruskan dan digabungkan dengan matriks U dan V ke dalam proses ISVD sehingga:

$$G = U * S_{wa} * V^T \quad (2.21)$$

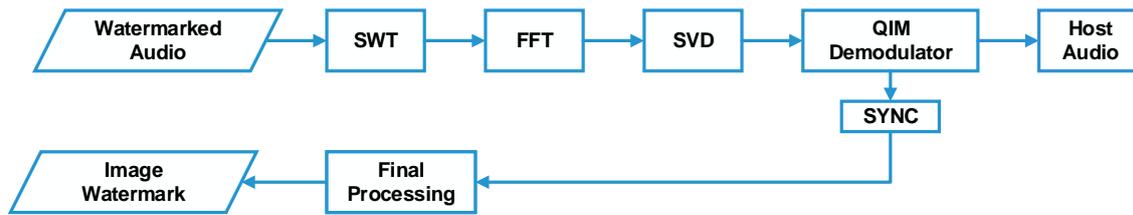
Langkah 10 : Lakukan proses ISVD sehingga menghasilkan $\hat{X}(k)$.

Langkah 11 : Lakukan proses IFFT untuk mengubah $\hat{X}(k)$ menjadi $\hat{x}(n)$ dalam domain waktu.

Langkah 12 : Lakukan proses ISWT untuk menggabungkan *sub-band* pilihan yang telah diproses dengan *sub-band* lain sehingga didapatkan sinyal audio yang telah diberi *watermark*. Sinyal audio disebut *watermarked audio* ($\hat{x}_w(n)$).

2.2.2. Proses Ekstraksi Watermark

Proses ekstraksi merupakan proses untuk mengambil kembali data *watermark* pada audio *watermark* yang telah melewati proses penyisipan atau penggabungan. Adapun langkah-langkah proses ekstraksi adalah sebagai berikut:



Gambar 5. Proses Ekstraksi

- Langkah 1 : Dekomposisi audio ter-*watermark* dengan menggunakan dekomposisi SWT level 2 menjadi beberapa *sub-band* sehingga didapatkan LL, LH, HL, dan HH.
- Langkah 2 : Ubah audio ter-*watermark* $\hat{x}_w(n)$ menjadi $\hat{x}_w(k)$ dalam domain frekuensi menggunakan metode FFT dengan persamaan 2.8 dan 2.9.
- Langkah 3 : Dekomposisi *sub-band* yang telah dipilih dari keluaran FFT ($\hat{x}_w(k)$) menggunakan SVD menjadi 3 buah matriks yaitu matriks U , V , dan \hat{S}_{wa} . Proses dekomposisi SVD ini dilakukan dengan persamaan :

$$A = U * \hat{S}_{wa} * V^T \tag{2.22}$$
- Langkah 4 : Selanjutnya, \hat{S}_{wa} akan diteruskan ke proses QIM demodulator untung memisahkan \hat{S}_H dan \hat{S}_w .
- Langkah 5 : Setelah itu, \hat{S}_w akan melewati proses desinkronisasi untuk mengetahui posisi awal *watermark* dengan mengetahui bit sinkronisasi yang telah ditambahkan pada saat proses penyisipan.
- Langkah 6 : Ubah citra *watermarked* dari 1-D menjadi 2-D menggunakan *final processing*. Setelah itu, ubah setiap bit menjadi pixel.

3. Hasil dan Analisis

3.1. Parameter pengujian

Pengujian dilakukan dengan penilaian terhadap kualitas audio ter-*watermark* yakni dengan mengukur nilai dari parameter ODG, SNR, dan MOS. Sedangkan untuk pengujian dari ketahanan citra sebagai data *watermark* hasil ekstraksi dapat diukur dengan menggunakan parameter BER. Dalam pengujian ketahanan (*robustness test*) sistem *audio watermarking* yang telah dibuat, audio ter-*watermark* diberikan beberapa macam serangan seperti serangan LPF (*Low Pass Filter*), *noise* AWGN, *resampling*, *time scale modification*, dan kompresi MP3, DA/AD, dan *ambient*.

1. Signal to Noise Ratio

Signal to Noise Ratio merupakan salah satu parameter yang digunakan untuk mengukur kualitas audio ter-*watermark* secara objektif. SNR adalah nilai yang menyatakan rasio antara daya sinyal audio yang telah disisipi *watermark* terhadap daya *noise*. Menurut *International Federation of the Phonographic Industry* (IFPI), SNR dari sinyal *audio watermarking* yang baik adalah lebih besar dari 20 dB [21]. Semakin besar nilai SNR (mendekati ∞), maka keberadaan *watermark* pada audio tidak dapat dirasakan oleh indera pendengaran atau dengan kata lain *watermarked audio* tersebut semakin *inaudible* atau *imperceptible*. Nilai SNR dapat dihitung dengan persamaan:

$$SNR = 10 \log_{10} \frac{\sum_n [x_0(n)]^2}{\sum_n [x_0(n) - x_w(n)]^2}$$

Keterangan:

- n = panjang sinyal audio
- $x_0(n)$ = sampel sinyal audio asli
- $x_w(n)$ = sampel sinyal audio ter-*watermark*

2. Objective Different Grade

Objective Differential Grade merupakan salah satu parameter objektif lainnya yang digunakan untuk mengetahui kualitas perseptual audio yang telah ter-*watermark*. Keluaran dari ODG merupakan salah satu *output* yang didapatkan dari teknik perhitungan *Perceptual Evaluation of Audio Quality* (PEAQ) yang ditetapkan dalam standar ITU-R BS.1387 (*International Telecommunication Union-Radio-communication Sector*). Penilaian dari ODG ini berkisar dari -4 sampai 0 seperti yang ditunjukkan pada tabel 3.1.

Tabel 3.1 Skala Penilaian ODG/PEAQ dan MOS

Skala ODG/PEAQ	Skala MOS	Deskripsi	Kualitas
0	5	<i>Watermark</i> tidak terdengar	Sangat baik
-1	4	<i>Watermark</i> terdengar tapi tidak mengganggu	Baik
-2	3	<i>Watermark</i> terdengar dan sedikit mengganggu	Cukup
-3	2	<i>Watermark</i> terdengar dan mengganggu	Buruk
-4	1	<i>Watermark</i> terdengar dan sangat mengganggu	Sangat buruk

3. Bit Error Rate

Bit Error Rate merupakan salah satu parameter objektif lainnya yang digunakan untuk mengukur kualitas/ketahanan hasil ekstraksi data *watermark* yang disisipkan pada *file* audio yaitu dengan cara menghitung perbandingan jumlah bit *watermark* yang error dari hasil ekstraksi dengan bit *watermark* keseluruhan sebelum dilakukan penyisipan. Persentase nilai BER dapat dihitung dengan persamaan:

$$BER = \frac{\text{jumlah bit error antara } w_o \text{ dan } w_e}{\text{jumlah bit keseluruhan } w_o} \times 100\%$$

Keterangan:

w_o = data *watermark* sebelum penyisipan

w_e = data *watermark* setelah ekstraksi

3.2 Hasil Pengujian

Kualitas audio yang diberi *watermark* dapat diukur secara subjektif atau objektif dan menghasilkan nilai yang baik. SNR dan ODG/ PEAQ adalah kriteria objektif yang digunakan untuk mengukur kualitas audio yang diberi *watermark* dengan nilai SNR ≥ 20 dB dan nilai PEAQ ≥ -1 . Kualitas *watermark* yang diekstraksi diukur dengan menggunakan parameter BER dengan nilai BER < 5%. Untuk kriteria subjektif, kita mendapatkan hasil analisis subjektif dari hasil survei menggunakan parameter MOS.

4. Kesimpulan

Dalam makalah ini, kami mengusulkan skema audio watermarking dengan menggunakan teknik kombinasi SWT-FFT-SVD dengan sinkronisasi dan QIM. *Imperceptibility* dan *robustness* yang didapatkan dengan menggunakan algoritma ini diuji dengan serangan yang berbeda seperti kompresi, *noise*, *filtering*, dll dan membuktikan bahwa metode ini kuat terhadap serangan dengan nilai SNR ≥ 20 dB dan BER < 5%.

Daftar Pustaka

- [1] M. C. Trivedi and N. K. Trivedi, "Audio masking for watermark embedding under time domain audio signals," *Proc. - 2014 6th Int. Conf. Comput. Intell. Commun. Networks, CICN 2014*, pp. 770–775, 2014.
- [2] A. Zaidi, R. Boyer, and P. Duhamel, "Audio watermarking under desynchronization and additive noise attacks," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 570–584, 2006.
- [3] V. Bhat K, I. Sengupta, and A. Das, "An adaptive audio watermarking based on the singular value decomposition in the wavelet domain," *Digit. Signal Process.*, vol. 20, no. 6, pp. 1547–1558, 2010.
- [4] I. Natgunanathan, Y. Xiang, T. Zong, and Y. Xiang, "Robustness enhancement of quantization based audio watermarking method using adaptive safe-band," *2014 IEEE Int. Conf. Commun. ICC 2014*, pp. 890–895, 2014.
- [5] C. M. Pun and X. C. Yuan, "Robust segments detector for de-synchronization resilient audio watermarking," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 21, no. 11, pp. 2412–2424, 2013.
- [6] M. Fallahpour and D. Megias, "Robust audio watermarking based on fibonacci numbers," *Proc. - 2014 10th Int. Conf. Mob. Ad-Hoc Sens. Networks, MSN 2014*, pp. 343–349, 2014.
- [7] B. Lei, I. Y. Soon, and E. L. Tan, "Robust SVD-based audio watermarking scheme with differential evolution optimization," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 21, no. 11, pp. 2368–2378, 2013.
- [8] N. V Lalitha and S. U. Rao, "Performance Analysis of DCT and DWT Audio Watermarking based on SVD," *Int. Conf. Circuit, Power Comput. Technol. [ICCPCT] Perform.*, 2016.
- [9] M. Hwang, J. Lee, M. Lee, and H. Kang, "SVD Based Adaptive QIM Watermarking on Stereo Audio Signals," vol. 9210, no. DECEMBER 2016, pp. 1–10, 2017.
- [10] A. S. S. Y. D. Chincholkar, S. R. Ganorkar, "Implementation of Audio Watermarking Technique for Copyright Protection Using SWT Algorithm," *Int. J. Eng. Appl. Technol.*, pp. 42–48, 2016.
- [11] K. K. Kumar, "FPGA Implementation of Denoising Speech Signal using Stationary Wavelet Transform," *Int. J. Adv. Res. TRENDS Eng. Technol.*, vol. 3777, pp. 156–162, 2015.
- [12] C. U. I. Delong, L. Qirui, Y. U. Guilan, and X. Jianbin, "Content-Based Audio Watermarking Method to Resist De-Synchronization Attacks," *2014 Int. Conf. Inf. Netw. Secur.*, 2014.
- [13] D. Ambika and V. Radha, "Speech Watermarking Using Discrete Wavelet Transform, Discrete Cosine Transform And Singular Value Decomposition," *Int. J. Comput. Sci. Eng. Technol.*, vol. 5, no. 11, pp. 1089–1093, 2014.
- [14] S. Wu, J. Huang, D. Huang, and Y. Q. Shi, "Efficiently self-synchronized audio watermarking for assured audio data transmission," *IEEE Trans. Broadcast.*, vol. 51, no. 1, pp. 69–76, 2005.
- [15] K. Khaldi and A. O. Boudraa, "Audio watermarking via EMD," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 21, no. 3, pp. 675–680, 2013.
- [16] J. H. J. Huang, Y. W. Y. Wang, and Y. Q. Shi, "A blind audio watermarking algorithm with self-synchronization," *2002 IEEE Int. Symp. Circuits Syst. Proc. (Cat. No.02CH37353)*, vol. 3, pp. 627–630, 2002.
- [17] H. Kaur and U. Kaur, "Proposal Paper for improving SVD and Quantization Technique for Audio Watermarking," *Int. J. Curr. Eng. Technol.*, vol. 3, no. 2, pp. 624–627, 2013.
- [18] J. Wang, M. Lai, K. Liang, and P. Chang, "Adaptive Wavelet Quantization Index Modulation Technique for Audio Watermarking," *Int. Comput. Symp.*, no. 1, 2006.
- [19] Y. Xiang, I. Natgunanathan, S. Guo, W. Zhou, and S. Nahavandi, "Patchwork-based audio watermarking method robust to de-synchronization attacks," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 22, no. 9, pp. 1413–1423, 2014.
- [20] J. M. Deng, H. Z. Yue, Z. Z. Zhuo, H. G. Yan, D. Liu, and H. Y. Li, "A stationary wavelet

- transform based approach to registration of planning CT and setup cone beam-CT images in radiotherapy,” *J. Med. Syst.*, vol. 38, no. 5, 2014.
- [21] A. C. Sekhar, C. Suneetha, G. NagaLakshmi, and B. V. RaviKumar, “Fast Fourier Transforms and Quadratic Forms for Digital Audio Watermarking,” *2009 Int. Conf. Adv. Recent Technol. Commun. Comput.*, pp. 449–452, 2009.
- [22] P. K. Dhar and T. Shimamura, “Audio Watermarking in Transform Domain Based on Singular Value Decomposition and Quantization,” *2012 18th Asia-Pacific Conf. Commun.*, pp. 516–521, 2012.
- [23] D. Tarun and J. Mishra, *Digital Signal Processing (Principles and Applications)*, India: S. K. Rastogi for KRISHNA Prakashan Media (P) Ltd., 2006.
- [24] G. Budiman, A. B. Suksmono, and D. Danudirdjo, “Fibonacci Sequence Based FFT and DCT Performance Comparison in Audio Watermarking,” *Pertanika J. Sci. Technol.*, vol. 24, no. 1, pp. 1–10, 2016.
- [25] G. Budiman, A. B. Suksmono, D. Danudirdjo, K. Usman, and D. H. Shin, “A modified multicarrier modulation binary data embedding in audio file,” *Int. J. Electr. Eng. Informatics*, vol. 8, no. 4, pp. 762–773, 2016.
- [26] A. Muliansyahputra, B. H. Akbar, and G. Budiman, “Perancangan dan Analisis Audio Watermarking Berbasis Teknik Modulasi Digital dengan Pengkodean Konvolusi,” *Semin. Nas. Inov. Dan Apl. Teknol. Di Ind.* ISSN 2085-4218, pp. 1–6, 2017.