

COMPRESSIVE SENSING PADA IMAGE WATERMARKING BERBASIS DWT-SVD DENGAN REKONSTRUKSI BASIS PURSUIT DENOISING

Alda Ghifary¹, Efri Suhartono², Irma Safitri³

^{1,2,3} Universitas Telkom, Bandung

ghifaryalda@student.telkomuniversity.ac.id¹, Efrisuhartono@telkomuniversity.ac.id²,

irmasaf@telkomuniversity.ac.id³

Abstrak - *Watermarking* adalah suatu cara untuk mencegah pembajakan dengan cara menyisipkan informasi yang tidak kasat mata, tahan terhadap serangan, dan tidak merusak kualitas citra *host*. Pada penelitian ini digunakan metode *compressive sensing* dalam sistem *image watermarking* dengan algoritma penyisipan *Discrete Wavelet Transform-Singular Value Decomposition* (DWT-SVD) dan algoritma rekonstruksi *Basis Pursuit Denoising* (BPDN). Pengujian sistem dengan menggunakan *measurement rate* 80%, *sub-band* LH, dan *mother wavelet* db2 menghasilkan nilai rata-rata PSNR sebesar 141,812dB, MSE sebesar 0,045, BER sebesar 0,026, dan SSIM sebesar 0,878. Sistem ini mampu bertahan dengan baik terhadap serangan *noise salt and pepper*.

Kata Kunci: *Image Watermarking, Compressive Sensing (CS), DWT-SVD, BPDN*

Abstract –*Watermarking is a way to prevent piracy by inserting invisible and robust information that do not damage the quality of the host image. This research uses compressive sensing method in an image watermarking system with Discrete Wavelet Transform-Singular Value Decomposition (DWT-SVD) insertion algorithm and Basis Pursuit Denoising (BPDN) reconstruction algorithm. The test that was conducted, which used 80% measurement rate, LH sub-band, and db2 mother wavelet produced an average output value as follows; 141,812dB of PSNR, 0,045 of MSE, 0,026 of BER, and 0,878 of SSIM. This system holds itself really well against noise salt and pepper attack.*

Keywords: *Image Watermarking, Compressive Sensing (CS), DWT-SVD, BPDN*

1. Pendahuluan

Watermarking adalah cabang penyembunyian informasi yang digunakan untuk menyembunyikan informasi hak milik dalam media digital seperti foto, musik, ataupun video^[1]. Agar bisa efektif, watermark harus memenuhi beberapa kriteria, yaitu tidak kasat mata dalam media *host*, tidak terlihat secara statistik untuk menangkal proses penghilangan tidak berizin, siap diekstraksi oleh pemilik konten, dan tahan terhadap distorsi sinyal yang terjadi dalam media *host*.

Permasalahan saat ini pada *watermarking* adalah keterbatasan ukuran penyimpanan. Oleh karena itu, dibutuhkan metode kompresi citra agar penyimpanan yang tersedia dapat menampung citra tersebut dan menjaga kualitasnya dengan baik. *Compressive sensing* adalah salah satu metode yang digunakan untuk memperkecil ukuran citra agar media penyimpanan dapat digunakan secara lebih efektif. Proses rekonstruksi citra watermark pada *compressive sensing* tidak dapat dilakukan apabila terdapat *noise*. Algoritma rekonstruksi *basis pursuit denoising* (BPDN) digunakan agar proses rekonstruksi citra watermark dapat dilakukan walaupun terdapat *noise*. Berdasarkan permasalahan ini, penulis melakukan penelitian yang bertujuan untuk memperkecil ukuran *file* citra dan melakukan rekonstruksi citra agar citra watermark yang dihasilkan berkualitas. Perancangan dilakukan dengan metode penyisipan *Discrete Wavelet Transform-Singular Value Decomposition* (DWT-SVD).

2. Dasar Teori

2.1. Watermarking

Watermarking adalah proses penyembunyian data dengan cara menyisipkan sebuah kode digital ke dalam data *host*. Watermark tidak dapat dihilangkan, tahan terhadap berbagai serangan, dan tidak kasat mata. Pada watermark terdapat informasi tentang asal, status, dan destinasi data.

2.2. JPEG^[2]

JPEG adalah format gambar yang dibuat oleh *Joint Photographic Experts Group*. JPEG digunakan untuk standar kompresi. JPEG dapat ditemukan dalam berbagai aplikasi. Sebagai contoh, kebanyakan kamera digital di pasaran mengeksport format ini, dan kebanyakan *image editing software* seperti Adobe Photoshop, GIMP, mendukung operasi kompresi JPEG.

2.3. Citra Digital^[3]

Citra adalah gambaran dari sebuah objek. Ada dua jenis citra, yaitu citra analog dan citra digital. Citra analog tidak dapat diolah oleh computer. Agar citra analog bisa diolah, perlu dilakukan proses konversi menjadi citra digital. Peralatan digital dapat menghasilkan citra digital dengan menggunakan proses *sampling* dan kuantisasi. Sistem *sampling* mengubah citra kontinu menjadi citra digital dengan membagi citra analog menjadi M baris dan N kolom, lalu menjadi citra diskrit. Semakin besar nilai M dan N, maka kualitas citra digital akan lebih bagus. Pertemuan baris dan kolom disebut *pixel*. Sistem kuantisasi melakukan perubahan intensitas analog ke intensitas diskrit untuk membuat gradasi warna sesuai kebutuhan. Kedua sistem ini menghasilkan resolusi citra yang diinginkan.

Pixel memiliki dua parameter, yaitu koordinat dan intensitas. Nilai yang ada di koordinat (x,y) adalah $f(x,y)$, yaitu besar intensitas *pixel* di titik itu. Matriks sebuah citra digital yaitu:

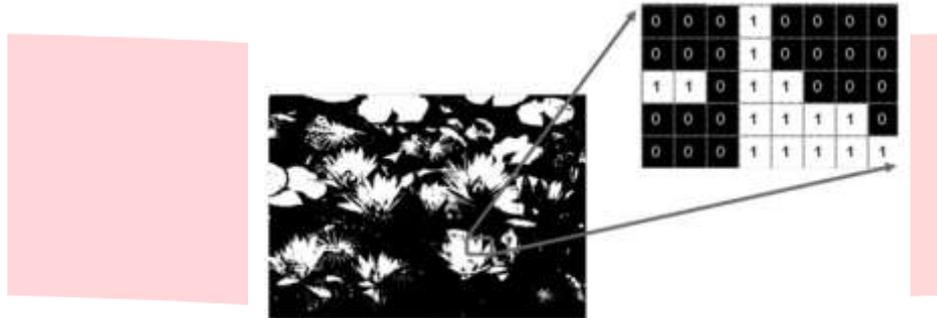
$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{pmatrix} \quad (2, 1)$$

2.3.1. Jenis Citra

Nilai suatu *pixel* secara umum memiliki rentang 0 hingga 255. Citra dengan penggambaran ini tergolong ke dalam citra integer. Jenis-jenis citra berdasarkan nilai *pixel*-nya yaitu:

1. Citra Biner

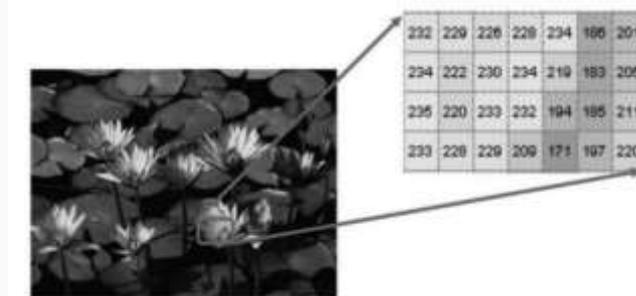
Citra biner hanya memiliki dua nilai *pixel*, yaitu hitam dan putih. Citra biner disebut juga sebagai citra B&W (*black and white*) atau citra monokrom. Hanya dibutuhkan 1 bit untuk mewakili nilai setiap pixel dari citra biner.



Gambar 1 Citra Biner^[3]

2. Citra Grayscale

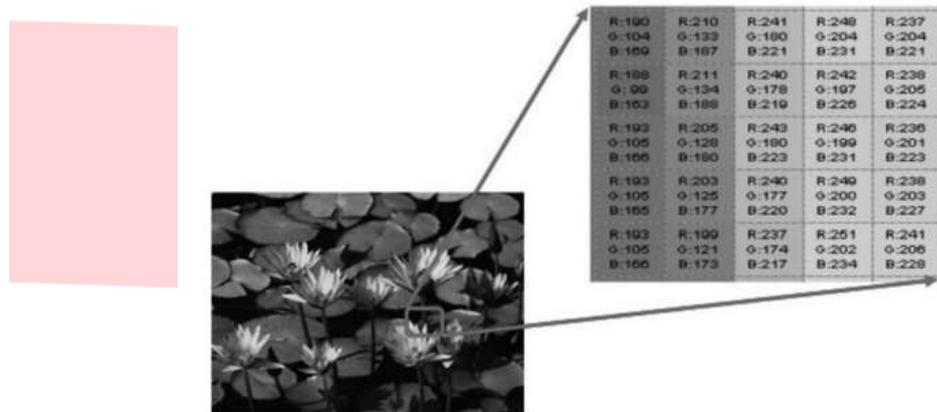
Citra *grayscale* adalah matriks yang nilainya mewakili intensitas setiap *pixel* dengan rentang nilai 0 hingga 255. Setiap *pixel* membutuhkan 8 bit memori. Gambar 2.2 memperlihatkan citra *grayscale* dari dekat beserta nilai intensitas *pixel*



Gambar 2 Citra Grayscale^[3]

3. Citra Warna

Citra Warna adalah citra dengan *pixel* yang memiliki tiga komponen warna, yaitu *red*, *green*, dan *blue*. Warna ditentukan oleh kombinasi dari intensitas warna merah, hijau, dan biru yang berada di bidang warna dalam okasi *pixel*. Setiap *pixel* membutuhkan 24 bit memori, masing-masing 8 bit berasal dari komponen merah, hijau, dan biru. Citra warna memiliki 24 juta kemungkinan warna.

Gambar 3 Citra Warna^[3]

2. 4. Compressive Sensing^[4]

Dalam pengolahan sinyal dan citra, ada proses untuk merekonstruksi sinyal dari data terukur. Saat proses akuisisi informasi linier, masalahnya yaitu menyelesaikan persamaan sistem linier. Dalam istilah matematika, data yang diamati yaitu $\mathbf{y} \in \mathbb{C}^m$, terkoneksi dengan sinyal $\mathbf{x} \in \mathbb{C}^N$ melalui

$$\mathbf{Ax} = \mathbf{y}. \quad (2, 2)$$

Matriks $\mathbf{A} \in \mathbb{C}^{m \times N}$ memodeli proses pengukuran linier informasi. Lalu akan dicoba memulihkan vector $\mathbf{x} \in \mathbb{C}^N$ dengan menyelesaikan sistem linier di atas. Jumlah pengukuran data yang terukur harus paling tidak sebesar panjang sinyal. Prinsip ini adalah dasar untuk kebanyakan perangkat yang digunakan dalam teknologi, seperti konversi *analog-to-digital*, pencitraan medis, radar, dan komunikasi bergerak. Tanpa informasi tambahan, kita tidak bisa memulihkan \mathbf{x} dari \mathbf{y} dalam kasus $m < N$. Fakta ini berhubungan dengan teorema *sampling* Shannon, yang menyatakan bahwa *sampling rate* suatu sinyal kontinu dua kali lebih besar dari frekuensi tertinggi untuk memastikan rekonstruksi. Fenomena ini disebut *compressive sensing*.

Oleh karena itu, memungkinkan untuk merekonstruksi sinyal menggunakan algoritma efisien saat m lebih kecil dari N . Asumsi yang membuat hal ini memungkinkan disebut *sparsity*. Sebuah sinyal dikatakan sparse apabila kebanyakan komponennya bernilai nol. Seperti yang diamati secara empiris, banyak sinyal di dunia nyata yang dapat dikompresi dalam artian sinyal-sinyal tersebut diperkirakan dengan bagus oleh sinyal *sparse*.

Untuk kebutuhan praktis, ketersediaan algoritma rekonstruksi yang cepat itu sangat penting. Fitur ini adalah hal yang menarik perhatian pada *compressive sensing*. Pendekatan algoritmik yang kiranya pertama terpikirkan yaitu minimisasi ℓ_0 . Notasi $\|\mathbf{x}\|_0$ adalah jumlah masukan *nonzero* pada vektor \mathbf{x} , wajar untuk merekonstruksi \mathbf{x} sebagai solusi masalah optimasi kombinatorial.

$$\text{minimize } \|\mathbf{z}\|_0 \text{ subject to } \mathbf{Az} = \mathbf{y}. \quad (2, 3)$$

Di sini kita mencari vektor konsisten yang paling *sparse* dengan data $\mathbf{y} = \mathbf{Ax}$ yang telah terukur. Namun, minimisasi ℓ_0 pada umumnya sulit. Oleh karena itu, digunakan metode yang sekarang populer dan dapat dipahami dengan baik, yaitu minimisasi ℓ_1 , yang terdiri dari pencarian *minimizer* dari masalah tersebut. Norma ℓ_1

$\| \cdot \|_1$ adalah fungsi cembung. Oleh karena itu, masalah optimasi ini dapat diselesaikan dengan metode efisien dari optimasi cembung.

2. 5. *Basis Pursuit Denoising (BPDN)*^[5]

Basis pursuit adalah algoritma rekonstruksi yang berfungsi untuk merekonstruksi sinyal yang telah dikompresi. *Basis pursuit* mengaplikasikan persamaan norma ℓ_1 yang digunakan dalam proses *compressive sensing* untuk mengestimasi sinyal sparse dari input sinyal dalam suatu matriks sebagai hasil dari kompresi. *Basis pursuit* hanya dapat dilakukan apabila pengukuran bebas dari *noise*. Terdapat nilai \hat{s} sebagai hasil estimasi, y sebagai *output* sinyal, dan Θ sebagai matriks *transform* M^*N . Persamaannya yaitu:

$$\hat{s} = \arg \min_s \|s\|_1; \text{ subject to } \Theta \cdot s = y \quad (2, 4)$$

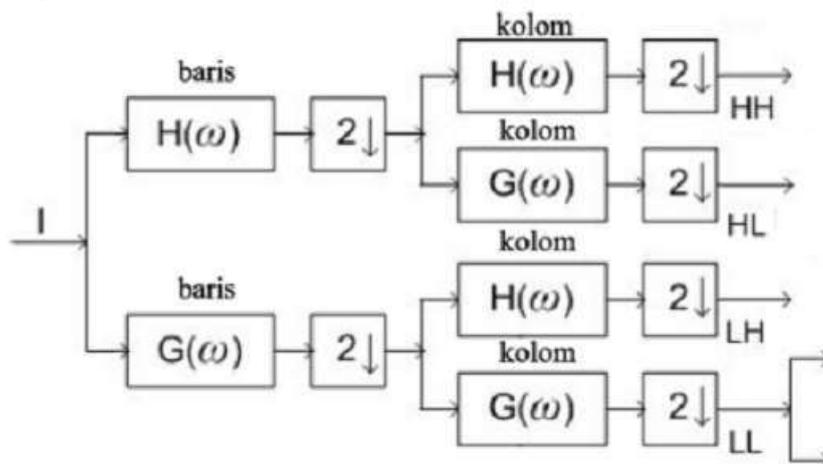
Denoising berfungsi untuk mengukur *noise*. Apabila pengukuran rusak karena *noise* yang terlalu besar, rekonstruksi tidak dapat dilakukan. *Denoising* dapat dilakukan dengan mengubah batasan dalam pengukuran *noise*.

Basis Pursuit Denoising digunakan agar dalam pengukuran terdapat *noise*, rekonstruksi tetap dapat dilakukan. BPDN mencari solusi minimum norma ℓ_1 pada keadaan *constraint*. *Constraint* kuadrat yang diaplikasikan BPDN mendefinisikan bahwa solusi norma ℓ_2 , disebut juga sebagai norma Euclidean, yang diperoleh dari *error* nilai y harus $< \epsilon$. Persamaannya yaitu:

$$\hat{s} = \arg \min_s \|s\|_1; \text{ subject to } \frac{1}{2} \|y - \Theta \cdot s\|_2^2 \leq \epsilon \quad (2, 5)$$

2. 6. *Discrete Wavelet Transform-Singular Value Decomposition (DWT-SVD)*

Wavelet Transform diperkenalkan oleh Morlet dan Grossman pada tahun 1980-an sebagai fungsi matematis yang berguna untuk merepresentasikan data. Teknik dasar dekomposisi sinyal waktu diskrit ini diaplikasikan pada tahun 1976. Dalam DWT, gambaran suatu skala sinyal digital diperoleh dengan teknik filterisasi digital. DWT membagi sebuah sinyal menjadi frekuensi tinggi dan frekuensi rendah. Proses tersebut dinamakan dekomposisi. *Output* dari *highpass filter* (HPF) dan *lowpass filter* (LPF) menghasilkan koefisien DWT. Citra asli direkonstruksi menggunakan koefisien ini. Dekomposisi ini menghasilkan HH (*high-high frequency*), HL (*high-low frequency*), LH (*low-high frequency*), dan LL (*low-low frequency*). Secara teknis, citra dua dimensi dapat didekomposisi seperti Gambar 2.4. dengan $H(\omega)$ sebagai filter *highpass* dan $G(\omega)$ sebagai filter *lowpass*.



Gambar 4 Dekomposisi *wavelet* terhadap suatu citra^[6]

Singular Value Decomposition (SVD) merupakan metode yang digunakan untuk diagonalisasi matriks dalam analisis numerik. Matriks sebuah citra C didekomposisi menjadi tiga sub matriks. Tiga sub matriks tersebut yaitu U , S , dan V .

$$C = U * S * V^T \quad (2, 6)$$

U dan V adalah matriks ortogonal, sehingga $U * U^T = I$ dan $V * V^T = I$, I adalah matriks identitas dan S adalah matriks diagonal. Nilai ini disebut *singular value*. U dan V adalah vektor *singular*. Proses dekomposisi ini disebut *Singular Value Decomposition*. SVD diaplikasikan pada matriks sebuah citra untuk memberi nilai tunggal matriks diagonal yang mewakili intensitas warna citra^[6].

Dari perspektif pengolahan citra, sebuah citra dapat dipandang sebagai matriks dengan entri skalar *nonnegative*. SVD dari sebuah citra A dengan besar $m \times m$ didapatkan oleh $A = U S V^T$. U dan V adalah matriks ortogonal, dan $S = \text{diag}(l_i)$ adalah matriks diagonal dari nilai tunggal l_i , $i = 1, \dots, m$, yang tersusun dari yang tertinggi menuju yang terendah. Kolom U adalah vektor tunggal kiri, sedangkan kolom V adalah vektor tunggal kanan sebuah citra A . Prinsip dasar di balik konsep teknik *watermarking* berbasis SVD yaitu untuk mencari SVD dari citra *cover* atau setiap blok dari citra *cover*, lalu mengubah nilai tunggalnya untuk menyisipkan watermark. Ada dua komponen utama untuk menerapkan metode SVD dalam skema *digital-watermarking*, yaitu saat gangguan kecil ditambahkan ke dalam citra, variasi besar dari nilai tunggalnya tidak terjadi, dan nilai tunggal yang merepresentasikan komponen citra aljabar intrinsik^[7].

2. 7. Jenis Serangan pada Watermark^[8]

Ada beberapa serangan yang dapat dilakukan pada watermark. Serangan ini dikategorikan menjadi dua kategori, yaitu *state-of-the-art* dan *estimation-based*. Untuk serangan *state-of-the-art* ada beberapa jenis serangan, diantaranya:

1. *Removal Attacks*

Serangan ini bertujuan untuk menghilangkan informasi watermark dari data yang telah disisipkan watermark tanpa menjebol sistem keamanan dari algoritma *watermarking*.

2. *Geometric Attacks*

Serangan ini tidak menghilangkan watermark yang telah disisipkan, justru bertujuan untuk mengubah detector sinkronisasi watermark dengan informasi yang telah disisipkan.

3. *Cryptographic Attacks*

Serangan ini menjebol metode keamanan dalam skema *watermarking* lalu menemukan cara untuk menghilangkan informasi watermark yang telah disisipkan atau untuk menyisipkan watermark palsu.

4. *Protocol Attacks*

Serangan ini menyerang seluruh konsep aplikasi *watermarking*. Salah satu tipe serangan ini berdasarkan konsep *invertible watermarks*.

Kategori kedua yaitu serangan *estimation-based*. Serangan ini mengeksploitasi statistik data asli dan sinyal watermark. Ada tiga jenis serangan dalam kategori ini, diantaranya:

1. *Remodulation Attacks*

Serangan ini bertujuan untuk memodifikasi watermark menggunakan metode modulasi yang berlawanan dengan metode yang digunakan untuk menyisipkan watermark tersebut.

2. Copy Attack

Watermark dapat dieksploitasi untuk mengimplementasikan serangan ini. Fenomena sensitivitas kontras dan *texture masking* dari *Human Visual System* (HVS) dapat dieksploitasi.

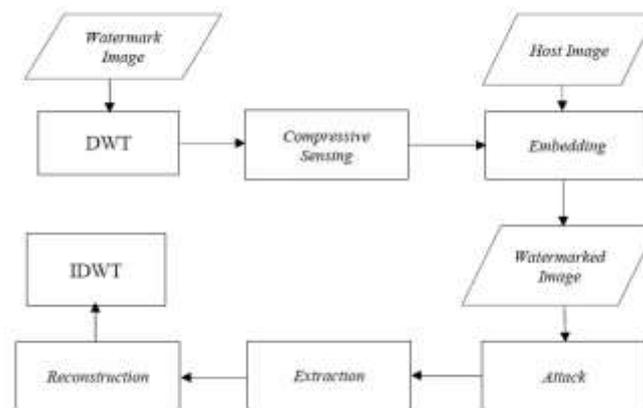
3. Synchronization Removal

Serangan ini bertujuan untuk mendeteksi pola sinkronisasi, menghilangkannya, lalu menerapkan teknik desinkronisasi, seperti *global affine transformation*.

3. Perancangan dan Implementasi Sistem

3.1. Deskripsi Sistem

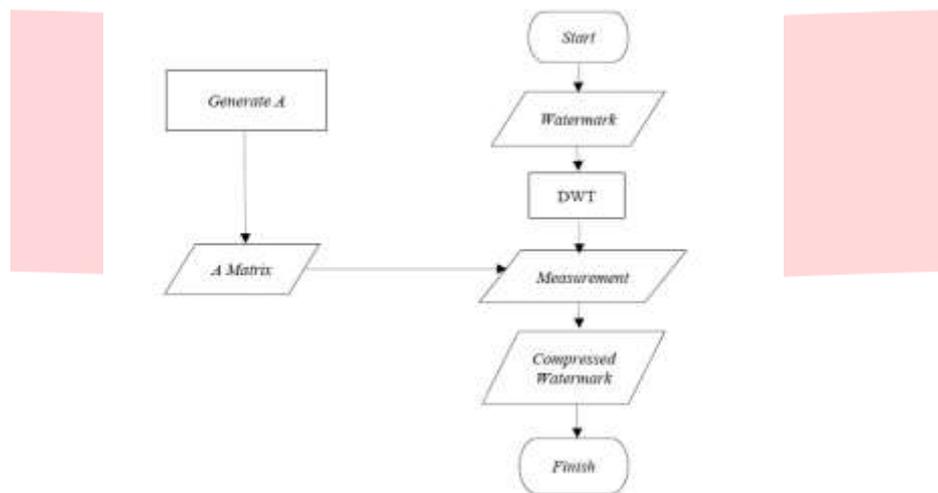
Pada penelitian ini dirancang dan diimplementasikan sistem *image watermarking* yang terdiri dari dua citra, yaitu citra *host* dan citra watermark dengan format JPEG. *Compressive Sensing* dilakukan pada citra watermark dengan algoritma penyisipan DWT-SVD. Setelah citra *host* disisipkan watermark, akan dilakukan beberapa skenario penyerangan untuk pengujian ketahanan watermark tersebut. Selanjutnya dilakukan proses ekstraksi dan rekonstruksi menggunakan algoritma *Basis Pursuit Denoising*, fungsinya untuk mengembalikan citra watermark yang terkompresi ke keadaan semula. Berikut adalah *flowchart* dari keseluruhan rancangan sistem.



Gambar 5 Flowchart Rancangan Sistem

3.2. Compressive Sensing

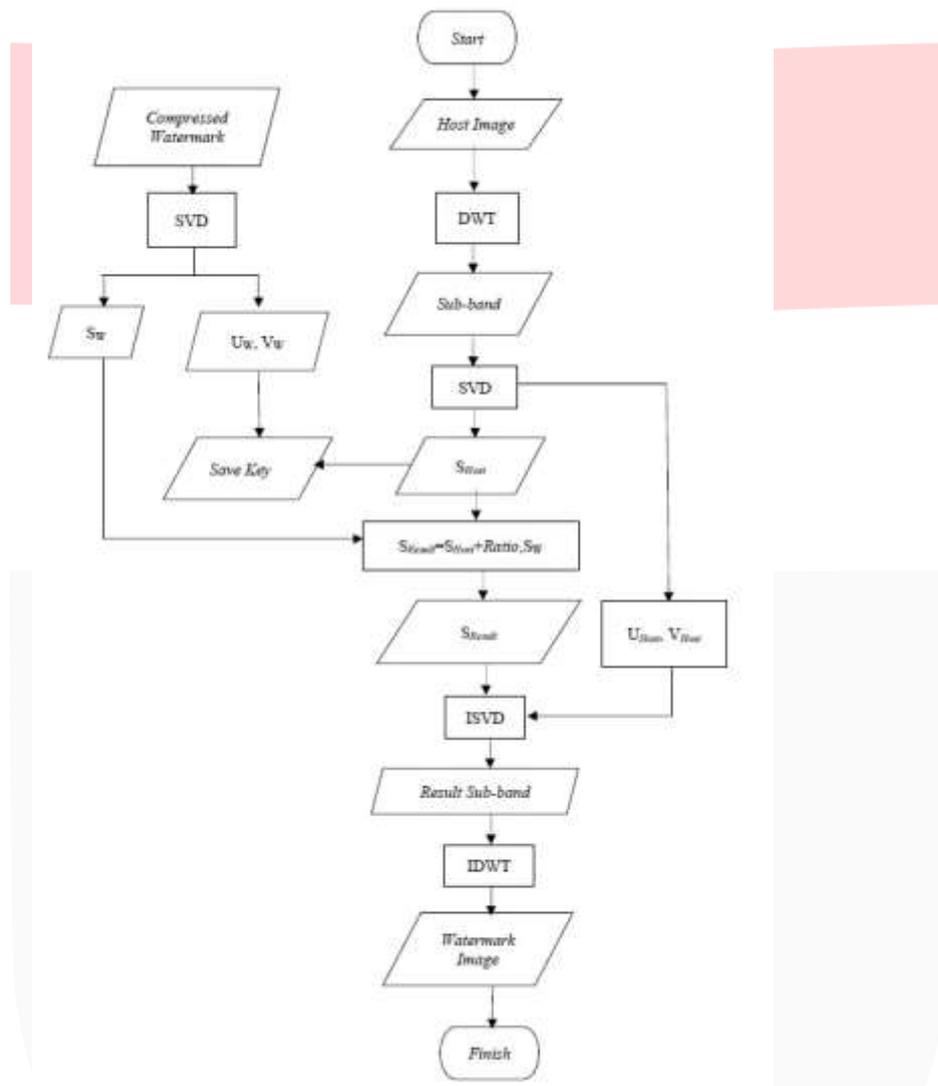
Pada penelitian ini digunakan *Compressive Sensing* berbasis *Discrete Wavelet Transform* (DWT) dan *Singular Value Decomposition* (SVD) untuk proses kompresi watermark. Pertama dilakukan transformasi DWT untuk memperoleh koefisien frekuensi, lalu untuk membagi frekuensi menjadi dua, yaitu frekuensi rendah dan tinggi, dilakukan transformasi SVD. Dari proses ini dihasilkan sinyal *sparse*. Nilai *sampling* yang diperoleh dari sinyal *sparse* direduksi menjadi matriks yang lebih kecil tergantung dengan ukuran matriks A. Matriks A adalah matriks ukuran. Jumlah resolusi baris di matriks ini kemudian dikompresi sesuai besar *measurement rate* yang telah ditentukan sebelumnya. Berikut adalah *flowchart* proses *Compressive Sensing*.



Gambar 6 *Flowchart Compressive Sensing*

3.3. Proses *Embedding*

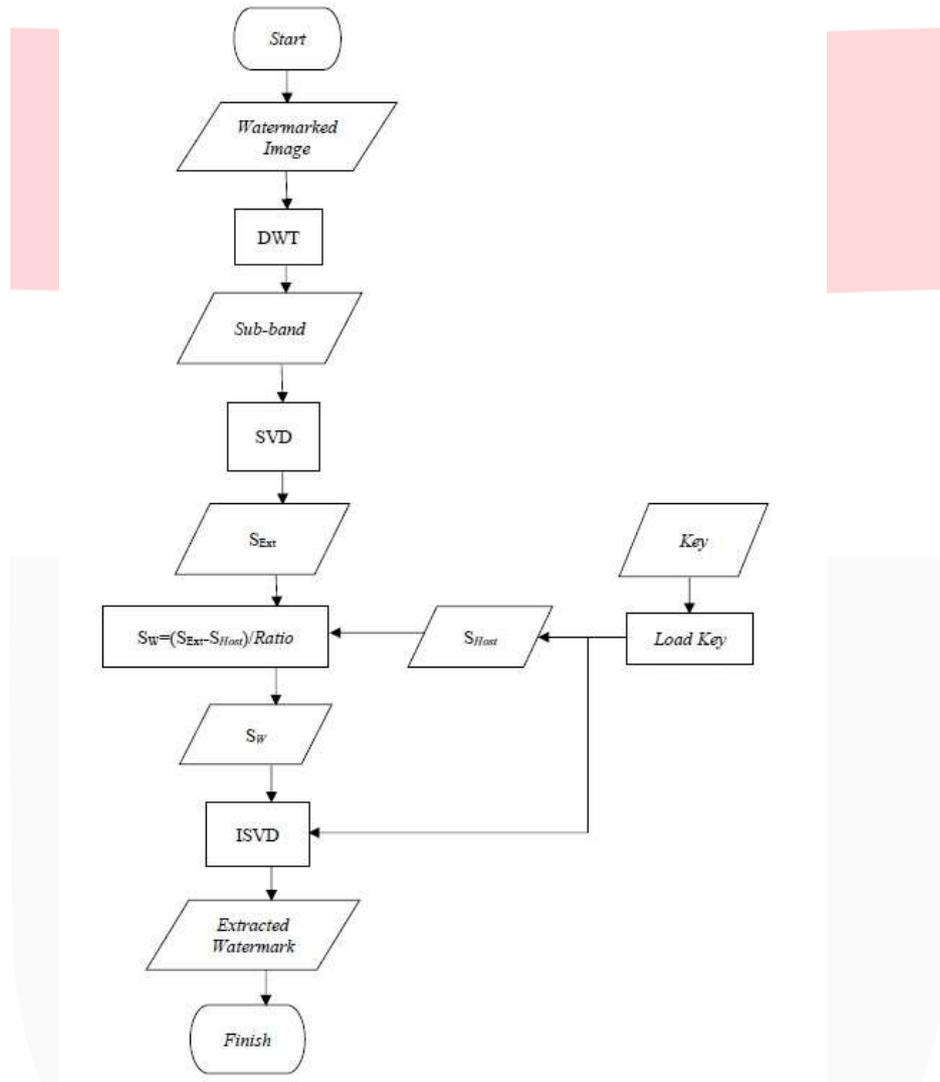
Pada proses ini watermark disisipkan ke dalam citra *host*. Tahapan yang dilakukan sesuai dengan metode DWT-SVD. Langkah pertama yaitu membuka *file* citra digital asli kemudian dijadikan citra *host*. Citra ini merupakan citra RGB dengan format JPEG. Selanjutnya, pilih *file* citra yang akan dijadikan watermark. Citra ini merupakan citra *black and white* dengan format PNG. Lalu transformasi DWT dilakukan. Citra didekomposisi menjadi empat *sub-band*, yaitu HH, HL, LH, dan LL. Berikutnya dilakukan pemilihan *sub-band* untuk disisipkan informasi tersembunyi. Proses *embedding* dilakukan dengan menggunakan metode SVD. Terdapat matriks U, S, dan V dalam proses ini. *Compressed watermark* menghasilkan U_w , S_w , dan V_w sebagai hasil SVD. *Sub-band* menghasilkan U_{Host} , S_{Host} , dan V_{Host} . Kemudian transformasi *inverse* dari SVD dilakukan. Transformasi IDWT dilakukan untuk penggabungan *sub-band* yang sebelumnya didekomposisi. Berikut adalah *flowchart* dari proses *embedding*.



Gambar 7 Flowchart Proses Embedding

3. 4. Proses Ekstraksi

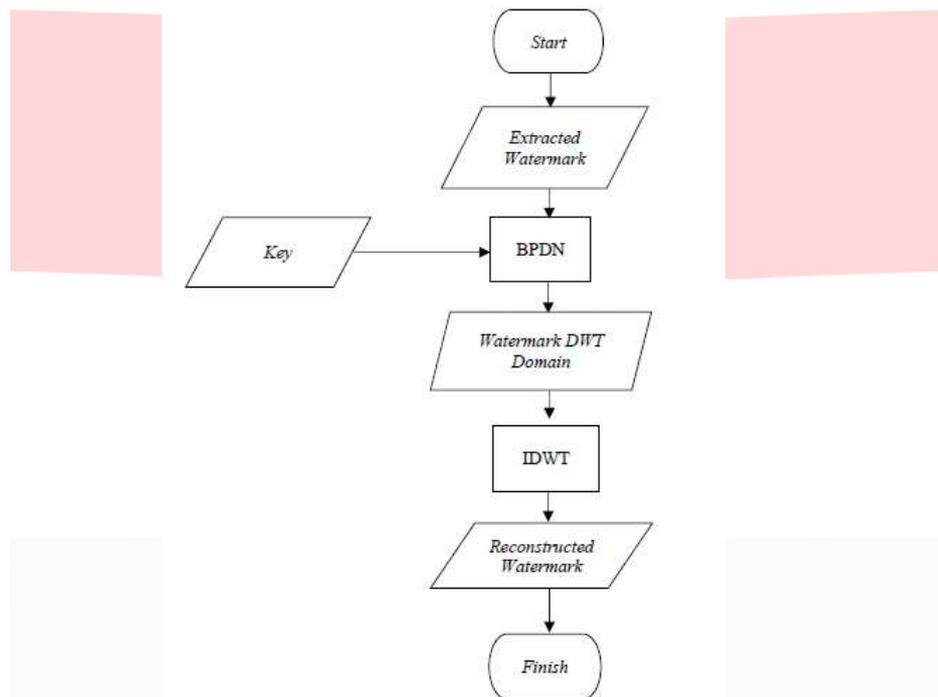
Pada proses ini dilakukan pemisahan informasi, yaitu watermark, dengan citra *host* agar didapatkan informasi yang asli. Informasi yang akan diekstraksi ini telah disisipkan dan diberi serangan. Pertama, buka *watermarked image*. Pada *watermarked image* dilakukan transformasi DWT, lalu *watermarked image* didekomposisikan menjadi empat *sub-band*, yaitu HH, HL, LH, dan LL. Proses ekstraksi ini menggunakan metode SVD, sehingga nilai matriks *sub-band* dikurangi dengan nilai matriks citra *sub-band*. Lalu dilakukan transformasi *inverse* DWT. Setelah itu dihasilkan *compressed watermark* sebagai hasil dari proses ekstraksi.



Gambar 8 Flowchart Proses Ekstraksi

3. 4. 3. Proses Rekonstruksi

Proses ini adalah proses terakhir untuk memperoleh kembali watermark. Tujuan dari proses rekonstruksi yaitu untuk melakukan rekonstruksi sinyal dari hasil pengukuran *Compressive Sensing*. Sinyal *sparse* merupakan hasil transformasi DWT dari watermark. Rekonstruksi sinyal pada penelitian ini menggunakan algoritma *Basis Pursuit Denoising* (BPDN). Berikut adalah *flowchart* dari proses rekonstruksi:



Gambar 9 Flowchart Proses Rekonstruksi

4. Pengujian dan Analisis

4. 1. Lingkup Pengujian Sistem

Dari sistem yang telah dirancang, penulis melakukan empat skenario pengujian agar mendapatkan hasil untuk dianalisis. Skenario tersebut yaitu pengujian *measurement rate*, pengujian *sub-band*, pengujian *mother wavelet*, dan pengujian serangan. Beberapa *output* dari sistem dapat dilihat pada saat proses *embedding* yang selanjutnya akan direkonstruksi menggunakan algoritma BPDN dengan metode penyisipan DWT-SVD. Pada citra watermark dilakukan proses *compressive sensing*. Baik atau buruknya hasil pengujian sistem dapat dilihat berdasarkan nilai *output* pada parameter PSNR, MSE, BER, dan SSIM. Pengambilan data dilakukan dengan format *file* citra watermark hitam putih dengan format *file* BMP dan citra *host* RGB dengan format *file* JPEG.

4. 1. 1. Citra Watermark

Pada skenario pengujian sistem, digunakan citra watermark dengan resolusi 64x64. Format citra watermark adalah BMP. Berikut adalah citra watermark yang digunakan.



Gambar 10 Citra Watermark

4. 1. 2. Citra *Host*

Ada dua citra *host* yang digunakan pada skenario pengujian sistem. Keduanya memiliki resolusi 800x800. Citra *host* pertama mempunyai komposisi warna lebih terang, sedangkan citra *host* kedua mempunyai komposisi warna lebih gelap. Dua citra *host* tersebut seperti gambar di bawah ini.



Gambar 11 Citra *Host* Pertama



Gambar 12 Citra *Host* Kedua

4. 2. Skenario Pengujian *Measurement Rate*

Pada pengujian ini, *measurement rate* digunakan pada proses *compressive sensing*. Dalam skenario ini, akan dianalisis pengaruh besarnya *measurement rate* terhadap baik buruknya nilai parameter yang sudah ditentukan. *Measurement rate* yang digunakan yaitu mulai dari 10% hingga 80%. Berdasarkan hasil yang ditunjukkan Tabel 4.1 di bawah, dapat disimpulkan bahwa semakin besar persentase *measurement rate*, nilai BER dan SSIM semakin baik. Hal ini dikarenakan *measurement rate* berbanding terbalik dengan rasio kompresi. Rasio kompresi pada citra *watermark* akan semakin rendah apabila *measurement rate* semakin besar.

Tabel 1 Hasil Rata-rata Pengujian *Measurement Rate* terhadap Parameter Performa

<i>Measurement Rate</i>	PSNR	MSE	BER	SSIM
10%	138,179	0,069	0,27	0,268
20%	140,908	0,053	0,254	0,272
30%	140,619	0,05	0,202	0,38
40%	141,174	0,049	0,145	0,491
50%	139,973	0,055	0,097	0,633
60%	141,184	0,048	0,072	0,718
70%	141,612	0,046	0,046	0,788
80%	140,625	0,05	0,034	0,8

4.3. Skenario Pengujian *Sub-Band*

Pada skenario ini akan terlihat *sub-band* manakah yang menghasilkan nilai parameter terbaik dalam citra *host*. Ada empat jenis *sub-band* yang diujikan. Jenis *sub-band* tersebut diantaranya *sub-band low low* (LL), *sub-band low high* (LH), *sub-band high low* (HL), dan *sub-band high high* (HH). Pada Tabel 4.2 dapat dilihat bahwa *sub-band* LL menghasilkan nilai PSNR yang terbaik, yaitu sebesar 143,081 dB. Kemudian untuk nilai BER terbaik dihasilkan oleh *sub-band* LH sebesar 0,011. Parameter terakhir yaitu SSIM, nilai terbaik kembali dihasilkan oleh *sub-band* LH. Dari skenario ini, dapat disimpulkan bahwa *sub-band* terbaik untuk menyisipkan watermark dengan metode-metode yang telah ditentukan sebelumnya pada penelitian ini yaitu *sub-band* LH.

Tabel 2 Hasil Rata-rata Pengujian *Sub-Band* terhadap Parameter Performa

Parameter	<i>sub-band</i> LL	<i>sub-band</i> LH	<i>sub-band</i> HL	<i>sub-band</i> HH
PSNR	143,081	140,664	142,767	141,159
MSE	0,04	0,05	0,04	0,047
BER	0,06	0,011	0,014	0,021
SSIM	0,711	0,939	0,9	0,881

4.4. Skenario Pengujian *Mother Wavelet*

Pada skenario ini, akan diujikan tiga *mother wavelet*, yaitu db1, db2, dan db3. Berdasarkan hasil pengujian pada tabel 4.3, nilai BER dan SSIM terbaik ada pada *mother wavelet* db2. Nilai BER yang dihasilkan pada *mother wavelet* db2 yaitu sebesar 0,024 dan nilai SSIM sebesar 0,865.

Tabel 3 Hasil Rata-rata Pengujian *Mother Wavelet* terhadap Parameter Performa

<i>Mother Wavelet</i>	PSNR	MSE	BER	SSIM
db1	141,118	0,048	0,029	0,823
db2	137,723	0,07	0,024	0,865
db3	140,573	0,05	0,046	0,795

4.5. Skenario Pengujian terhadap Serangan

Pada skenario ini akan dilakukan analisis ketahanan sistem terhadap dua jenis serangan, yang pertama berupa *noise salt and pepper*, dan yang kedua berupa *noise gaussian blur*. *Noise salt and pepper* adalah *noise* berupa titik-titik berwarna merah, hijau, dan biru pada citra *host* RGB. *Noise gaussian blur* adalah serangan dengan memberikan *white noise* pada citra *host* yang terdistribusi secara *gaussian blur*. Ada dua citra *host* beresolusi 800x800 yang telah disisipkan watermark beresolusi 64x64 yang akan diujikan. Perbandingan hasil rekonstruksi citra watermark tanpa diberi serangan dan yang telah diberi serangan, serta hasil nilai parameter, dapat dilihat di bawah.

Tabel 4 Hasil Pengujian Serangan terhadap Parameter Performa

Jenis Serangan	Citra Host	PSNR	MSE	BER	SSIM
Tanpa Serangan	Host 1	138,223	0,064	0,027	0,806
	Host 2	140,67	0,05	0,023	0,872
<i>Salt and Pepper</i>	Host 1	138,702	0,065	0,291	0,252
	Host 2	141,119	0,048	0,22	0,306
<i>Gaussian Blur</i>	Host 1	138,881	0,06	0,273	0,29
	Host 2	142,118	0,043	0,459	0,001

Tabel 5 Perbandingan Hasil Watermark Tanpa Serangan & Dengan Serangan

Jenis Serangan	Citra Host	Output Citra Watermark
Tanpa Serangan	Host 1	
	Host 2	
<i>Salt and Pepper</i>	Host 1	
	Host 2	
<i>Gaussian Blur</i>	Host 1	
	Host 2	

5. Kesimpulan dan Saran

5.1. Kesimpulan

Pada bab sebelumnya telah dilakukan empat skenario pengujian. Hasil dari pengujian tersebut berupa parameter performa, yaitu PSNR, MSE, BER, dan SSIM. Berdasarkan hasil-hasil tersebut, dapat ditarik kesimpulan sebagai berikut:

1. *Compressive sensing* pada proses *image watermarking* menggunakan metode DWT-SVD berhasil diimplementasikan dengan nilai parameter yang memuaskan yaitu PSNR sebesar 140,625 dB, MSE sebesar 0,05, BER sebesar 0,034, dan SSIM sebesar 0,8.
2. Pada skenario pengujian *sub-band*, nilai BER dan SSIM terbaik diperoleh *sub-band* LH, sedangkan pada skenario pengujian *mother wavelet*, nilai BER dan SSIM terbaik diperoleh *mother wavelet* db2. Maka, untuk hasil rekonstruksi citra watermark terbaik dipilih *sub-band* LH dan *mother wavelet* db2.
3. Semakin besar *measurement rate*, semakin baik kualitas rekonstruksi citra watermark karena *measurement rate* berbanding terbalik dengan rasio kompresi citra.
4. Ketahanan sistem terhadap jenis serangan *noise salt and pepper* dinilai baik karena hasil rekonstruksi watermark pada kedua *host* menghasilkan nilai BER < 0,3. Jenis serangan *noise gaussian blur* pada *host* pertama menghasilkan nilai BER yang baik, yaitu < 0,3. Namun pada *host* kedua, nilai BER yang dihasilkan > 0,3. Serangan *noise gaussian blur* cukup berdampak buruk pada *host* kedua.

5.2. Saran

Dalam penelitian ini terdapat keterbatasan. Ada beberapa saran untuk mengembangkan penelitian ini lebih lanjut. Saran ini meliputi resolusi citra, jenis citra, dan algoritma rekonstruksi lainnya yang bisa digunakan. Saran-saran tersebut diantaranya:

1. Resolusi citra *host* dan citra watermark yang digunakan pada penelitian ini tidak begitu besar, maka untuk penelitian lebih lanjut, gunakan resolusi citra *host* dan citra watermark yang lebih besar.
2. Lakukan penelitian lebih lanjut dengan citra watermark RGB.
3. Bandingkan penggunaan algoritma rekonstruksi BPDN pada sistem yang digunakan pada penelitian ini dengan algoritma rekonstruksi yang lainnya, seperti *Orthogonal Matching Pursuit*, *L1-Magic* dan yang lainnya.

REFERENSI

- [1] Potdar, V., Han, S., Chang, E. 2005. "A Survey of Digital Image Watermarking Techniques," 2005 3rd IEEE International Conference on Industrial Informatics (INDIN) pp. 709-716.
- [2] Luo, W., Huang, J., Qiu, G. 2010. "JPEG Error Analysis and Its Applications to Digital Image Forensics," IEEE Transactions on Information Forensics and Security, Vol. 5, no. 3 pp. 480-491.
- [3] Andono, Pulung N., T. Sutojo, dan Muljiono. 2017. Pengolahan Citra Digital. Yogyakarta: Penerbit ANDI.
- [4] Foucart, S., Rauhut, H. 2013. "An Invitation to Compressive Sensing," Springer Science Business Media New York.

- [5] Rani, M., Dhok, S. B., Deshmukh, R. B. 2018. "A Systematic Review of Compressive Sensing: Concepts, Implementations, and Applications" *IEEE Access*, Vol. 6, pp. 4875-4894.
- [6] Ruswiansari, M., Novianti, A., Wirawan. 2016. "Implementasi Discrete Wavelet Transform (DWT) dan Singular Value Decomposition (SVD) pada Image Watermarking," *Jurnal Elektro Telekomunikasi Terapan*.
- [7] Lai, C., Tsai, C. 2010. "Digital Image Watermarking Using Discrete Wavelet Transform and Singular Value Decomposition," *IEEE Transactions on Instrumentation and Measurement*, Vol. 59, no. 11 pp. 3060-3063.
- [8] Voloshynovskiy, S., Pereira, S., Pun, T., Eggers, Joachim J., Su, Jonathan K. 2001. "Attacks on Digital Watermarks: Classification, Estimation-Based Attacks, and Benchmarks," *IEEE Communications Magazine* pp. 118-126.