

Digital Watermarking Image dengan Menggunakan Discrete Wavelet Transform dan Singular Value Decomposition (DWT-SVD) untuk Copyright Labeling

Nur Anisah

Jurusan Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami 36A Kentingan
Surakarta 57126

nnur.anisah@gmail.com

Bambang Harjito

Jurusan Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami 36A Kentingan
Surakarta 57126

bambang_harjito@staff.uns.ac.id

Esti Suryani

Jurusan Informatika
Universitas Sebelas Maret
Jl. Ir. Sutami 36A Kentingan
Surakarta 57126

suryapalapa@yahoo.com

ABSTRAK

Penyebaran konten digital yang begitu mudah membuat pemberian identitas pada konten digital menjadi sangat penting. Konten digital yang tidak beridentitas membuatnya mudah diklaim oleh pihak yang tidak bertanggungjawab. Watermarking merupakan salah satu cara pemberian watermark yang mudah dan tanpa merusak konten yang disisipi.

Teknik watermarking pada image digital dilakukan dengan mentransformasikan image dengan transformasi domain frekuensi Discrete Wavelet Transform (DWT). Pengolahan image input seperti cover medium dan watermark sebelumnya diperlukan agar cover medium dapat disisipi. Pada makalah ini, DWT dipadukan dengan faktorisasi matrik Singular Value Decomposition (SVD) untuk Copyright Labeling yang bertujuan agar hasil watermarking lebih robust.

Hasil penelitian menunjukkan bahwa teknik watermarking dengan DWT-SVD mampu menghasilkan watermark yang tahan terhadap beberapa serangan seperti Gaussian Noise, Salt & Pepper Noise, dan Poisson Noise tetapi tidak tahan terhadap serangan blurring dan geometrical attacks (rotation dan cropping).

Kata Kunci : Image Digital, Discrete Wavelet Transform, Singular Value Decomposition, Watermarking

1. Pendahuluan

Kemudahan penyebaran konten digital di dunia maya membuat konten rawan terhadap pembajakan. Konten digital seperti digital image memerlukan suatu identitas yang melekat pada konten agar tidak mudah diklaim oleh pihak lain yang tidak bertanggung jawab. Identitas yang melekat pada kontendigital image disebut dengan watermark image

Copyright labeling adalah salah satu aplikasi dari watermarking digital [1]. Watermarking dilakukan dengan cara menyisipkan watermark atau informasi ke dalam cover medium yang menghasilkan media yang terwatermark. Watermark dapat berfungsi sebagai bukti otentikasi kepemilikan konten. Watermarking dapat dilakukan berbagai media seperti media digital.

Digital Watermarking adalah proses penyisipan informasi ke media digital yang tak terlihat oleh mata manusia tetapi dapat terdeteksi oleh komputer [2]. Cover medium dan watermark dapat berupa teks, image digital, audio, dan video. Sedangkan watermarking pada image digital disebut Digital Watermarking Image.

Digital Watermarking Image bekerja pada dua domain pengolahan image digital, yaitu domain spasial dan domain frekuensi [3]. Transformasi domain spasial bekerja dengan

mengubah nilai piksel pada image digital sedangkan transformasi domain frekuensi menyisipkan watermark pada koefisien frekuensi hasil transformasi image asli. Teknik ini lebih rumit dari teknik yang berkerja pada domain spasial, tetapi memiliki ketahanan yang lebih baik. Beberapa teknik yang memanfaatkan domain frekuensi adalah Fast Fourier Transform (FFT), Discrete Fourier Transform (DFT), Discrete Cosinus Transform (DCT), dan Discrete Wavelet Transform (DWT).

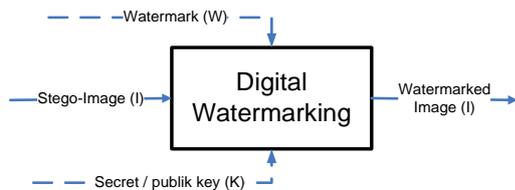
Beberapa jurnal terkait watermarking dengan DWT-SVD antara lain penelitian yang dilakukan oleh Dean Fatony Alfatawa [4]. Dean mencoba menerapkan Discrete Wavelet Transform pada image hitam putih (binary image). Sayangnya, image terwatermark hanya tahan terhadap beberapa serangan seperti perubahan format file image dari PNG ke BMP. Anumol dan Karthigaikumar [5] menerapkan Discrete Wavelet Transform pada image RGB yang dikonversi ke HSV (Hue, Saturation, and Value) terlebih dahulu. Hasil penelitian menunjukkan bahwa watermark yang disisipkan bisa diekstrak kembali. Terakhir penelitian yang dilakukan oleh Praful Saxena dan rekan-rekan [6]. Praful menggabungkan teknik transformasi DWT dan Singular Value Decomposition untuk penyisipan watermark pada image grayscale. Pada penelitian ini watermark disisipkan pada subband HH. Image terwatermark dapat diekstraksi dengan baik setelah diberi serangan-serangan tertentu.

Mengacu pada beberapa jurnal di atas, penelitian ini menggunakan Image Digital dengan model warna RGB yang dikonversi ke grayscale kemudian baru ditransformasikan dengan Discrete Wavelet Transform (DWT) dan faktorisasi matrik pada subband yang akan disisipi watermark dengan Singular Value Decomposition (SVD).

2. Teknik Digital Watermarking Image

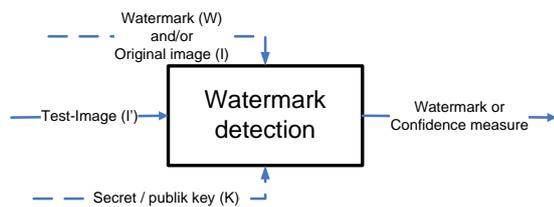
Watermarking merupakan proses pemberian tanda air yang berupa data ke dalam suatu media [1]. Watermark bisa disisipkan di berbagai media seperti media digital. Penyisipan informasi yang dilakukan di media image digital disebut Image Digital Watermarking [2].

Teknik watermarking secara umum dibagi menjadi dua tahap, yaitu: penyisipan watermark (embedding atau encoding) dan recovery watermark (ekstraksi atau decoding). Pemberian kunci bertujuan agar watermark hanya dapat diekstraksi oleh pihak yang sah [2]. Berikut proses encoding watermark dan pemberian kunci pada image digital:



Gambar 1 Proses *encoding image* digital (M. Kutter dan F. A. P. Petitcolas, 2000)

Encoding merupakan proses penyisipan informasi ke dalam *image digital* sedangkan *decoding* bertujuan untuk mengungkap kembali *watermark* yang ditanam dalam *image digital*. Proses *decoding* perlu adanya verifikasi *watermark* untuk membuktikan status kepemilikan *image digital*. Verifikasi *watermark* dilakukan dengan cara mencocokkan kunci yang diinputkan dengan kunci yang ada pada *image*. Berikut proses ekstraksi *image watermark*:



Gambar 2 Proses *decoding Image* Digital (M. Kutter dan F. A. P. Petitcolas, 2000)

Kedua proses tersebut saling berhubungan, untuk selanjutnya proses *encoding* disebut sebagai proses penyisipan dan proses *decoding* disebut sebagai proses ekstraksi.

3. Discrete Wavelet Transform dalam Image Digital

Dalam pengolahan *image digital*, diperlukan proses transformasi *image* agar informasi yang tertentu yang terkandung dalam *image* dapat diekstrak [7]. Transformasi *image* dibagi menjadi dua domain, yaitu domain spasial dan domain frekuensi. Domain spasial berkerja dengan melibatkan piksel-piksel pada *image digital*, sedangkan domain frekuensi dilakukan transformasi untuk mengetahui koefisien pada *image digital* tersebut sehingga proses penyisipan dapat dilakukan. Penelitian ini menggunakan transformasi dalam domain frekuensi yaitu Wavelet yang diterapkan pada *image 2* dimensi sebagai metode transformasi *image* digital.

Transformasi Wavelet digunakan untuk menganalisa suatu bentuk sinyal (gelombang) yang merupakan kombinasi dari waktu (skala) dan frekuensi. Wavelet memiliki dua fungsi utama yaitu penskalaan dan *filtering*.

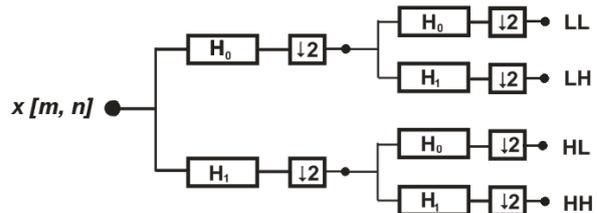
Proses *filtering* pada *image digital* bertujuan untuk mendapatkan informasi tertentu dari sinyal *image*. *Filter bank* mentransformasikan suatu sinyal *image* yang menghasilkan data berupa koefisien yang biasa disebut tapis *low pass* dan *high pass* yang kemudian dapat menentukan subimage dari *image* tersebut. Sedangkan fungsi penskalaan menghasilkan *Mother Wavelet*, Berikut persamaan dari fungsi penskalaan ϕ :

Wavelet paling tua yang lahir dari hasil penskalaan, dilasi, dan transformasi *Mother Wavelet* [7]. Dari *mother wavelet*, lahir Haar Wavelet dan kemudian disusul Wavelet lain seperti Daubechies. Beberapa transformasi Wavelet yang sering digunakan saat ini adalah *Discrete Wavelet Transform* dan *Continous Wavelet Transform*. DWT digunakan pada transformasi *image* digital, sedangkan CWT digunakan pada *image* yang bergerak, seperti video.

Konsep transformasi Wavelet pada *image 2* dimensi sangat sederhana. *Image* ditransformasi bagi (dekomposisi)

terlebih dahulu menjadi 4 subimage yaitu LL, LH, HL, dan HH. Setiap *subimage* berukuran seperempat kali *image* asli. *Subimage* LL yang terletak di kiri atas merupakan *subimage* yang paling halus (*smooth*) dari *image* asli. Sedangkan *subimage* lain, seperti HL, LH, dan HH mengandung frekuensi tinggi dari *image* sehingga dapat disebut versi kasar dari *image* asli.

Proses transformasi dimulai dengan proses dekomposisi. Dekomposisi adalah proses perubahan menjadi bentuk yang lebih sederhana atau penguraian. Dalam transformasi domain frekuensi, dekomposisi berarti proses perubahan cira digital menjadi koefisien frekuensi. Karena *image* yang didekomposisikan 2 dimensi, proses dekomposisi dibagi menjadi dua tahap, yaitu dekomposisi baris dan dekomposisi kolom.

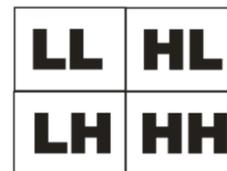


Gambar 3 Transformasi Wavelet 2D 1 level (Darma Putra, 2010)

Hasil dekomposisi *image 1* level menghasilkan 4 koefisien, yaitu:

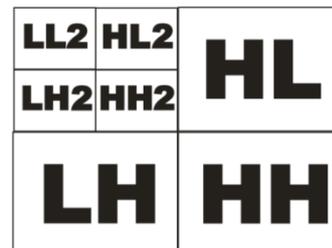
- LL diperoleh melalui proses tapis *Low pass* dilanjutkan dengan *Low pass*. *Image* dibagian ini mirip dengan *image* aslinya dan lebih halus.
- LH merupakan koefisien yang didapat dari proses tapis *Low pass* yang kemudian dilanjutkan dengan *High pass*.
- HL merupakan hasil dari proses tapis *High pass* kemudian dilanjutkan *Low pass*.
- HH adalah hasil dari proses tapis *High pass* kemudian *High pass* lagi.

Hasil transformasi wavelet digambarkan dalam skema berikut ini:



Gambar 4 Skema Transformasi wavelet 2D 1 level (Darma Putra, 2010)

Skema diatas menggambarkan transformasi wavelet 2 dimensi level 1. Sedangkan untuk level 2, koefisien *image* pada bagian LL akan didekomposisikan lagi menjadi empat bagian, yaitu LL2, LH2, HL2, dan HH2, begitu seterusnya pada level-level berikutnya. Berikut skema transformasi wavelet 2 level:



Gambar 5 Skema Transformasi wavelet 2D 2 level (Darma Putra, 2010)

4. Singular Value Decomposition (SVD)

SVD merupakan suatu pemfaktoran matrik dengan menguraikan suatu matrik ke dalam dua matrik P dan Q. Jika diketahui matrik A berukuran $m \times n$ dengan rank $r > 0$, maka dekomposisi dari matrik A dirumuskan sebagai berikut:

$$A = P\Delta Q^T$$

Keterangan:

P = matriks ortogonal berukuran $m \times r$

Q = matriks orthogonal berukuran $n \times r$

Δ = matriks diagonal berukuran $r \times r$

Terbentuknya matrik Δ tergantung kondisi matrik A, yaitu:

- a. Δ bila $r = m = n$
- b. $\begin{bmatrix} \Delta \\ (0) \end{bmatrix}$ bila $r = n$ dan $r < m$
- c. $\begin{bmatrix} \Delta & (0) \end{bmatrix}$ bila $r = m$ dan $r < n$
- d. $\begin{bmatrix} \Delta & (0) \\ (0) & (0) \end{bmatrix}$ bila $r < m$ dan $r < n$

Matrik P diperoleh melalui perkalian A, Q, dan Δ^{-1} dapat dinyatakan sebagai berikut:

$$P = A\Delta Q^{-1}$$

5. PSNR dan NC

Kualitas *image* yang disisipi *watermark* biasanya tidak berubah banyak. Perbedaan antara *image* ber*watermark* dengan *image* asli dan *watermark* asli dengan *watermark* hasil ekstraksi tidak dapat dilihat dengan mata telanjang. Untuk itu, diperlukan pengujian secara subjektif yaitu dengan menghitung nilai PSNR dan NC pada kedua *image* yang dibandingkan tersebut.

5.1 Peak Signal to Noise Ratio (PSNR)

Peak Signal to Noise Ratio adalah perbandingan antara nilai maksimum dari sinyal yang diukur dengan besarnya *noise* yang berpengaruh pada *image* tersebut [7]. PSNR digunakan untuk mengetahui perbandingan antara *image* asli sebelum dan sesudah disisipi *watermark*. Untuk mendapatkan nilai PSNR, ditentukan nilai MSE (*Mean Square Error*) terlebih dahulu. MSE adalah nilai error kuadrat rata-rata antara *image* asli dengan *image* ber*watermark*. Berikut cara menghitung nilai MSE:

$$MSE = \frac{1}{MN} \sum_{y=1}^M \sum_{x=1}^N [I(x,y) - I'(x,y)]^2$$

Keterangan:

MSE = Nilai *Mean Square Error*

M = panjang *image watermark* (dalam piksel)

N = lebar *image watermark* (dalam piksel)

$I(x,y)$ = nilai piksel dari *image* asli

$I'(x,y)$ = nilai piksel pada *image watermark*

Setelah diperoleh nilai MSE, nilai PSNR dapat dihitung dengan rumus sebagai berikut:

$$PSNR = 10 \cdot \log \left(\frac{MAX_i^2}{MSE} \right)$$

Keterangan:

MSE = nilai *Mean Square Error*

MAX_i = nilai maksimum dari piksel *image* yang digunakan.

Nilai MSE menunjukkan banyaknya error dari *image*. Semakin rendah nilai MSE maka akan semakin baik. Sedangkan nilai PSNR yang didapat dari nilai maksimum piksel dibagi nilai MSE, sehingga semakin besar nilai PSNR

yang dihasilkan, menunjukkan kualitas *image* semakin baik. *Image* 8-bit dikatakan berkualitas baik jika rentan nilai PSNR antara 30-50 dB.

5.2 Normalized Correlation Coefficient (NC)

NC atau nilai kolerasi dapat digunakan juga untuk mendeteksi kehadiran *watermark*. Rumus nilai kolerasi adalah sebagai berikut:

$$NC(W, W') = \frac{\sum W W'}{\sqrt{\sum W_i^2} \sqrt{\sum W_i'^2}}$$

Keterangan:

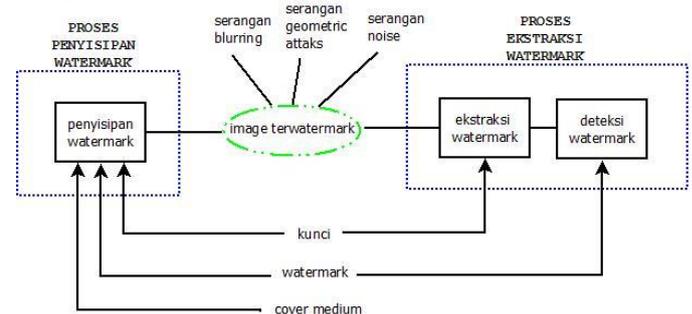
W = *watermark* asli

W' = *watermark* hasil ekstraksi

Nilai kolerasi menunjukkan perbandingan antara *watermark* asli dengan *watermark* hasil ekstraksi. Nilai kolerasi bernilai 0 (nol) berarti kedua *image* yang dibandingkan tidak memiliki kemiripan sama sekali, sedangkan kolerasi yang bernilai 1 (satu) menunjukkan kedua *image* yang dibandingkan sama persis.

6. Implementasi Sistem

Implementasi sistem merupakan inti proses dalam penelitian. Langkah - langkah implementasi sistem ditampilkan pada gambar berikut ini:



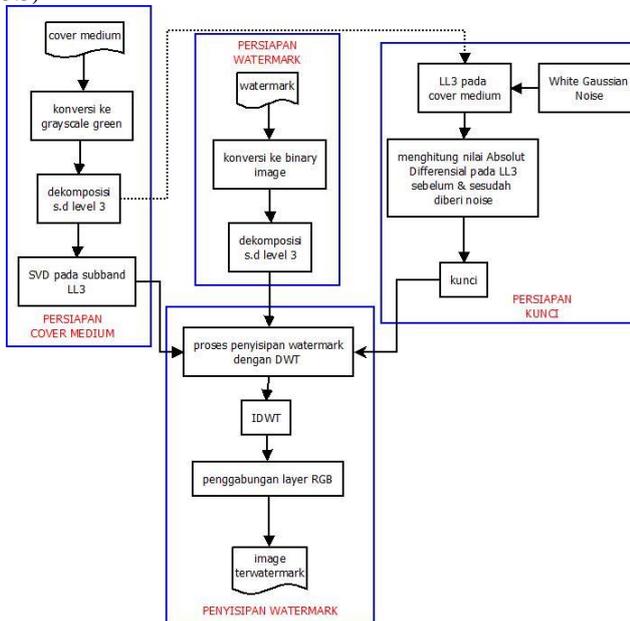
Gambar 6 Diagram alir implementasi

Proses implementasi sistem terdiri dari dua proses dasar, yaitu proses penyisipan *watermark* dan ekstraksi *watermark*. Penyisipan *watermark* merupakan proses penyisipan *image watermark* ke *image* asli sedangkan ekstraksi *watermark* adalah proses mengeluarkan *watermark* yang ada didalam *image* asli. Antara kedua proses tersebut terdapat proses lain seperti serangan *image terwatermark* dan deteksi *watermark*.

Serangan *image* dilakukan dengan memberikan serangan pada *image terwatermark* yang berupa *noise*, *geometrical attacks*, *blurring*, dan sebagainya. Serangan ini bertujuan untuk menguji ketahanan *watermark* yang ada di dalam *image*. Selain serangan *image terwatermark*, proses deteksi *watermark* juga diperlukan untuk menentukan apakah *watermark* berhasil diekstraksi dengan baik dan layak untuk ditampilkan. Penentuan keberhasilan ekstraksi dilakukan dengan membandingkan nilai kolerasi dan *threshold*. *Watermark* dengan nilai kolerasi dibawah nilai *threshold* bisa dikatakan tidak berhasil diekstraksi dengan baik.

6.1 Proses Penyisipan Watermark

Proses penyisipan watermark terdiri dari persiapan image asli dan watermark yang akan menjadi input perangkat lunak, dan proses penyisipan watermark. Proses penyisipan watermark digambarkan pada skema berikut ini: (lihat gambar 3.3)



Gambar 7 Diagram Proses Penyisipan Watermark

Image asli dengan model warna RGB perlu diolah terlebih dahulu agar bisa disisipi watermark. Proses pengolahan terdiri dari konversi dari image berwarna menjadi grayscale dan dekomposisi image. Konversi ke grayscale dilakukan dengan memisahkan image asli ke dalam tiga layer warna yaitu red, green, dan blue sehingga terbentuk tiga image grayscale dari masing-masing warna red, green, dan blue. Tetapi yang digunakan dalam penyisipan watermark hanya image grayscale layer green saja.

Proses selanjutnya adalah proses transformasi. Image asli yang ditransformasikan dengan Discrete Wavelet Transform dan Singular Value Decomposition kemudian disisipi watermark dan terbentuklah image terwatermark. Algoritma penyisipan watermark adalah sebagai berikut:

Algoritma 1

Input : cover medium, watermark, kunci

Output : image terwatermark

Proses :

1. Pertama konversi cover medium ke grayscale green dengan perintah: `image=image(:, :2);`
2. Dekomposisi image layer green dengan DWT sampai level 3. Dekomposisi level 1 menghasilkan 4 subband, yaitu: LL, HL, LH, HH dengan perintah: `[LL, LH, HL, HH] = dwt2(image, 'haar');` Dekomposisikan dengan DWT level 2, subband LL didekomposisikan menjadi subband LL2, HL2, LH2, HH2, HL, LH, HH dengan perintah `[LL2, LH2, HL2, HH2] = dwt2(LL, 'haar');` Kemudian pada level 3 subband LL2 didekomposisikan lagi hingga menghasilkan subband LL3 dengan perintah: `[LL3, LH3, HL3, HH3] = dwt2(LL2, 'haar');`
3. Faktorisasi subband LL3 yang berupa matrik image dengan menggunakan SVD. Rumus SVD adalah $A = UDV^T$ dengan A = subband LL3

Keterangan:

A = matrik ukuran (m x n)

U = singular vektor dari matrik A (vektor ini orthonormal)

D = diagonal vektor yang menyimpan singular value dari koresponding singular vektornya

V^T = singular vector dari matrik A yang juga orthonormal (T berarti transpose)

Sehingga menjadi 3 matriks yaitu matriks U, D, dan V $svd(LL3) = [U, D, V];$

4. Pembuatan kunci dilakukan dengan mengambil subband LL3 kemudian dilakukan Add White Gaussian Noise dengan perintah: `n = awgn(LL3, 1, 0, 'linear');` `n = LL3` setelah diberi White Gaussian Noise
5. Matrik yang telah diberi White Gaussian Noise kemudian dilakukan perhitungan perbandingan nilai absolute pada LL3 sebelum dan sesudah diberi noise dengan perintah: `N = imabsdiff(n, LL3);` N = nilai absolute differensial
6. Watermark yang akan disisipi dikonversi ke binary image terlebih dahulu dengan perintah `watermark=rgb2gray(wmark);`
7. Watermark yang telah menjadi binary image kemudian didekomposisikan 3 level dengan DWT dengan perintah: Dekomposisi level 1: `[L_L, L_H, H_L, H_H] = dwt2(watermark, 'haar');` Dekomposisi level 2: `[L_L2, L_H2, H_L2, H_H2] = dwt2(L_L, 'haar');` Dekomposisi level 3: `[L_L3, L_H3, H_L3, H_H3] = dwt2(L_L2, 'haar');`
8. Melakukan SVD pada matriks L_L3 watermark dengan cara yang sama dengan SVD pada matriks LL3 sebelumnya sehingga menghasilkan `Svd(L_L3)=[U_wmark, D_wmark, V_wmark];`
9. Membuat matriks LL3 yang baru (setelah proses SVD) dengan cara mengganti matriks D dengan matriks D gabungan dari matriks D cover medium dan watermark seperti berikut: `D_watermark = D+(0.01*D_wmark);` `LL3 = U*D_watermark*transpose(V);`
10. Watermark yang telah berbentuk L_L3 disisipi di subband LL2 yang telah diolah dengan menggunakan rumus

$$f' = f + \alpha \cdot w(k) + K(k), k = 1, \dots, L$$

f' = image terwatermark

f = cover medium

w = watermark

α = nilai ambang yang menunjukkan tingkat kekuatan penyisipan watermark

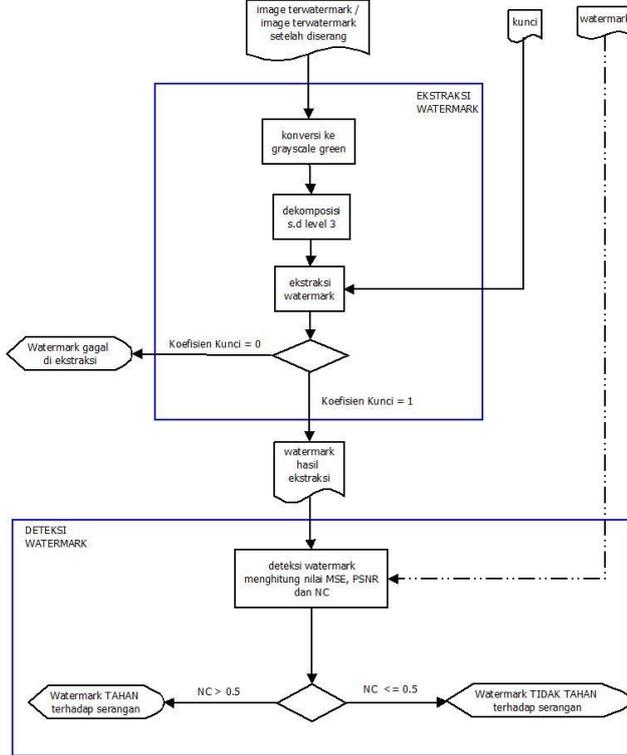
K = kunci

k = (m,n) posisi pada koefisien subband LL2

11. Proses IDWT 3 level sehingga menghasilkan image terwatermark dengan perintah Untuk IDWT level 1: `imagerwatermark_level1=idwt2(imagerwatermark, LH3, HL3, HH32, 'haar');` Untuk IDWT level 2: `Imagerwatermark_level2=idwt2(imagerwatermark_level1, LH2, HL2, HH2, 'haar');` Untuk IDWT level 3: `Imagerwatermark_final=idwt2(imagerwatermark_level2, LH, HL, HH, 'haar');`
12. Penggabungan image layer green dengan layer red dan blue dengan perintah: `newColorImage(:, :, 1) =wimg;` `newColorImage(:, :, 2) =wimg2;` `newColorImage(:, :, 3) =wimg3;` `newColorImage = image RGB hasil penggabungan`
13. Setelah proses penggabungan menghasilkan output image terwatermark

6.2 Proses Ekstraksi Watermark

Proses ekstraksi watermark terdiri dari pengolahan image terwatermark agar siap diekstraksi, proses ekstraksi itu sendiri dan deteksi watermark. Proses ekstraksi watermark digambarkan dengan skema berikut ini:



Gambar 8 Diagram Proses Ekstraksi Watermark

Image terwatermark tidak dapat langsung diekstraksi, diperlukan proses konversi image dari image berwarna (true color) ke grayscale green dan dekomposisi dengan Haar Wavelet terlebih dahulu. Algoritma ekstraksi watermark adalah sebagai berikut:

Algoritma 2

Input : image terwatermark (image terwatermark yang telah diserang), kunci

Output : watermark hasil ekstraksi

Proses :

- Konvensi image terwatermark ke grayscale green dengan perintah:
image=image(:, :2);
- Dekomposisi image layer green dengan DWT sampai level 3. Dekomposisi level 1 menghasilkan subband LL, HL, LH, HH dengan perintah sebagai berikut:
[LL, LH, HL, HH] = dwt2(image, 'haar');
- Dekomposisi level 2 dari subband LL menghasilkan subband LL2, HL2, LH2, HH2, HL, LH, HH dengan perintah
[LL2, LH2, HL2, HH2] = dwt2(LL, 'haar');
- Kemudian pada level 3 subband LL2 didekomposisikan yang menghasilkan subband LL3 dengan perintah:
[LL3, LH3, HL3, HH3] = dwt2(LL2, 'haar');
- Sebelum proses ekstraksi, dilakukan pencocokan kunci dengan mencocokkan nilai besaran scalar pada kunci yang dimasukkan dengan nilai besaran scalar pada cover medium.
- Proses ekstraksi dilakukan dengan mengeluarkan watermark dari cover medium. Ekstraksi dilakukan dengan perintah:
recovered_image = ((LL2' - LL2) - k) / 0.001;

Keterangan:

LL2' : image terwatermark

LL2 : cover medium

k : matriks kunci

- Proses deteksi watermark dengan melakukan perhitungan PSNR, MSE, dan NC untuk menentukan apakah watermark berhasil diekstraksi atau gagal diekstraksi. Jika watermark memiliki nilai NC ≥ 0.5 maka dikatakan watermark berhasil diekstraksi, sedangkan jika nilai NC < 0.5, watermark gagal diekstraksi.

$$NC(W, W') = \frac{\sum W(n)W'(n)}{\sqrt{\sum W^2(n)}\sqrt{\sum W'^2(n)}}$$

Keterangan:

w = watermark

w' = watermark hasil ekstraksi

- Membandingkan nilai kolerasi image watermark (NC) dengan nilai threshold, nilai threshold = 0.5.
- Jika NC < T, berarti watermark berhasil diekstraksi dengan baik, sedangkan jika nilai NC ≥ T, watermark tidak dapat diekstraksi dengan baik dan dianggap tidak layak untuk ditampilkan di monitor.

7. HASIL DAN PEMBAHASAN

Pengujian yang dilakukan dalam tiga macam. Pertama cover medium disisipi watermark dengan berbagai ukuran dan format file, pengujian ini dijadikan acuan untuk analisis pada pengujian yang lain. Pengujian kedua adalah menyisipkan watermark ke cover medium dengan berbagai level dekomposisi, sedangkan pengujian ketiga dilakukan dengan memberi berbagai serangan pada image terwatermark.

Penelitian ini dilakukan dengan cover medium lena.jpg yang berukuran (512 x 512 piksel) dengan beberapa watermark dalam berbagai ukuran dan format file, yaitu uns.png yang berukuran (400 x 400 piksel), dogs.jpg yang berukuran (640 x 480 piksel), dan pepper.bmp yang berukuran (512 x 512 piksel).

Pengujian dilakukan untuk mendapatkan nilai PSNR, MSE dan NC (Normalized Coefficient Correlation) untuk mengetahui kualitas image watermark hasil ekstraksi. Analisa dari ketiga pengujian dideskripsikan pada subbab-subbab berikut ini:

7.1 Analisis Hasil Ekstraksi Image dengan Berbagai Ukuran dan Format File Watermark

Pengujian dilakukan menggunakan cover medium lena.jpg dengan 3 macam watermark, yaitu uns400.png, dogs.jpg, dan pepper.bmp. Hasil pengujian ditampilkan pada tabel 1.

Tabel 1 Hasil Ekstraksi Image terwatermark

Watermark Asli	Watermark setelah dikonversi ke Binary Image	Watermark Hasil Ekstraksi	PSNR (dB), MSE, dan NC
 uns400.png (400 x 400 piksel)			PSNR: 48.9434 MSE: 0.829346 NC: 0.889784
 dogs.jpg (640x480 piksel)			PSNR: 54.152 MSE: 0.249967 NC: 0.959068

			PSNR: 51.4941 MSE: 0.460964 NC: 0.97296
<i>pepper.bmp</i> (512x512 piksel)			

Nilai-nilai yang ditunjukkan pada tabel 1 merupakan nilai dasar sebagai acuan pada pengujian-pengujian berikutnya. Nilai NC tertinggi diperoleh saat watermark *pepper.bmp* dengan format file *.bmp*. Watermark dengan tingkat kemiripan tertinggi dengan watermark asli dikarenakan ukuran file watermark yang sama dengan ukuran file *cover medium*, yaitu (512 x 512 piksel).

7.2 Analisis Hasil Ekstraksi Image dengan Berbagai Level Dekomposisi

Pengujian dilakukan pada file *cover medium* dengan disisipi 3 macam file watermark secara bergantian yang diuji pada beberapa level dekomposisi, yaitu dekomposisi level 1, level 2, dan level 3. Hasil Pengujian ditampilkan pada Tabel 2.

Tabel 2 Hasil Pengujian Watermark dengan Berbagai Level Dekomposisi

Dekomposisi	File Watermark	Image Hasil Ekstraksi	PSNR (dB), MSE, dan NC
Level 1	<i>uns400.png</i> (400 x400 piksel)		PSNR: 48.9385 MSE: 0.830297 NC: 0.977922
Level 2	<i>uns400.png</i> (400 x400 piksel)		PSNR: 48.9402 MSE: 0.829974 NC: 0.947448
Level 3	<i>uns400.png</i> (400 x400 piksel)		PSNR: 48.9434 MSE: 0.829346 NC: 0.889784
Level 1	<i>dogs.jpg</i> (640x480 piksel)		PSNR: 54.1472 MSE: 0.250244 NC: 0.991549
Level 2	<i>dogs.jpg</i> (640x480 piksel)		PSNR: 54.1488 MSE: 0.25015 NC: 0.975812
Level 3	<i>dogs.jpg</i> (640x480 piksel)		PSNR: 54.152 MSE: 0.249967 NC: 0.959068
Level 1	<i>pepper.bmp</i> (512x512 piksel)		PSNR: 51.4891 MSE: 0.461497 NC: 1
Level 2	<i>pepper.bmp</i> (512x512 piksel)		PSNR: 51.4909 MSE: 0.461312 NC: 1
Level 3	<i>pepper.bmp</i> (512x512 piksel)		PSNR: 51.4941 MSE: 0.460964 NC: 0.97296

Tabel 2 menunjukkan hasil pengujian watermark hasil ekstraksi pada beberapa level dekomposisi. Dari ketiga image pola perubahan nilai NC, MSE dan PSNR hampir sama. Nilai NC tertinggi didapat pada dekomposisi level 1, diikuti dengan nilai-nilai pada dekomposisi level 2 dan nilai NC terendah pada dekomposisi level 3. Image dengan level dekomposisi terendah memiliki tingkat kemiripan tertinggi dengan watermark asli.

Nilai PSNR tertinggi diperoleh pada level 3, semakin sedikit level dekomposisi yang dijalankan, semakin kecil nilai PSNR yang didapat. Sedangkan nilai MSE tertinggi didapat pada image level dekomposisi terendah, yaitu level 1. Nilai MSE menurun seiring dengan meningkatnya level dekomposisi. Hal ini menunjukkan image dengan level

dekomposisi tertinggi memiliki tingkat error terendah.

7.3 Analisis Hasil Ekstraksi Image dengan Berbagai Serangan

Pengujian selanjutnya dilakukan dengan memberi beberapa serangan terhadap image terwatermark secara bergantian dan menguji kualitas image watermark hasil ekstraksinya. Serangan yang dilakukan antara lain pemberian noise, geometrical attacks dan blurring.

7.3.1 Gaussian Noise

Gaussian Noise merupakan noise yang terdiri dari titik-titik putih dengan dua macam nilai input yaitu nilai rata-rata dan variasi. Semakin besar nilai rata-rata dan variasinya maka image yang terlihat semakin kabur begitu pula sebaliknya. Pengujian dilakukan dengan berbagai variasi nilai rata-rata (M) dengan nilai variasi (V) tetap, yaitu (1 x 10⁻⁷). Hasil pengujian ditampilkan pada tabel 3.

Tabel 3 Hasil Pengujian Serangan Gaussian Noise

Watermark	Gaussian Noise V: (1 x 10 ⁻⁷)	Image terwatermark	Watermark Hasil Ekstraksi	PSNR (dB), MSE dan NC
<i>uns400.png</i> (400 x400 piksel)	M: (1 x 10 ⁻⁸)			PSNR: 48.9504 MSE: 0.828019 NC: 0.832774
<i>uns400.png</i> (400 x400 piksel)	M: (1 x 10 ⁻⁵)			PSNR: 48.9504 MSE: 0.828025 NC: 0.0764321
<i>uns400.png</i> (400 x400 piksel)	M: 0.1			PSNR: 49.0274 MSE: 0.81347 NC: 0.0218132
<i>dogs.jpg</i> (640x480 piksel)	M: (1 x 10 ⁻⁸)			PSNR: 54.1589 MSE: 0.249567 NC: 0.918677
<i>dogs.jpg</i> (640x480 piksel)	M: (1 x 10 ⁻⁵)			PSNR: 54.1587 MSE: 0.249582 NC: 0.0993994
<i>dogs.jpg</i> (640x480 piksel)	M: 0.01			PSNR: 54.1651 MSE: 0.249214 NC: 0.0458374
<i>pepper.bmp</i> (512x512 piksel)	M: (1 x 10 ⁻⁸)			PSNR: 51.5011 MSE: 0.460227 NC: 0.942206
<i>pepper.bmp</i> (512x512 piksel)	M: (1 x 10 ⁻⁵)			PSNR: 51.501 MSE: 0.46024 NC: 0.117945
<i>pepper.bmp</i> (512x512 piksel)	M: 0.1			PSNR: 51.5662 MSE: 0.45338 NC: 0.0462066

Hasil pengujian menunjukkan nilai-nilai yang didapat memiliki pola yang sama. Saat konstanta input noise kecil (1 x 10⁻⁸) watermark berhasil diekstraksi, nilai NC yang didapat tinggi. Tetapi ketika konstanta noise yang diinputkan lebih besar (1 x 10⁻⁵), watermark gagal diekstraksi dengan nilai NC rendah, kemiripan antara watermark hasil ekstraksi dengan watermark asli rendah..

Nilai MSE yang didapat bervariasi, menurun seiring dengan konstanta noise yang diinputkan semakin kecil. Tetapi secara keseluruhan, MSE yang didapat bernilai rendah, sehingga image yang dikenai serangan memiliki tingkat error yang rendah.

Nilai PSNR yang didapat menunjukkan penambahan yang signifikan walau sangat kecil seiring dengan semakin kecil konstanta input noise yang dimasukkan. Semakin kecil nilai konstanta Gaussian Noise yang diinputkan, semakin besar nilai PSNR yang didapat.

7.3.2 Salt & Pepper Noise

Salt & Pepper merupakan noise yang terdiri titik-titik hitam dan titik-titik putih. Pada Matlab, salt & pepper diinputkan dengan nilai konstanta antara 0 – 1, semakin besar nilai konstanta yang diinputkan maka image akan semakin kabur. Pengujian dengan Salt & Pepper ini dilakukan menggunakan file cover medium lena.jpg dan 3 file watermark, yaitu uns400.png, dogs.jpg, dan pepper.bmp. Hasil pengujian ditampilkan pada tabel 4.

Tabel 4 Hasil Pengujian dengan Serangan Salt & Pepper Noise

Watermark	Salt & Pepper Noise	Image terwater mark	Watermark Hasil Ekstraksi	PSNR (dB), MSE, dan NC
uns400.png (400 x400 piksel)	(1 x 10 ⁻⁸)			PSNR: 48.9434 MSE: 0.829346 NC: 0.889784
uns400.png (400 x400 piksel)	(1 x 10 ⁻⁵)			PSNR: 48.9435 MSE: 0.829333 NC: 0.12513
uns400.png (400 x400 piksel)	(1 x 10 ⁻²)			PSNR: 49.0012 MSE: 0.818399 NC: 0.0098972
dogs.jpg (640x480 piksel)	(1 x 10 ⁻⁸)			PSNR: 54.152 MSE: 0.249967 NC: 0.959068
dogs.jpg (640x480 piksel)	(1 x 10 ⁻⁵)			PSNR: 54.1519 MSE: 0.249969 NC: 0.186604
dogs.jpg (640x480 piksel)	0.01			PSNR: 54.1929 MSE: 0.247624 NC: 0.0302736
pepper.bmp (512x512 piksel)	(1 x 10 ⁻⁸)			PSNR: 51.4941 MSE: 0.460964 NC: 0.97296
pepper.bmp (512x512 piksel)	(1 x 10 ⁻⁵)			PSNR: 51.4942 MSE: 0.460957 NC: 0.242827
pepper.bmp (512x512 piksel)	0.01			PSNR: 51.5622 MSE: 0.453799 NC: 0.0521021

Hasil pengujian yang ditunjukkan pada Tabel 4 memiliki pola yang sama dengan hasil pengujian pada serangan Gaussian Noise. Watermark dapat diekstraksi dengan baik saat input konstanta kecil (1 x 10⁻⁸) dengan nilai NC yang tinggi, tingkat kemiripan watermark hasil ekstraksi dengan watermark asli tinggi. Tetapi saat input konstanta noise lebih besar, watermark gagal diekstraksi, nilai NC yang didapat rendah menunjukkan tingkat kemiripan watermark asli dengan watermark hasil ekstraksi rendah.

Nilai MSE menunjukkan bahwa semakin rendah konstanta noise yang diinputkan, nilai MSE yang diperoleh semakin tinggi. Hal ini berarti seiring dengan tingkat kemiripan yang naik tingkat error yang didapat semakin rendah. Sedangkan nilai PSNR yang didapat berkisar 48 – 50 dB menunjukan kualitas image baik.

7.3.3 Poisson Noise

Poisson noise merupakan noise yang dapat ditambahkan langsung tanpa memasukkan parameter apapun. Pengujian dilakukan menggunakan file cover medium lena.jpg dan 3 file watermark, yaitu uns400.png, dogs.jpg, dan pepper.bmp. Hasil pengujian ditampilkan pada tabel 5.

Tabel 5 Hasil Pengujian dengan Serangan Poisson Noise

Watermark	Image	Watermark Hasil Ekstraksi	PSNR (dB),
-----------	-------	---------------------------	------------

	terwater mark	k Hasil Ekstraksi	MSE dan NC
uns400.png (400 x400 piksel)			PSNR: 48.9434 MSE: 0.829346 NC: 0.889782
dogs.jpg (640x480 piksel)			PSNR: 54.152 MSE: 0.249967 NC: 0.959066
pepper.bmp (512x512 piksel)			PSNR: 51.4941 MSE: 0.460964 NC: 0.972959

Hasil pengujian menunjukkan image yang diberi serangan poisson noise dapat diekstraksi dengan baik dengan nilai NC yang didapat tinggi sehingga dapat dikatakan tingkat kemiripan antara watermark hasil ekstraksi dengan watermark asli tinggi. Nilai PSNR yang didapat sekitar 48 – 54 dB menunjukkan kualitas watermark hasil ekstraksi baik. Sedangkan error pada image yang ditunjukkan dengan nilai MSE rendah, yaitu sekitar 0.4 – 0.8.

7.3.5 Geometrical Attacks

Geometrical attacks merupakan serangan yang sangat merubah image, karena dapat membuat seluruh piksel berpindah tempat atau hilang. Serangan geometrical attacks dapat berupa cropping dan rotasi. Pengujian dilakukan menggunakan file cover medium lena.jpg dan 3 file watermark, yaitu uns400.png, dogs.jpg, dan pepper.bmp. Hasil pengujian serangan rotation ditampilkan pada tabel 6 sedangkan pengujian pada serangan cropping ditunjukkan pada Tabel 7.

Tabel 6 Hasil Pengujian dengan Serangan Rotation

Watermark	Rotasi	Image terwater mark	Watermark Hasil Ekstraksi	PSNR (dB), MSE dan NC
uns400.png (400 x400 piksel)	1°			PSNR: 44.9739 MSE: 2.06866 NC: -0.00996111
uns400.png (400 x400 piksel)	30°			PSNR: 37.5684 MSE: 11.3825 NC: -0.032968
uns400.png (400 x400 piksel)	90°			PSNR: 39.8552 MSE: 6.72292 NC: -0.00857926
dogs.jpg (640x480 piksel)	1°			PSNR: 47.0091 MSE: 1.29471 NC: 0.0668491
dogs.jpg (640x480 piksel)	30°			PSNR: 38.7895 MSE: 8.59262 NC: 0.275835
dogs.jpg (640x480 piksel)	90°			PSNR: 40.2667 MSE: 6.11517 NC: 0.00574302
pepper.bmp (512x512 piksel)	1°			PSNR: 46.1264 MSE: 1.5865 NC: 0.0246164
pepper.bmp (512x512 piksel)	30°			PSNR: 38.226 MSE: 9.78311 NC: 0.0908375
pepper.bmp (512x512 piksel)	90°			PSNR: 40.4286 MSE: 5.8914 NC: 0.185294

Hasil pengujian image yang dikenai serangan rotasi dengan persentasi rotasi yang bervariasi seluruhnya gagal diekstraksi. Seluruh image yang diujikan memiliki nilai NC rendah sekitar 0.01 – 0.2 yang menunjukkan tingkat kemiripan antara watermark hasil ekstraksi dengan watermark asli sangat

rendah.

Nilai MSE yang didapat pada seluruh *image* tinggi, yaitu antara angka 1 – 9.7 sehingga tingkat *error* yang didapat tinggi. Sedangkan untuk nilai PSNR pada keseluruhan *image* sekitar 36 – 42 dB yang menunjukkan kualitas *image* baik.

Tabel 7 Hasil Pengujian dengan Serangan Cropping

Watermark	Image terwater mark	Watermark Hasil Ekstraksi	PSNR (dB), MSE dan NC
uns400.png (400 x400 piksel)			PSNR: 40.472 MSE: 5.83287 NC: 0.0154622
dogs.jpg (640x480 piksel)			PSNR: 40.5899 MSE: 5.67657 NC: 0.0788692
pepper.bmp (512x512 piksel)			PSNR: 40.8074 MSE: 5.39934 NC: 0.189743

Tabel 7 menunjukkan seluruh *image* yang dikenai serangan *cropping* gagal diekstraksi. Nilai NC yang diperoleh sangat kecil antara 0.01 – 0.1 yang menunjukkan tingkat kemiripan antara *watermark* asli dengan *watermark* hasil ekstraksi sangat rendah. Sedangkan nilai MSE yang didapat pada ketiga *image* tinggi, sekitar 5 – 6, sehingga dapat dikatakan *error* pada *image* tinggi. Kualitas *watermark* hasil ekstraksi baik karena Nilai PSNR yang didapat berkisar 40 dB.

7.3.6 Blurring

Blurring adalah operasi pengaburan *image*. Pengaburan *image* adalah suatu filter *low-pass* yang melenyapkan detail halus dari suatu *image*. Pengujian dilakukan menggunakan file cover *medium lena.jpg* dan 3 file *watermark*, yaitu *uns400.png*, *dogs.jpg*, dan *pepper.bmp*. Hasil pengujian ditampilkan pada tabel 8.

Tabel 8 Hasil Pengujian dengan Serangan Blurring

Watermark	Image terwater mark	Watermark Hasil Ekstraksi	PSNR (dB), MSE dan NC
uns400.png (400 x400 piksel)			PSNR: 47.4451 MSE: 1.17103 NC: -0.0040617
dogs.jpg (640x480 piksel)			PSNR: 50.4313 MSE: 0.588778 NC: 0.00212683
pepper.bmp (512x512 piksel)			PSNR: 49.1225 MSE: 0.795855 NC: 0.00891258

Tabel 8 menunjukkan hasil pengujian pada ketiga *image* gagal diekstraksi. Nilai NC yang didapat pada ketiga *image*

rendah, yaitu berkisar (-0.004) – 0.008 yang berarti tingkat kemiripan *watermark* hasil ekstraksi dengan *watermark* asli sangat rendah. Sedangkan nilai PSNR menunjukkan ketiga *image* berkualitas baik walaupun dengan tingkat *error* yang tinggi.

8. Kesimpulan

Berdasarkan penelitian, hasil dan pembahasan yang telah dipaparkan di atas, maka dapat ditarik kesimpulan bahwa *Watermarking Image Digital* dengan *Discrete Wavelet Transform* dan *Singular Value Decomposition (DWT-SVD)* dapat diekstraksi kembali dengan baik. Selain itu, *watermark* tahan terhadap beberapa serangan seperti *Gaussian Noise*, *Speckle Noise*, *Salt & Pepper Noise*, dan *Poisson Noise* dengan rata-rata nilai NC 0.8, *watermark* tidak tahan terhadap serangan *rotation*, *cropping*, dan *blurring*. Untuk penelitian selanjutnya variasi serangan pada pengujian dapat ditambah, atau dapat menggunakan filter *Wavelet* lain seperti *Daubechies* dan atau menggunakan model warna *image* selain RGB, seperti HSI (*Hue, Saturation, Intensity*) dan CMY (*Cyan, Magenta, Yellow*).

9. Daftar Pustaka

- [1]. Cox, Ingemar J., dkk. 2008. *Digital Watermarking and Steganography – Second Edition*. Burlintong, USA: Morgan Kaufmann Publishers.
- [2]. Katzenbeisser, S., F. A. P. Petitcolas. 2000. *Information Hiding: Techniques for Steganography and Digital Watermarking*. Computer Security Series: Artech House Books.
- [3]. Gonzales, R.C., Woods, R.E. 2010. *Digital Image Processing – Third Edition*. Prentice Hall. Upper Saddle River, New Jersey, 07458.
- [4]. Alfatwa, Dean Fathony. 2011. *Watermarking pada Image Digital menggunakan Discrete Wavelet Transform*. Program Studi Teknik Informatika, Institut Teknologi Bandung.
- [5]. T. J., Anumol., Karthigaikumar, P. 2011. *DWT based Invisible Image Watermarking Algorithm for Color Images*. IJCA Special Issue on “Computational Science – New Dimensions & Perspectives” NCCSE 2011 pp. 76-79.
- [6]. Praful Saxena, dkk. 2012. *DWT-SVD Semi-Blind Image Watermarking Using High Frequency Band*. 2nd International Conference on Computer Science and Information Technology (ICCSIT’2012) Singapore April 28-29, 2012 pp.138-142.
- [7]. Putra, Dharma. 2010. *Pengolahan Image Digital*. Yogyakarta: Penerbit Andi.