



***WATERMARKING CITRA MENGGUNAKAN DISCRETE  
WAVELET TRANSFORM DAN ERROR DETECTION  
CORRECTION PADA MATLAB***

**SKRIPSI**

Oleh

**Dicky Ardiyanto  
NIM 131910201026**

**PROGRAM STUDI STRATA-1 TEKNIK ELEKTRO  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2018**



***WATERMARKING CITRA MENGGUNAKAN DISCRETE  
WAVELET TRANSFORM DAN ERROR DETECTION  
CORRECTION PADA MATLAB***

**SKRIPSI**

diajukan guna melengkapi skripsi dan memenuhi salah satu syarat  
untuk menyelesaikan Program Studi Teknik Elektro (S1)  
dan mencapai gelar Sarjana Teknik

Oleh

**Dicky Ardiyanto  
NIM 131910201026**

**PROGRAM STUDI STRATA-1 TEKNIK ELEKTRO  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2018**

**PERSETUJUAN PEMBIMBING**

Skripsi berjudul ” *Watermarking* Citra Menggunakan *Discreate Wavelet Transform* dan *Error Detection Correction* ” pada Matlab” telah disetujui pada:

hari, tanggal : Rabu, 17 Juli 2018

tempat : Fakultas Teknik Universitas Jember.

Dosen Pembimbing Utama,

Dosen Pembimbing Anggota

Catur Suko Sarwono, S.T., M.Si.

Samsul Bachri M, S.T., M.MT.

NIP. 196801191997021001

NIP. 196403171998021001

## PERSEMBAHAN

Dengan rasa syukur saya panjatkan kepada Allah SWT atas segala karunia-Nya hingga saya dapat menyelesaikan skripsi ini. Skripsi ini merupakan langkah awal kesuksesan yang saya raih untuk mendekati masa depan dan meraih cita-cita di dalam hidup saya. Dengan penuh rasa syukur dengan ketulusan hati saya persembahkan karya ini kepada :

1. Allah SWT Yang Maha atas segalanya;
2. Nabi besar Muhammad SAW yang menjadi suri tauladan bagi seluruh umat manusia;
3. Kedua orang tua, Pudji Astuti dan Bapak Sukardi yang tercinta dan tersayang dan Kakak saya yang sangat saya banggakan;
4. Dosen Pembimbing Catur Suko Sarwono, S.T., M.Si. dan Bapak Samsul Bachri M, S.T., M.TT. atas kesabaran dan keikhlasan dalam membimbing saya menyelesaikan skripsi ini;
5. Keluarga besar Teknik Elektro Fakultas Teknik Universitas Jember;
6. Keluarga INTEL UJ 13 “Ikatan Teknik Elektro Universitas Jember 2013”;
7. Almamater Fakultas Teknik Universitas Jember.

## MOTTO

Sembahlah Allah dan janganlah kamu mempersekutukan-Nya dengan sesuatupun. Dan berbuat baiklah kepada dua orang ibu-bapa, karib-kerabat, anak-anak yatim, orang-orang miskin, tetangga yang dekat dan tetangga yang jauh, dan teman sejawat, ibnu sabil dan hamba sahayamu. Sesungguhnya Allah tidak menyukai orang-orang yang sombong dan membangga-banggakan diri. \*)

وَالصَّلَاةِ الصَّابِرِ وَاسْتَعِينُوا

Mintalah pertolongan dengan sabar dan shalat. \*\*)

“Akal dan belajar adalah itu seperti raga dan jiwa. Tanpa raga, jiwa hanyalah udara hampa. Tanpa jiwa, raga adalah kerangka tanpa makna.” \*\*\*)

---

\*) (Q.S An Nisaa', 4:36)..

\*\*) QS Al-Baqarah: 45.

\*\*\*) (Kahlil Gibran)

**PERNYATAN**

Saya yang bertanda tangan di bawah ini:

Nama : Dicky Ardiyanto

NIM : 131910201026

Menyatakan dengan sesungguhnya bahwa proyek akhir yang berjudul “*Watermarking Citra Menggunakan Discrete Wavelet Transform dan Error Detection Correction* pada Matlab” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan dalam institusi mana pun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak mana pun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, 17 Juli 2018

Yang menyatakan,

Dicky Ardiyanto  
NIM 131910201026

**SKRIPSI**

***WATERMARKING CITRA MENGGUNAKAN DISCRETE WAVELET  
TRANSFORM DAN ERROR DETECTION CORRECTION PADA MATLAB***

Dicky Ardiyanto  
NIM 131910201026

Pembimbing :

Dosen Pembimbing Utama : Catur Suko Sarwono, ST., MSi.  
Dosen Pembimbing Anggota : Samsul Bachri M., S.T., MT.

**PENGESAHAN**

Skripsi berjudul “*Watermarking Citra Menggunakan Discreate Wavelet Transform dan Error Detection Correction pada Matlab*” karya Dicky Ardiyanto telah diuji dan disahkan pada :

hari, tanggal : 17 Juli 2018

tempat : Fakultas Teknik Universitas Jember.

**Tim Penguji:**

Ketua,

Anggota I,

Catur Suko Sarwono, ST., MSi.  
NIP. 19680119 199702 1 001

Samsul Bachri M., ST., MMT.  
NIP. 19840531 200812 1 004

Anggota II,

Anggota III,

Dodi Setiabudi, S.T., M.T.  
NIP. 19840531 200812 1 004

Andrita Ceriana Eska, S.T., M.T  
NIP. 760014640

Mengesahkan  
Dekan,

Dr. Ir. Entin Hidayah M.U.M  
NIP. 19661215 199503 2 001

## RINGKASAN

**“Watermarking Citra Menggunakan Discrete Wavelet Transform Dan Error Detection Correction Pada Matlab”**; Dicky Ardiyanto 131910201026; 2018: 71 halaman; Program Studi Strata 1 (S1) Teknik, Jurusan Teknik Elektro, Fakultas Teknik Universitas Jember.

*Watermarking* merupakan teknologi untuk memberikan dan menentukan hak otentikasi terhadap suatu karya (Felyana, 2013) seperti foto dan video. Berbagai masalah dapat muncul dengan produk tersebut contohnya penggantian, plagiat, pelanggaran hak cipta (*copyright*). *International Association of Chief's of Police* (IACP) mengumumkan data bahwasannya di facebook setiap hari ada 350 juta foto yang diunggah, dan pada instagram lebih dari 67 juta foto yang diunggah. Dan menurut *International Intellectual Property Alliance* (IIPA), sebanyak 23.37% dari lalu lintas di dunia maya adalah pelanggaran *copyright* pada tahun 2013 (taufanwijaya.wordpress.com). Untuk menangani permasalahan tersebut maka dapat diselesaikan dengan beberapa cara salah satunya dengan menggunakan *Watermarking*

Teknik *watermarking* yang digunakan yaitu *Discrete Wavelet Transform* (DWT) menggunakan *wavelet Dabauchies*. Pada teknik *watermarking*, setiap informasi yang disisipkan harus tahan terhadap apa yang mengakibatkan penurunan kualitas media digital tersebut. Sehingga mata manusia tidak dapat membedakan citra asli atau citra terwatermark. Untuk meningkatkan ketahanan citra *watermark* maka *error detection and correcting* yang digunakan. *Error detection and correcting* berfungsi untuk mengurangi *noise* bahkan mengatasi *noise* tersebut saat proses watermarking maupun saat pengekstrakan. Pada penelitian ini menggunakan *BCH code*. Pada penerapan metode ini diterapkan menggunakan simulasi MATLAB R2012b.

Dasil Pengujian menggunakan simulasi MATLAB dapat disimpulkan nilai MAD dari citra rekontruksi mendekati citra asli. Untuk nilai PSNR pada citra rekontruksi tergolong bagus yaitu pada kisaran 50.6-52.9 dB. Pada analisi

ketahanan citra, citra watermark dapat diekstrak dapat diartikan program bisa digunakan.



## PRAKATA

*Bismillahirrohmanirohim.*

Puji syukur kehadiran Allah SWT atas limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “*Watermarking Citra Menggunakan Discreate Wavelet Transform Dan Error Detection Correction*”. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan program studi strata satu (S1) Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Selama penyusunan skripsi ini penulis mendapat bantuan dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada:

1. Dosen Pembimbing Catur Suko Sarwono, S.T., M.Si. dan Bapak Samsul Bachri M, S.T., M.TT. atas kesabaran dan keikhlasan dalam membimbing saya menyelesaikan skripsi ini;
2. Dodi Setiabudi, ST., MT. selaku Dosen Penguji I, Andrita Ceriana Eska, S.T., M.T. selaku Dosen Penguji II yang telah memberikan kritik dan saran yang sangat membangun demi penyempurnaan skripsi ini;
3. Dr. Ir. Bambang Sri Kaloko, S.T., M.T., selaku Dosen Pembimbing Akademik yang telah membimbing selama penulis menjadi mahasiswa;
4. Ibu Pudji Astuti dan Bapak Sukardi selaku orangtua tercinta saya yang telah memberikan dukungan moril dan materiil serta kasih sayang yang tak terhingga sepanjang masa;
5. Seluruh Dosen Teknik Elektro Universitas Jember yang telah memberikan ilmu pengetahuan dan bimbingan selama mengikuti pendidikan di Universitas Jember;
6. Keluarga besar Teknik Elektro khususnya angkatan 2013 (INTEL UNEJ), terimakasih atas dukungan dan motivasi yang kalian berikan;
7. Erli Indirasari yang telah memberikan semangat tiada hentinya, kritik dan masukan yang sangat membantu dalam proses pengerjaan skripsi saya;
8. Sahabat dari sejak SD, SMP hingga SMK sampai sekarang yang selalu memberikan dukungan untuk menyelesaikan skripsi;

9. Teman-teman Teknik Mesin 2013 (M. Ody Alfaris, M. Angga Saputra, Hirzan) yang telah membantu untuk menyelesaikan skripsi ini;
10. Serta semua pihak yang tidak dapat disebutkan satu per satu, terimakasih banyak yang mana telah mendukung dan memberikan semangat dalam penyelesaian skripsi ini;

Penulis menyadari bahwa kesempurnaan hanya milik-Nya sehingga sebagai manusia biasa, penulis selalu terbuka terhadap masukan dan menerima segala kritik dan saran dari semua pihak yang sifatnya membangun demi kesempurnaan skripsi ini. Penulis berharap, semoga skripsi ini dapat bermanfaat bagi pembaca dan tidak lupa juga penulis menyampaikan permohonan maaf yang sebesar-besarnya jika terdapat kesalahan dan kekeliruan di dalam skripsi ini.

Jember, 17 Juli 2018

Penulis

DAFTAR ISI

	Halaman
HALAMAN SAMPUL.....	i
HALAMAN JUDUL .....	ii
PERSETUJUAN PEMBIMBING .....	iii
PERSEMBAHAN.....	iv
MOTTO .....	v
PERNYATAN.....	vi
PENGESAHAN .....	viii
RINGKASAN .....	ix
PRAKATA .....	xi
DAFTAR ISI.....	xiii
DAFTAR TABEL .....	xvi
DAFTAR GAMBAR.....	xvii
<b>BAB 1. PENDAHULUAN .....</b>	<b>1</b>
<b>1.1 Latar Belakang.....</b>	<b>1</b>
<b>1.2 Rumusan Masalah .....</b>	<b>2</b>
<b>1.3 Batasan Masalah .....</b>	<b>2</b>
<b>1.4 Tujuan Penelitian.....</b>	<b>2</b>
<b>1.5 Manfaat Penelitian.....</b>	<b>3</b>
<b>BAB 2. TINJAUAN PUSTAKA.....</b>	<b>4</b>
<b>2.1 Watermarking .....</b>	<b>4</b>
2.1.1 Kinerja Watermarking .....	5
2.1.2 Jenis-jenis Watermarking .....	6
2.1.3 Type Watermarking .....	6
2.1.4 Klasifikasi Watermarking.....	7
<b>2.2 Transformasi Wavelet (Wavelet Transform) .....</b>	<b>8</b>
2.2.1 Discreate Wavelet Transform.....	9
<b>2.3 Error Detection And Correction .....</b>	<b>11</b>
2.3.1 Error Correcting Code .....	12

2.4	<b>MATLAB</b> .....	12
2.4.1	Aplikasi <i>M-file</i> .....	13
2.5	<b>Citra Digital</b> .....	13
2.5.1	Jenis Citra Digital .....	14
2.5.2	Transformasi Citra.....	16
<b>BAB 3.</b>	<b>METODE PENELITIAN</b> .....	<b>17</b>
3.1	<b>Tahap Penelitian</b> .....	<b>17</b>
3.2	<b>Alat dan Bahan</b> .....	<b>18</b>
3.3	<b>Blok Diagram</b> .....	<b>19</b>
3.3.1	Proses Penyisipan Citra <i>Watermark</i> dan DWT .....	19
3.3.2	Proses Ekstrasi Citra.....	22
3.4	<b>Flowchart Tahapan Penyisipan Citra dan Ekstrasi</b> .....	<b>23</b>
3.5	<b>Flowchart Tahapan Pengujian Ketahanan Citra</b> .....	<b>26</b>
3.5.1	Tahapan pengambilan data PSNR .....	27
3.5.2	Tahapan pengambilan data ketahanan citra.....	28
3.6	<b>Metode Pengujian</b> .....	<b>29</b>
3.6.1	Analisa Citra Asli dan <i>Watermark</i> .....	30
3.6.2	PSNR .....	31
3.6.3	Analisa Ketahanan Citra.....	31
<b>BAB 4.</b>	<b>HASIL DAN PEMBAHASAN</b> .....	<b>32</b>
4.1	<b>Analisa Perbedaan Citra Asli dan Citra Watermark Secara Visual</b> .....	<b>33</b>
4.2	<b>Peak Signal-to-Noise Ratio (PSNR)</b> .....	<b>38</b>
4.2.1	Menentukan PSNR .....	38
4.2.2	Hasil Pengujian PSNR.....	38
4.3	<b>Pengujian Citra Menggunakan Parameter Wavelet</b> .....	<b>39</b>
4.3.1	Analisis Citra Menggunakan <i>Wavelet Dabauchies</i> .....	40
4.4	<b>Pengujian Ketahanan Citra</b> .....	<b>47</b>
4.4.1	Analisa Ketahanan Citra Menggunakan <i>Noise Gaussian</i> .....	47
4.4.2	Analisa ketahanan Menggunakan <i>Image Brightness</i> .....	50
<b>BAB 5.</b>	<b>KESIMPULAN DAN SARAN</b> .....	<b>52</b>

5.1 Kesimpulan.....	52
5.2 Saran .....	52
DAFTAR PUSTAKA .....	53
LAMPIRAN.....	54



**DAFTAR TABEL**

Tabel 2.1 PSNR.....	6
Tabel 3.1 Citra asli .....	19
Tabel 3.2 Contoh citra yang akan diuji .....	29
Tabel 3.3 Survei citra terhadap responden .....	30
Tabel 3.4 PSNR.....	31
Tabel 3.5 Pengujian setelah diberi <i>noise</i> .....	31
Tabel 4.1 Daftar jenis citra.....	32
Tabel 4.2 Pemilihan citra koresponden .....	34
Tabel 4.3 Pemilihan citra koresponden 2 .....	35
Tabel 4.4 Pemilihan citra koresponden 3 .....	36
Tabel 4.5 Pemilihan citra koresponden 4 .....	37
Tabel 4.6 Nilai PSNR citra ter <i>watermark</i> .....	38
Tabel 4.7 Hasil citra menggunakan <i>wavelet debauchies</i> .....	42
Tabel 4.8 Jenis citra noise .....	48
Tabel 4.9 Nilai PSNR rujukan untuk pengujian citra <i>noise</i> .....	49
Tabel 4.10 Hasil pengujian citra menggunakan <i>noise gaussian</i> .....	49

**DAFTAR GAMBAR**

Gambar 2.1 Dekomposisi <i>wavelet</i> tiga tingkat .....	10
Gambar 2.2 Rekontruksi <i>wavelet</i> 3 tingkat .....	11
Gambar 2.3 Contoh citra digital .....	14
Gambar 2.4 Citra biner .....	14
Gambar 2.5 Citra grayscale 4 <i>bi</i> .....	15
Gambar 2.6 Citra warna .....	15
Gambar 3.1 Tahap penelitian .....	17
Gambar 3.2 Diagram alur proses <i>watermarking</i> .....	19
Gambar 3.3 Citra <i>watermark</i> yang digunakan .....	20
Gambar 3.4 Dekomposisi citra .....	21
Gambar 3.5 Proses ekstraksi citra .....	22
Gambar 3.6 <i>Flowchart</i> proses <i>watermarking</i> .....	23
Gambar 3.7 Proses penyisipan citra .....	24
Gambar 3.8 Proses menampilkan kembali hasil penyisipan .....	24
Gambar 3.9 Proses ekstraksi .....	25
Gambar 3.10 <i>Flowchart</i> pengujian .....	26
Gambar 3.11 Pengambilan citra asli dan rekontruksi .....	27
Gambar 3.12 Perhitungan nilai MSE dan PSNR .....	27
Gambar 3.13 Proses pemberian citra <i>noise gaussian</i> .....	28
Gambar 3.14 Proses mengubah kekontrasan citra .....	28
Gambar 4.1 Citra 1 asli dan citra 2 ter <i>watermark</i> .....	33
Gambar 4.2 Citra 1 citra asli dan citra <i>watermark</i> (ECC) .....	34
Gambar 4.3 Citra 2 citra asli dan citra <i>watermark</i> (ECC) .....	35
Gambar 4.4 Citra 1 citra asli dan citra <i>watermark</i> (ECC) .....	36
Gambar 4.5 Awal tampilan sebelum proses rekrontruksi .....	40
Gambar 4.6 <i>Statistic</i> citra pertama level 4 .....	40
Gambar 4.7 Citra <i>Horizontal</i> hasil rekrontruksi level 4 .....	41
Gambar 4.8 Citra <i>vertical</i> hasil rekrontruksi level 4 .....	41
Gambar 4.9 Citra <i>diagonal</i> hasil rekrontruksi level 4 .....	42

Gambar 4.10 Citra terwatermark dan citra *noise gaussian*..... 48  
Gambar 4.11 Citra terwatermark (ECC) dan citra *noise gaussian* ..... 48  
Gambar 4. 12 (A) Citra watermark (B) citra ekstrasi watermark *noise gaussian* 49



## BAB 1. PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi informasi di dunia sudah tidak terbendung lagi. Perkembangan teknologi membuat informasi dalam bentuk media dapat tersebar dengan cepat. Apalagi dengan keberadaan internet informasi dapat menyebar dengan cepat dan mudah tanpa mengenal ruang dan waktu (Solichin, 2010). Berkembangnya teknologi informasi banyak orang ingin menginvestasikan di bidang ini contohnya berupa foto. Foto atau citra merupakan karya intelektual manusia. Produk ini adalah hasil dari beberapa karya orang mulai dari fotografi dan desainer yang memamerkan karyanya menggunakan produk digital. Berbagai masalah dapat muncul dengan produk tersebut contohnya pengandaan, plagiat, mengubah informasi (manipulasi) yang dilakukan dengan cara tidak sah (*illegal*), pelanggaran hak cipta (*copyright*). Permasalahan tersebut dapat diatasi salah satunya menggunakan teknik watermarking.

*Watermarking* merupakan teknologi untuk memberikan dan menentukan hak otentikasi terhadap suatu karya (Felyana, 2013). Teknik *watermarking* ada dua yaitu *Discrete Cosine Transform* (DCT) dan *Discrete Wavelet Transform* (DWT) (Betri, 2014). Pada *watermarking* setiap informasi yang disisipkan harus tahan terhadap apa yang mengakibatkan penurunan kualitas media digital tersebut sehingga mata manusia tidak dapat membedakan citra asli atau citra terwatermark. Proses penyisipan citra *watermark* terdapat permasalahan ketahanan pada saat pengekstrakan solusinya adalah menggunakan *error detection and correcting* untuk mempertahankan citra watermark pada proses pengekstrakan. *Error detection and correcting* berfungsi untuk mengurangi *noise* bahkan mengatasi *noise* tersebut saat proses watermarking maupun saat pengekstrakan.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, berikut ini adalah beberapa masalah yang akan diselesaikan dalam penelitian ini yaitu:

1. Bagaimana cara membuat aplikasi *watermarking* dengan menggunakan *Discrete Wavelet Transform* (DWT) menggunakan MATLAB pada citra?
2. Bagaimana pengaruh *error detection and correction* terhadap citra *watermark* ?
3. Bagaimana cara menanamkan citra *watermark* ke dalam citra asli dengan menggunakan *Discrete Wavelet Transform* (DWT) ?

## 1.3 Batasan Masalah

Berdasarkan rumusan masalah penelitian diatas, agar pembahasan tidak terlalu luas maka diperlukan suatu pembatasan masalah sebagai berikut.

1. Media yang disisipkan *watermark* berupa citra berformat PNG.
2. Implementasi proses watermaking citra menggunakan perangkat lunak MATLAB R2012b
3. Citra yang digunakan untuk proses *watermarking* adalah citra berwarna (RGB)

## 1.4 Tujuan Penelitian

Penelitian yang diusulkan dalam proposal ini memiliki beberapa tujuan sebagai berikut:

1. Membuat simulai progam untuk melindungi hak cipta pada citra digital
2. Merancang progam aplikasi untuk *watermarking* digital pada citra dengan menggunakan *Discrete Wavelet Transform* (DWT) menggunakan jenis *Wavelet daubachies*
3. Mempertahankan kualitas citra ter*watermark* dari serangan yang menyebabkan penurunan kualitas citra

### 1.5 Manfaat Penelitian

Penelitian ini diharapkan bermanfaat sebagai:

1. Untuk melindungi hak cipta citra atau foto dari penggandaan, plagiat, mengubah informasi (manipulasi) yang dilakukan dengan cara tidak sah.
2. Penelitian ini dapat digunakan sebagai salah satu referensi untuk membuat aplikasi *watermarking* dengan *Discrete Wavelet Transform* (DWT) pada citra



## BAB 2. TINJAUAN PUSTAKA

### 2.1 *Watermarking*

Menurut Supangkat (2000), *watermark* merupakan sebuah informasi yang disisipkan pada media lain dengan tujuan melindungi media yang disisipi oleh informasi tersebut dari pembajakan, penyalahgunaan hak cipta, dsb. *Watermarking* sendiri adalah suatu cara untuk menyembunyikan atau menanamkan data tertentu ke dalam suatu data digital lainnya, tetapi tidak diketahui kehadirannya oleh indera manusia dan mampu menghadapi proses-proses pengolahan sinyal digital sampai pada tahap tertentu (Hakim, 2012).

*Digital watermarking* sendiri adalah sebuah kode identifikasi yang secara permanen disisipkan ke dalam data digital dengan membawa informasi yang berhubungan dengan perlindungan hak cipta dan otentikasi data. Ada beberapa tujuan yang ingin dicapai dari penggunaan *watermarking* sebagai suatu teknik penyembunyian data pada suatu data digital lain yaitu:

#### 1. *Tamper-proofing*

*Watermarking* digunakan sebagai indikator yang menunjukkan apakah data digital yang asli telah mengalami perubahan dari yang aslinya (mengecek integritas data).

#### 2. *Feature llocation*

*Watermarking* sebagai alat identifikasi isi dari data digital pada lokasi-lokasi tertentu, misalnya penamaan suatu objek tertentu dari beberapa objek yang ada pada suatu citra digital.

#### 3. *Annotation / caption*

*Watermark* berisi keterangan tentang data digital itu sendiri. Selain itu *watermark* juga dapat digunakan untuk mengirimkan pesan rahasia.

#### 4. *Copyright-labelling*

*Watermarking* digunakan sebagai metode untuk menyembunyikan label hak cipta pada data digital atau sebuah bukti otentik kepemilikan atas dokumen digital tersebut.

Teknik *watermarking* pada citra digital dapat diklasifikasikan dalam dua kategori, yaitu teknik ranah spasial dan teknik ranah frekuensi. Pada ranah spasial, penyisipan *watermark* dilakukan secara langsung ke dalam piksel citra sedangkan pada ranah frekuensi, penyisipan *watermark* dilakukan ke dalam koefisien transformasi. Biasanya teknik *watermarking* yang kuat (susah dipecahkan oleh berbagai serangan) memiliki kualitas gambar ber*watermark* yang kurang memuaskan, demikian juga sebaliknya, teknik *watermarking* yang menghasilkan kualitas gambar yang memuaskan biasanya kurang kuat menghadapi serangan.

### 2.1.1 Kinerja *Watermarking*

Kinerja *watermarking* diukur dari kualitas citra yang dihasilkan, nilai *Normalized Crosscorrelation* (NC) *watermark* hasil ekstraksi dengan *watermark* asli, dan lama waktu penyisipan *watermark* (Tyas, 2011).

#### a. *Peak Signal to Noise Ratio* (PSNR)

*Peak Signal to Noise Ratio* (PSNR) digunakan untuk menentukan kualitas citra. Nilai *PSNR* diperoleh dengan membandingkan citra asli dan citra rekonstruksi. Semakin besar nilai SNR semakin baik kualitas citra yang dihasilkan.

Untuk menghitung SNR terlebih dahulu harus menghitung nilai *Mean Squared Error* dari suatu citra hasil rekonstruksi N

$$MSE = \frac{\sum [F(i,j) - f(i,j)]^2}{N} \dots\dots\dots 2.1$$

N menyatakan hasil perkalian panjang dan lebar citra dalam piksel.  $f(i,j)$  merupakan citra hasil rekonstruksi, sedangkan  $F(i,j)$  adalah citra asal.

Untuk menentukan nilai PSNR digunakan rumus :

$$PNSR = 10 \log_{10} \left( \frac{255^2}{MSE} \right) \dots\dots\dots 2.2$$

dengan N adalah baris dan kolom citra, F dan f adalah citra asli dan citra rekonstruksi. Ukuran kualitas citra disajikan pada Tabel 2.1.

Tabel 2.1 PSNR

PSNR (dB)	Kualitas citra
$50 \leq \text{PSNR} < 60$	Bagus
$40 \leq \text{PSNR} < 50$	Layak
$30 \leq \text{PSNR} < 40$	Cukup
$20 \leq \text{PSNR} < 30$	Tidak dapat dipakai

(Sumber: Tyas, 2011)

### 2.1.2 Jenis-jenis *Watermarking*

Pada digital *watermarking* terdapat empat jenis berdasarkan media digital yang disisipi (Hakim, 2012), yaitu :

1. *Text watermarking*

*Watermark* disisipkan pada media digital seperti dokumen atau teks

2. *Image watermarking*

*Watermark* disisipkan pada citra digital

3. *Audio watermarking*

*Watermark* disisipkan pada gambar bergerak atau disebut dengan video digital

4. *Video watermarking*

*Watermark* disisipkan pada gambar bergerak atau disebut dengan video digital

### 2.1.3 *Type Watermarking*

a. *Visible*

Dikatakan *visible* karena jenis *watermark* dapat dilihat oleh panca indra manusia (mata telanjang). Sifat *watermark* ini sangat kuat bahkan sangat sulit dihapus keberadaannya walaupun tidak menjadi bagian dari image. Sebagai contohnya adalah logo transparan dari stasiun televisi yang ada pada sudut kanan atau kiri atas televisi (Hakim, 2012).

b. *Invisible*

Dikatakan *invisible* karena *watermark* jenis ini tidak dapat dilihat oleh panca indera, yang bertujuan memberikan informasi yang bersifat rahasia dan untuk melindungi hak cipta orang lain dari orang yang tidak bertanggung jawab.

Jenis *watermark* ini dapat dilihat melalui proses komputasi, dengan cara mengekstrak gambar yang ter*watermark* (Hakim, 2012).

#### 2.1.4 Klasifikasi *Watermarking*

Klasifikasi teknik *watermarking* digital pada saat ini cukup banyak. Ada beberapa teknik *watermarking* berdasarkan *domain* kerjanya (Hakim, 2012) yaitu:

a. Teknik *watermarking* yang bekerja pada *domain* spasial (*spatial domain watermarking*).

Teknik ini bekerja dengan cara menyisipkan *watermark* secara langsung kedalam domain spasial dari suatu citra. Istilah domain spasial sendiri mengacu pada piksel-piksel penyusun sebuah citra. Teknik *watermarking* jenis ini beroperasi secara langsung pada piksel-piksel tersebut. Beberapa contoh teknik yang bekerja pada domain spasial adalah teknik penyisipan pada *Least Significantbit (LSB)*.

b. Teknik *watermarking* yang bekerja pada domain *transform* (*transform domain watermarking*).

Pada *transform domain watermarking* (sering juga disebut dengan *frequency domain watermarking*) ini penyisipan *watermark* dilakukan pada koefisien frekuensi hasil transformasi citra asalnya. Ada beberapa transformasi yang umum digunakan oleh para peneliti, yaitu: *discrete cosine transform (DCT)*, *discrete fourier transform (DFT)*, *discrete wavelet transform (DWT)* maupun *discrete laguerre transform (DLT)*.

c. Teknik *watermarking* yang bekerja pada kedua domain diatas (*hybrid techniques watermarking*).

Teknik *watermarking* jenis ini bekerja dengan menggabungkan kedua teknik diatas. Pada teknik ini biasanya penanaman *watermark* dilakukan pada *domain* frekuensi beberapa bagian citra yang dipilih berdasarkan karakteristik spasial citra tersebut.

## 2.2 Transformasi *Wavelet* (*Wavelet Transform*)

Transformasi *wavelet* adalah sebuah transformasi matematika yang digunakan untuk menganalisis sinyal bergerak. Sinyal bergerak ini dianalisis untuk didapatkan informasi spektrum frekuensi dan waktunya secara bersamaan. Salah satu seri pengembangan transformasi *wavelet* adalah *Discrete Wavelet transform* (DWT) (Alfatwa, 2009).

Transformasi sinyal merupakan bentuk lain dari penggambaran sinyal yang tidak mengubah isi informasi dalam sinyal tersebut. Transformasi *wavelet* (*wavelet transform*) menyediakan penggambaran frekuensi waktu dari sinyal. Pada awalnya, transformasi *wavelet* digunakan untuk menganalisis sinyal bergerak (*non-stationary signals*). Sinyal bergerak ini dianalisis dalam transformasi *wavelet* dengan menggunakan teknik analisis *multiresolution*. Secara umum teknik analisis *multiresolution* adalah teknik yang digunakan untuk menganalisis frekuensi dengan cara frekuensi yang berbeda dianalisis menggunakan resolusi yang berbeda. Resolusi dari sinyal merupakan ukuran jumlah informasi di dalam sinyal yang dapat berubah melalui operasi filterisasi (Hakim, 2012).

Transformasi *wavelet* merupakan perbaikan dari transformasi *fourier*. Pada transformasi *fourier* hanya dapat menentukan frekuensi yang muncul pada suatu sinyal, namun tidak dapat menentukan kapan (dimana) frekuensi itu muncul. Dengan kata lain, transformasi *fourier* tidak memberikan informasi tentang *domain* waktu (*time domain*). Kelemahan lain dari transformasi *fourier* adalah perubahan sedikit terhadap sinyal pada posisi tertentu akan berdampak atau mempengaruhi sinyal pada posisi lainnya. Hal ini disebabkan karena transformasi *fourier* berbasis  $\sin$   $\cos$  yang bersifat periodik dan kontinu (Alfatwa, 2009).

Transformasi *wavelet* memiliki dua seri dalam pengembangannya yaitu *Continous Wavelet transform* (CWT) dan *Discrete Wavelet transform* (DWT). Semua fungsi yang digunakan dalam transformasi CWT dan DWT diturunkan dari *wavelet* melalui translasi atau pergeseran dan penskalaan atau kompresi. *Wavelet* merupakan fungsi dasar yang digunakan dalam transformasi *wavelet*. *Wavelet* menghasilkan semua fungsi *wavelet* yang digunakan dalam transformasi melalui translasi dan penskalaan, maka *wavelet* juga akan menentukan karakteristik dari

transformasi *wavelet* yang dihasilkan. Oleh karena itu, perlu pencatatan secara teliti terhadap penerapan *wavelet* dan pemilihan yang tepat terhadap *wavelet* harus dilakukan agar dapat menggunakan transformasi *wavelet* secara efisien. (Alfatwa, 2009)

### 2.2.1 Discrete Wavelet Transform

Dasar dari DWT dimulai pada tahun 1976 dimana teknik untuk mendekomposisi sinyal waktu discrete ditemukan. Didalam CWT, sinyal dianalisis menggunakan seperangkat fungsi dasar yang saling berhubungan dengan penskalaan dan transisi sederhana. Sedangkan di dalam DWT, penggambaran sebuah skala waktu sinyal digital didapatkan dengan menggunakan teknik filterisasi digital. Secara garis besar proses dalam teknik ini adalah dengan melewati sinyal yang akan dianalisis pada filter dengan frekuensi dan skala yang berbeda.

Filterisasi sendiri merupakan sebuah fungsi yang digunakan dalam pemrosesan sinyal. *Wavelet* dapat direalisasikan menggunakan iterasi filter dengan penskalaan. Resolusi dari sinyal, yang merupakan rata-rata dari jumlah detail informasi dalam sinyal, ditentukan melalui filterisasi ini dan skalanya didapatkan dengan *upsampling* dan *down sampling (subsampling)* (Hakim, 2012).

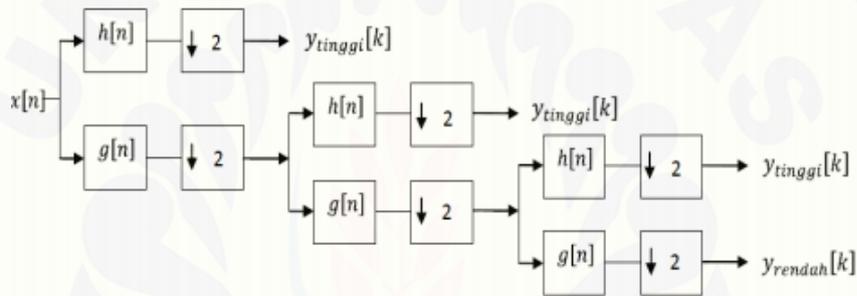
Pembagian sinyal menjadi frekuensi tinggi dan frekuensi rendah dalam proses filterisasi *highpass filter* dan *lowpass filter* disebut sebagai dekomposisi. Proses dekomposisi dimulai dengan melewati sinyal asal melewati *highpass filter* dan *lowpass filter*. Misalkan sinyal asal ini memiliki rentang frekuensi dari 0 sampai dengan  $\pi$  rad/s. Dalam melewati *highpass filter* dan *lowpass filter* ini, rentang frekuensi *subsample* menjadi dua, sehingga rentang frekuensi tertinggi pada masing-masing *subsample* menjadi  $\pi/2$  rad/s. Setelah filterisasi, setengah dari *sample* atau salah satu *subsample* dapat dieliminasi berdasarkan aturan *nyquist*. Sehingga sinyal dapat selalu di *subsample* dengan cara mengabaikan setiap *sample* yang kedua (Hakim, 2012).

Proses dekomposisi ini dapat melalui satu atau lebih tingkatan. Dekomposisi satu tingkat ditulis dengan ekspresi matematika (Hakim, 2012) pada persamaan (2.3) dan (2.4) :

$$Y_{tinggi}\{k\} = (x + a)^n = \sum_n x[n]h[2k - n] \dots\dots\dots(2.3)$$

$$Y_{rendah}\{k\} = (x + a)^n = \sum_n x[n]g[2k - n] \dots\dots\dots(2.4)$$

Y tinggi [k] dan Y rendah [k] yang merupakan hasil *highpass filter* dan *lowpass filter*, k adalah koefisien DWT, x[n] merupakan sinyal asal, h[n] adalah *highpass filter* dan g[n] adalah *lowpass filter*. Untuk dekomposisi lebih dari satu tingkat, prosedur pada persamaan (2.3) dan (2.4) dapat digunakan pada masing masing tingkatan. Contoh penggambaran dekomposisi dipaparkan pada gambar 2.1 dengan menggunakan dekomposisi *wavelet* tiga tingkat:



Gambar 2.1 Dekomposisi *wavelet* tiga tingkat

Pada Gambar diatas, Y<sub>tinggi</sub>[k] dan Y<sub>rendah</sub>[k] yang merupakan hasil dari *high pass filter* dan *lowpass filter*, Y<sub>tinggi</sub> [k] disebut sebagai koefisien DWT. Y<sub>tinggi</sub> [k] merupakan detil dari informasi sinyal, sedangkan Y<sub>rendah</sub> [k] merupakan taksiran kasar dari fungsi penskalaan. Nilai *wavelet* menggunakan koefisien DWT ini maka dapat dilakukan proses *inverse discrete wavelet transform* (IDWT) untuk merekonstruksi menjadi sinyal asal (Hakim, 2012).

$$X[n] = k(Y_{tinggi} [k] h[-n + 2 k] + Y_{rendah} [k] g[-n + 2 k] ) \dots\dots\dots(2.5)$$

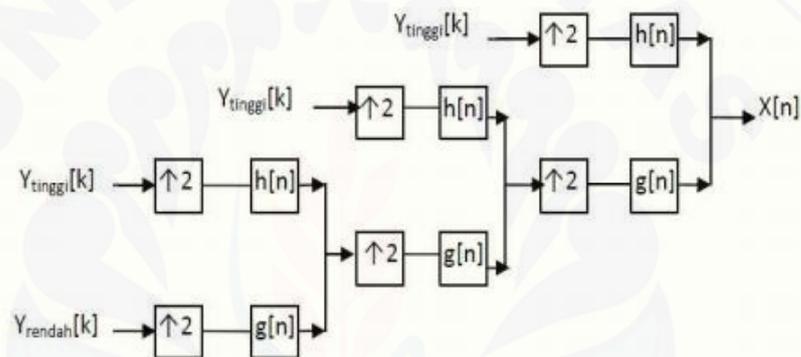
proses rekontruksi merupakan kebalikan dari proses dekomposisi sesuai dengan tingkatan pada proses dekomposisi.

Untuk mendapatkan hasil rekonstruksi setelah didekomposisi maka langkah awal proses rekonstruksi diawali dengan menggabungkan koefisien DWT dari yang

berada pada akhir dekomposisi dengan sebelumnya meng-upsample melalui *highpass filter* dan *lowpass filter*. Proses rekonstruksi ini sepenuhnya merupakan kebalikan dari proses dekomposisi sesuai dengan tingkatan pada proses dekomposisi. Sehingga persamaan rekonstruksi pada masing-masing tingkatan (Hakim, 2012) dapat ditulis:

$$X[n] = \sum_k (Y_{\text{tinggi}}[k]h[-n + 2k] + Y_{\text{rendah}}[k]g[-n + 2k]) \quad \dots\dots\dots(2.6)$$

Proses rekonstruksi *wavelet* untuk mendapatkan sinyal asal dengan tiga tingkatan digambarkan pada Gambar 2.2



Gambar 2.2 Rekonstruksi *wavelet* 3 tingkat

### 2.3 Error Detection And Correction

Pada saat data berada dalam *transmission system* terdapat kemungkinan data *error*. Data *error* tersebut akan diperbaiki oleh *receiver* melalui proses *error detection* dan *error correction*. Proses *error detection* dilakukan oleh transmitter dengan cara menambahkan beberapa *bit* tambahan ke dalam data yang akan ditransmisikan. Proses *error detection* dan *correction* ini sering dijumpai pada telepon selular secara garis besar, metode pendeteksi *error* (*error detection*) dan pengontrol *error* (*error controller*) dapat dibagi menjadi dua bagian besar yaitu *Error detection and correction* terdiri dari dua metoda (Lubis, 2012) yaitu:

- a. Manual, dengan menggunakan operator.
- b. *Automatic*

### 2.3.1 Error Correcting Code

#### 2.3.1.1 BCH Code

Bose, Chaudhuri, and Hocquenghem (BCH) code merupakan sebuah metode *error correction* yang dibangun pada bidang finite (terbatas). Kode ini merupakan generalisasi dari *hamming code* untuk *multiple error correction*. Kode BCH diperkenalkan pertama kali oleh A. Hocquenghem pada tahun 1959 dan secara terpisah pada tahun 1960 oleh R. C. Bose dan Ray-Chaudhuri. Kode BCH merupakan *Cyclic codes* dimana beberapa simbol tersusun dari  $m$ -bit yang berurutan, dimana  $m$  adalah integer positif yang lebih besar dari 2. Pada *binary BCH code* terdapat beberapa parameter sebagai berikut:

Panjang blok :  $n = 2^m - 1$

Jumlah digit *parity check* :  $n - k \leq mt$

Jarak minimal :  $d_{\min} \geq 2t + 1$

Kode ini mampu mengoreksi berbagai kombinasi dari jumlah kesalahan ( $t$ ) atau kesalahan lebih kecil dalam blok  $n$  digit. Kita menyebutnya kode BCH  $t$  *error correcting* (Sutarto, 2014).

## 2.4 MATLAB

Matlab merupakan salah satu bahasa pemrograman dengan unjuk kinerja tinggi (*high-performance*) untuk komputasi teknis, yang menintegrasikan komputasi, visualisasi, dan pemrograman didalam lingkungan yang mudah penggunaannya dalam memecahkan persoalan dengan solusinya yang dinyatakan dengan notasi matematik. Penggunaan MATLAB (Hakim, 2012), yaitu:

1. Matematika dan komputasi
2. Pengembangan algoritma
3. Pemodelan, simulasi dan pembuatan 'prototipe'
4. Analisa data, eksplorasi dan visualisasi
5. Grafik untuk sains dan teknik
6. Pengembangan aplikasi, termasuk pembuatan antarmuka grafis untuk
7. pengguna (*Graphical User Interface*)

MATLAB adalah sebuah sistem interaktif yang menggunakan elemen data dasarnya adalah *array* yang tidak membutuhkan dimensi. Hal ini mempermudah anda untuk menyelesaikan masalah komputasi, terutama yang menyangkut matriks dan vektor. MATLAB merupakan singkatan dari '*matrix laboratory*'. Pada awalnya MATLAB dibuat untuk mempermudah pengembangan perangkat lunak berbasis matriks oleh proyek LINPACK dan EISPACK. Fitur-fitur MATLAB untuk menyelesaikan spesifik disebut *toolboxes*. *Toolboxes* adalah koleksi komprehensif dari fungsi-fungsi MATLAB (*M-files*) yang memperlebar lingkungan MATLAB dalam menyelesaikan kelas-kelas tertentu dari permasalahan. Beberapa *toolboxes* yang tersedia meliputi bidang: pengolahan sinyal, sistem kendali, jaringan syaraf (*neural network*), logika *fuzzy*, *wavelet*, simulasi dan lain sebagainya (Hakim, 2012)

#### 2.4.1 Aplikasi M-file

Pada saat membuat program GUI, matlab membuat program aplikasi berupa *M-file* yang menyediakan kerangka untuk mengontrol GUI. Kerangka ini dapat membantu membuat program menjadi lebih efisien dan sempurna. Fungsi *M-file* mirip dengan script *M-file* dimana keduanya merupakan *file* teks dengan ekstensi dot m(m). sebagaimana *script M-file*, fungsi *M-file* tidak dimasukkan ke dalam jendela *command*, tetapi merupakan suatu *file* tersendiri yang dibuat dengan editor teks. Fungsi *M-file* menyediakan cara sederhana untuk menambah kemampuan MATLAB, bahkan banyak fungsi standar MATLAB yang merupakan fungsi *M-file* sederhana untuk menambah kemampuan MATLAB, bahkan banyak fungsi standar MATLAB yang merupakan fungsi *M-file*.

## 2.5 Citra Digital

Citra atau image merupakan istilah lain untuk gambar. Data atau informasi tidak hanya disajikan dalam bentuk teks akan tetapi bisa juga berupa citra. Citra digital adalah citra yang dapat diolah oleh komputer. Data berupa citra mempunyai karakteristik yang berbeda dengan bentuk data yang lainnya, yakni banyak informasi yang dapat diambil dari sebuah citra. Contoh citra digital pada gambar 2.3.



Gambar 2.3 Contoh citra digital (*sample* MATLAB)

Secara matematis citra merupakan suatu fungsi intensitas cahaya pada bidang 2-dimensi sehingga dapat disimbolkan dengan  $f(x, y)$  dengan  $(x, y)$  merupakan titik kordinat pada bidang 2-dimensi dan  $f(x, y)$  merupakan nilai intensitas cahaya pada titik tersebut. Simbol  $x$ ,  $y$  dan  $f(x, y)$  bernilai diskrit (Adriansyah, 2011)

### 2.5.1 Jenis Citra Digital

#### 1. Citra Biner (*Monochrome*)

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai *pixel* yaitu hitam dan putih. Dimana gradasi warna hitam= 0, dan putih = 1. Pada standar citra untuk ditampilkan dilayar komputer, nilai biner ini berhubungan dengan ada tidaknya cahaya yang ditembakkan oleh electron gunyang terdapat didalam monitor komputer. Angka 0 menyatakan tidak ada cahaya, dengan demikian warna yang direpresentasikan adalah hitam. Untuk angka 1 terdapat cahaya, sehingga warna yang direpresentasikan adalah putih.

Standar tersebut disebut sebagai standar citra cahaya, sedangkan standar citra tinta/cat adalah berkebalikan, karena biner tersebut menyatakan ada tidaknya tinta. Gambar 2.4. memperlihatkan contoh dari citra biner (Hakim, 2012).



Gambar 2.4 Citra biner

## 2. Citra *Grayscale* (Scala Keabuan)

Pada citra ini warna tergantung pada jumlah *bit* yang disediakan oleh memori untuk menampung kebutuhan warna ini. Misalnya 2 *bit* (2<sup>2</sup>) mewakili 4 warna, 3*bit* (2<sup>3</sup>) mewakili 8 warna, dan seterusnya sampai maksimal 8 *bit* (2<sup>8</sup>) yang mewakili 256 warna. Semakin besar jumlah *bit* warna yang disediakan memori, maka semakin halus gradiasi warna yang terbentuk. Gambar 2.5 memperlihatkan contoh citra *grayscale* (Hakim, 2012).



Gambar 2.5 Citra grayscale 4 *bit* (Hakim, 2012)

## 3. Citra Warna

Setiap *pixel* pada citra warna mewakili warna yang merupakan kombinasi dari 3 warna dasar (RGB= *red*, *green*, *blue*). Setiap warna dasar menggunakan penyimpanan 8 *bit* = 1 byte, yang berarti warna mempunyai gradiasi sebanyak 256 warna. Berarti setiap *pixel* mempunyai kombinasi warna sebanyak  $28.28.28 = 16$  juta warna lebih (8). Itulah sebabnya mengapa disebut citra *true color* karena mempunyai jumlah warna yang cukup besar sehingga bisa dikatakan hampir mencakup semua warna di alam. Contoh citra warna seperti pada gambar 2.6.

Dalam citra warna (*true color*) penyimpanan didalam memori berbeda dengan penyimpanan pada grayscale. Setiap *pixel* dari citra *grayscale* 256 gradiasi warna diwakili oleh 1 byte, sedangkan pada 1 *pixel* citra *true color* diwakili oleh 3 *byte*, dimana masing-masing data *byte* mempresentasikan warna merah (*red*), hijau (*green*), dan biru (*blue*) (Hakim, 2012).



Gambar 2.6 Citra warna

### 2.5.2 Transformasi Citra

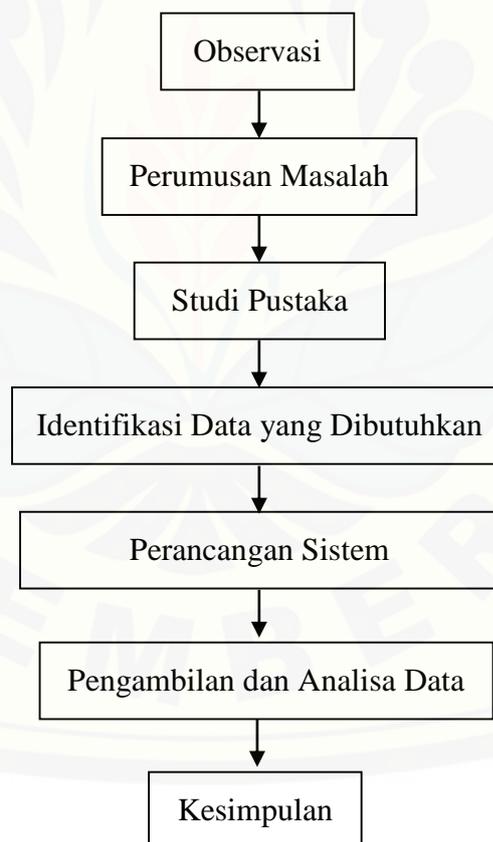
Transformasi citra dapat diartikan sebagai perubahan bentuk suatu citra. Perubahan bentuk tersebut dapat berupa perubahan geometri piksel seperti perputaran (rotasi), pergeseran (translasi), penskalaan atau berupa perubahan ruang (domain) citra ke domain lainnya. Transformasi citra merupakan pokok bahasan yang sangat penting dalam pengolahan citra. Citra hasil proses transformasi dapat dianalisis kembali, diinterpretasikan, dan dijadikan acuan untuk melakukan pemrosesan selanjutnya. Tujuan diterapkannya transformasi citra adalah untuk memperoleh informasi (*feature ex-traction*) yang lebih jelas yang terkandung dalam suatu citra. Melalui proses transformasi, suatu citra dapat dinyatakan sebagai kombinasi linear dari sinyal dasar (*basic signals*) yang sering disebut dengan fungsi basis (*basic function*). Suatu citra yang telah mengalami transformasi dapat diperoleh kembali dengan menggunakan transformasi balik (*invers transformation*). Pada transformasi citra terdapat berbagai macam transformasi diantaranya, *Fast Fourier Transform* (FFT), *discrete Cosine Transform* (DFT), *Discrete Wavelet Transform* (DWT) (Hakim, 2012).

### BAB 3. METODE PENELITIAN

Pembahasan pada bab metode penelitian ini dijelaskan beberapa hal pokok yaitu obyek penelitian, tahap penelitian, tempat dan waktu penelitian, alat dan bahan, langkah-langkah dalam pengambilan data dan manajemen penelitian di lapangan, pengolahan data serta *software* yang digunakan dalam penelitian.

#### 3.1 Tahap Penelitian

Penyusunan laporan ini memiliki beberapa tahap untuk memperoleh informasi yang dibutuhkan, adapun tahap pengambilan data yang digunakan dalam penelitian ini adalah sebagai berikut :



Gambar 3.1 Tahap penelitian

Tahap penelitian ini dilakukan observasi dengan melihat dan menentukan tema objek yang akan diteliti. Selanjutnya merumuskan suatu masalah dari beberapa masalah yang terdapat dalam kehidupan sehari-hari yang berhubungan dengan objek yang diteliti. Tahap berikutnya mengumpulkan dan mempelajari literatur tentang *watermarking* citra dengan menggunakan DWT. Setelah mengumpulkan landasan teori hal yang perlu dilakukan adalah mengidentifikasi data yang dibutuhkan untuk menganalisa performa dari sistem yang akan di buat. Setelah itu kita dapat menentukan *software* yang digunakan. Tahap selanjutnya merancang atau membuat aplikasi *watermarking* citra dengan menggunakan metode DWT. Jika telah selesai dirancang tahap selanjutnya yaitu pengujian aplikasi pada *software*. Pada penelitian ini menggunakan *software* MATLAB Selanjutnya untuk pengambilan data dilakukan dengan menggunakan jenis *wavelet* yang digunakan . Kemudian diberi *noise* untuk perbedaan hasil dari citra sendiri. Semua pengambilan data menggunakan teknik PSNR dan MSE. Tahap akhir dari penelitian ini yaitu membuat kesimpulan.

### 3.2 Alat dan Bahan

Adapun alat dan bahan yang perlu dipersiapkan untuk melakukan penelitian adalah sebagai berikut :

1. PC / Laptop

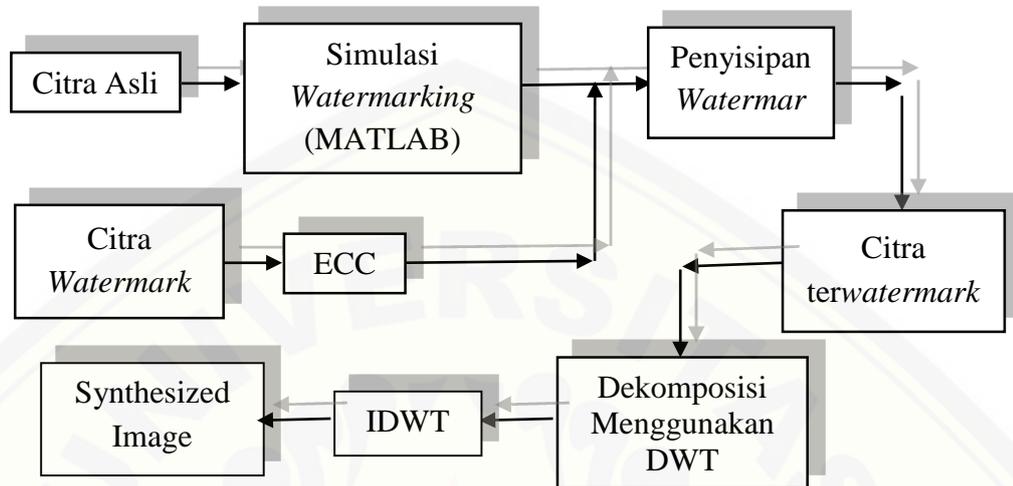
PC / Laptop digunakan untuk melakukan simulasi rancangan sistem serta melakukan analisis hasil simulasi.

2. *Software* MATLAB R2012b

MATLAB R2012b digunakan untuk merancang simulasi dengan menggunakan GUI MATLAB yang akan digunakan unuk membuat aplikasi *watermarking* pada citra

### 3.3 Blok Diagram

#### 3.3.1 Proses Penyisipan Citra *Watermark* dan DWT



Gambar 3.2 Diagram alur proses *watermarking*

Gambar 3.2 merupakan gambar diagram blok dari skema yang akan di buat, diagram tersebut dibagi beberapa bagian, sebagai berikut :

a. Citra Asli

Pada pemilihan citra ini dengan menggunakan citra warna RGB atau keabuan (*grayscale*). Pemilihan citra berdasarkan tingkat gradiasi warna seperti tingkat kekontrasan, kombinasi warna, dll

Berikut contoh citra yang digunakan :

Tabel 3.1 Citra asli

No	Nama Citra	Hasil Citra
1	telukijo.jpg	

2

gunung.jpg



---

b. Simulasi *Watermarking* MATLAB

Simulasi *watermarking* digunakan untuk menyisipkan dan ekstraksi citra. Simulasi ini menggunakan MATLAB 2012b berfungsi sebagai media yang menyisipkan citra atau text sehingga menghasilkan citra yang ter*watermark*

c. Citra *Watermark*

Pada pemilihan citra *watermark* ini menggunakan citra hitam putih. Pada penelitian ini diharuskan saat pemilihan citra *watermark* harus mempunyai resolusi lebih kecil dari citra asli agar bisa diproses lebih cepat

**DA**

Gambar 3.3 Citra *watermark* yang digunakan

d. *Error Correction Code*

Sebelum citra *watermark* didisipkan citra *watermark* dilakukan proses ECC terlebih dahulu. Proses ECC dilakukan dengan cara menambahkan beberapa *bit* tambahan ke dalam data yang akan ditransmisikan, proses ini bertujuan agar citra *watermark* saat di ekstrak nanti tidak rusak atau masih sama ketahanannya seperti sebelum didisipkan. Metode yang digunakan pada proses ECC ini adalah *BCH Code*.

e. Penyisipan *Watermark*

Pada tahap ini dilakukan penyisipan citra *watermark* ke citra asli menggunakan aplikasi *watermarking* pada matlab. Proses penyisipan atau penanaman pada citra asli menggunakan 2 objek. Pertama menggunakan citra *watermark* dan kedua menggunakan *text*

f. Citra *Watermark*

Pada proses ini merupakan hasil citra yang telah disisipkan atau ditanamkan oleh citra *watermark* maupun *text*

g. Dekomposisi Menggunakan DWT

Setelah mendapatkan citra ter*watermark* maka langkah selanjutnya mendekomposisikan citra asli menggunakan DWT dan jenis *wavelet* yang digunakan dekomposisi citra digunakan untuk mendapatkan nilai dan parameter yang nantinya digunakan untuk membandingkan citra asli dan citra ter*watermark*. Untuk mendapatkan nilai parameter yang diinginkan data didapatkan dengan menggunakan *software* MATLAB. Pada proses dekomposisi ini dilakukan dengan dua tahap, tahap pertama melakukan proses dekomposisi citra pada pada seluruh baris dan tahap kedua dilakukan proses dekomposisi citra dalam arah kolom menggunakan hasil citra tahap pertama. Pada saat dekomposisi akan menghasilkan LL, LH, HL dan HH. LL adalah koefisien yang didapatkan dari *low pass*. Citra ini merupakan citra versi lebih halus dari citra aslinya yang disebut komponen aprokmasinya. LH adalah bagian koefisien yang didapat dari proses tapis *low pass* kemudian dilanjutkan *high pass*. LH menunjukkan citra tepi arah *Horizontal*. HL merupakan proses diawali proses *high pass* lalu dilanjutkan *low pas*. HL menunjukkan koefisien citra tepi arah *vertical*. HH merupakan proses *high pass* lalu dilanjutkan dengan *high pass*. HH menunjukkan koefisien citra tepi arah *diagonal*.



Gambar 3.4 Dekomposisi citra

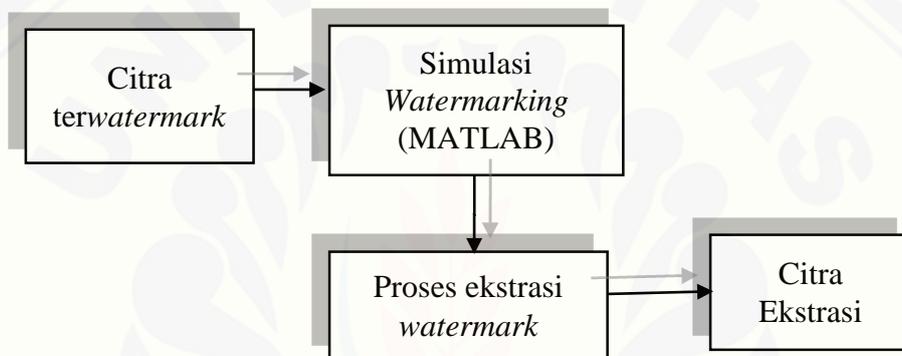
h. *Invers Discreate Wavelet Transform (IDWT)*

Pada proses ini dilakukan setelah proses dekomposisi dan proses mendapatkan nilai yang diinginkan dengan cara mentransformasikan kembali koefisien koefisien yang telah dianalisa menggunakan DWT sehingga mendapatkan hasil citra yang diinginkan

i. *Synthesized Image*

Hasil dari proses IDWT

### 3.3.2 Proses Ekstrasi Citra



Gambar 3.5 Proses ekstrasi citra

Gambar 3.5 merupakan gambar blok diagram yang akan dibuat untuk proses ekstrasi. Diagram ini terbagi menjadi beberapa bagian, sebagai berikut :

a. *Citra terwatermark*

Citra ini berasal dari proses penanaman atau penyisipan citra *watermark* ke dalam citra asli melalui aplikasi *watermarking*

b. *Simulasi watermarking (MATLAB)*

Simulasi ini digunakan untuk mengekstrasi citra *watermark*. Aplikasi ini juga sama seperti yang digunakan saat proses pembentukan citra *terwatermark*

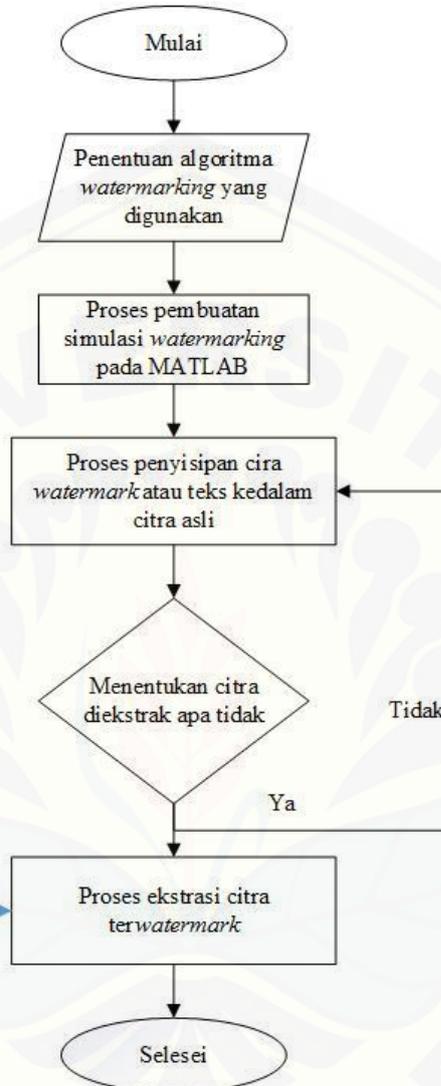
c. *Proses ekstrasi*

Pada tahap ini dilakukan untuk menampilkan kembali citra yang ditanamkan atau disisipkan dengan menggunakan aplikasi *watermarking* tersebut

d. *Citra Ekstrasi*

Menampilkan kembali citra *watermark*

### 3.4 Flowchart Tahapan Penyisipan Citra dan Ekstraksi



Gambar 3.6 Flowchart proses watermarking

Gambar 3.6 merupakan gambar langkah-langkah dalam *watermarking* citra dengan mengawali penentuan algoritma yang digunakan. Setelah menentukan algoritma langkah selanjutnya pembuatan simulasi *watermarking*. Simulasi ini bertujuan untuk proses penyisipan dan ekstrasi citra. Hasil dari proses penyisipan kemudian dianalisis menggunakan *discreate wavelet transform* atau DWT dengan mengaplikasikan jenis *wavelet* yang digunakan dan parameter parameter yang ada

```

img=imread(img_in);
st='';
watermark=get(handles.edit5,'String');
global key
global ext_out
global w_type
imgw=0;
k_plot=1;

if key==0
    if w_type==0
        [imgw st]=watermark_k(img,watermark);
    else
        [imgw st]=watermark_img(img,imread(watermark));
        k_plot=2;
    end
end

```

Gambar 3.7 Proses penyisipan citra

Gambar 3,7 menjelaskan bagaimana proses menyisipkan citra *watermark*. *Img* pada gambar adalah hasil dari inputan citra asli (*img\_in*) yang akan disisipkan citra *watermark* (*watermark\_img*). Gambar 3.7 menjelaskan proses penyisipan dilakukan dengan dua cara. Pertama menggunakan citra *watermark* (*watermark\_img*) dan kedua menggunakan teks (*watermark\_k*).

```

img_out=get(handles.edit3,'String');
imwrite(imgw,[img_out '.' ext_out],ext_out);
set(handles.text8,'String',st);
figure
subplot(k_plot,2,1)
imshow(img)
title(['Citra Original'])
subplot(k_plot,2,2)
imshow(imgw)
title(['Citra Watermark'])

```

Gambar 3.8 Proses menampilkan kembali hasil penyisipan

Gambar 3.8 menjelaskan tentang proses menampilkan hasil penyisipan citra ter*watermark* (*imgw*). “*img\_out*” adalah inisiasi nama sebagai tempat untuk proses penyisipan. Hasil “*img\_out*” diproses dengan perintah “*imwrite*” untuk

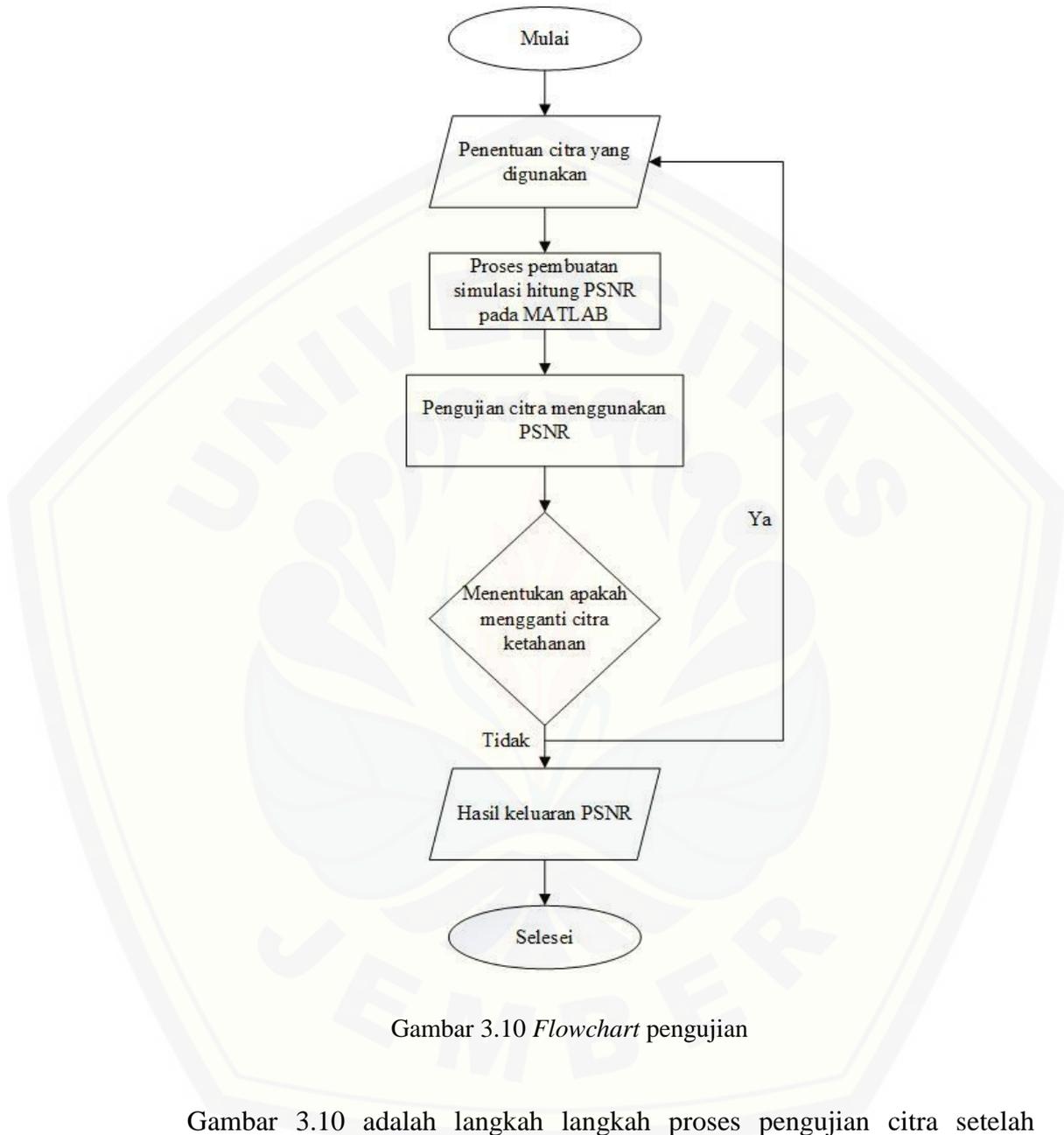
meneruskan perintah selanjutnya agar ditampilkan oleh MATLAB. Untuk menampilkan hasil penyisipan dilakukan dengan menggunakan perintah “imshow” sesuai letak yang diinginkan.

```
img=get(handles.edit1,'String');
if isempty(img)
    set(handles.edit3,'String','Tidak ada file');
    return;
end
global key_d
global type
text='';
if key_d==0
    if type==0
        st=dewatermark_k(imread(img));
    else
        img_logo=dewatermark_img(imread(img));
    end
end
```

Gambar 3.9 Proses ekstrasi

Gambar 3.9 adalah proses *dewatermarking* atau ekstrasi. Pada gambar 3.9 dijelaskan “img” adalah citra terwatermark yang dihasilkan melalui penyisipan citra. Hasil proses pada “img” didapatkan dari pengambilan citra melalui *string* atau nama citra terwatermark yang sudah disimpan dari proses sebelumnya. Hasil ekstrasi pada gambar 3.9 ada dua macam. Pertama adalah hasil ekstrasi berupa teks (*dewatermark\_k*) dan berupa citra (*dewatermark\_img*). Hasil tersebut didapatkan pada saat menginputkan data berupa citra atau teks pada proses penyisipan.

### 3.5 Flowchart Tahapan Pengujian Ketahanan Citra



Gambar 3.10 Flowchart pengujian

Gambar 3.10 adalah langkah langkah proses pengujian citra setelah melalui proses *watermarking*. Ada tiga dalam proses pengujian dilakukan. Pertama menganalisis citra menggunakan PSNR, kedua menganalisis citra setelah citra diberi noise, ketiga menganalisis citra setelah nilai kecerahan citra terwatermark diubah. Pada analisis kedua dan ketiga masing masing hasil citra dicari menggunakan PSNR juga. Masing masing pengujian memiliki algoritma

berbeda. Pengujian dilakukan diawali dengan menentukan jenis citra yang digunakan. Setelah ditentukan langkah selanjutnya membuat simulasi PSNR. Simulasi ini bertujuan untuk pengambilan data dari citra yang ditentukan. Hasil dari PSNR kemudian dianalisis menggunakan metode yang diinginkan.

### 3.5.1 Tahapan pengambilan data PSNR

```
%Read Original & Distorted Images  
origImg = imread('lenna.png');  
distImg = imread('lennal.png');
```

Gambar 3.11 Pengambilan citra asli dan rekontruksi

Gambar 3.11 adalah proses pengambilan citra asli dan rekontruksi. Citra rekontruksi adalah hasil citra yang telah melewati proses *watermarking* atau penyisipan. Gambar 3.11 menjelaskan “origImg” adalah citra asli dan “distImg” adalah citra rekontruksi.

```
origImg = double(origImg);  
distImg = double(distImg);  
[M N] = size(origImg);  
error = origImg - distImg;  
MSE = sum(sum(error .* error)) / (M * N);  
if(MSE > 0)  
    PSNR = 10*log(255*255/MSE) / log(10);  
else  
    PSNR = 99;  
end
```

Gambar 3.12 Perhitungan nilai MSE dan PSNR

Gambar 3.12 menjelaskan tentang proses perhitungan MSE dan PSNR pada citra. Rumus perhitungan dapat dilihat pada persamaan 2.1 dan 2.2. Hasil dari perumusan citra akan muncul pada *command window* dan simulasi pada MATLAB.

### 3.5.2 Tahapan pengambilan data ketahanan citra

Pada pengujian ini bertujuan untuk melihat apakah citra *watermark* yang telah diubah dari bentuk aslinya dapat diekstrak kembali atau tidak. Ada dua cara untuk mengubah bentuk asli citra. Pertama menggunakan *noise gaussian* untuk merusak citra dan kedua dengan cara mengubah tingkat kekontrasan citra. Pengujian hasil citra pada tahap ini menggunakan hasil perhitungan PSNR seperti gambar 3.11 dan 3.12.

```
a=imread('sail.png');
b=imnoise(a,'gaussian',0.3,0.08);
%imwrite(b,'coba.png');
figure, imshow(b)
```

Gambar 3.13 Proses pemberian citra *noise gaussian*

Pada gambar 3.13 menjelaskan proses pemberian *noise* pada citra. “A” adalah inisiasi nama untuk citra asli yang akan digunakan. “B” adalah inisiasi nama untuk memberikan nilai koefisien noise pada citra. Semakin besar nilai yang diberi semakin rusak citra tersebut. Hasil citra *noise* dianalisis menggunakan PSNR seperti gambar 3.11 dan 3.12

```
%mengambil data gambar
tmp = imread('lenna.png');
%untuk mengatur konsentrasi brightnes
brigthVal = 150;
%lakukan pembacaan pixel sebesar dimensi gamb
for i=1:512
    for j=1:512
        %manipulasi matrix untuk memberikan b:
        img(i,j,1) = tmp(i,j,1) + brigthVal;
        img(i,j,2) = tmp(i,j,2) + brigthVal;
        img(i,j,3) = tmp(i,j,3) + brigthVal;
    end
end
```

Gambar 3.14 Proses mengubah kekontrasan citra

Pada gambar 3.14 menjelaskan proses mengubah kekontrasan citra. “tmp” adalah inisiasi nama untuk citra asli yang akan diubah. “brigthVal” adalah nilai kecerahan atau kekontrasan yang membuat citra semakin terang atau gelap. Nilai yang akan digunakan untuk tahap ini yaitu -75, 100 dan 175. Semakin kecil nilai

yang digunakan maka semakin gelap citra yang akan diubah. Semakin besar nilai yang digunakan maka semakin terang citra yang akan diubah. Hasil dari proses tersebut kemudian dicari nilai PSNRnya untuk dianalisis seperti gambar 3.11 dan 3.12.

### 3.6 Metode Pengujian

Pada sub bab ini akan dilakukan metode pengujian untuk mengetahui hasil citra yang telah di*watermark*. Adapun citra yang akan digunakan adalah citra *color* (RGB) dan *grayscale*. Pada saat pemilihan citra *watermark*, citra *watermark* harus dipastikan citra *watermark* resolusinya lebih kecil dari pada citra asli untuk mempercepat proses *watermarking* Pada penelitian ini citra yang digunakan sejumlah 3 buah sesuai tabel 3.1 dan gambar 3.3. Pemilihan citra berdasarkan tingkat gradiasi warna seperti tingkat kekontrasan dan kombinasi warna. Contoh citra dapat dilihat pada tabel 3.2

Tabel 3.2 Contoh citra yang akan diuji

No	Nama Citra	Jenis Citra	Hasil Citra
1	Telukijo.jpg	Citra Asli	
2	Gunung.jpg	Citra Asli	

### 3.6.1 Analisa Citra Asli dan *Watermark*

Pada metode pengujian ini dilakukan dengan tujuan menganalisa bagaimana perbedaan citra asli dan citra *watermark*. Citra tersebut kemudian dibandingkan dengan 10 orang atau 10 responden dengan tujuan apakah responden tersebut dapat melihat perbedaan secara visual antara citra asli dan *watermark*. Para responden menguji dengan menilai hasil gambar ter*watermark* secara subjektif. Cara menentukan citra ter*watermark* pada penelitian ini, dengan memberi tanda 1 untuk citra asli dan 2 untuk citra ter*watermark* seperti pada tabel 3.3.

1. Untuk koresponden memilih citra ditandai dengan kata (ya)
2. Untuk koresponden yang tidak memilih atau tidak melihat perbedaanya ditandai dengan kata (tidak)

Tabel 3.3 Survei citra terhadap responden

No	Responden	Ada perbedaan
1	Responden 1	
2	Responden 2	
3	Responden 3	
4	Responden 4	
5	Responden 5	
6	Responden 6	
7	Responden 7	
8	Responden 8	
9	Responden 9	
10	Responden 10	

### 3.6.2 PSNR

Pada pengujian ini menggunakan PSNR. PSNR merupakan salah satu parameter untuk melihat kualitas citra dengan cara menghitung nilai PSNR. Semakin besar nilai PSNR semakin bagus kualitas citra tersebut. Cara menghitung PSNR pertama yang harus dilakukan adalah menghitung nilai *Mean Squared Error* (MSE) dari citra tersebut, semakin kecil nilai MSE maka semakin baik citra tersebut.

Tabel 3.4 PSNR

No	Citra	Ukuran citra	MSE	PSNR (dB)
1				
2				
3				
4				

### 3.6.3 Analisa Ketahanan Citra

#### 1. *Noise Gaussian*

Dari pengujian sebelumnya melalui responden dan PNSR. Selanjutnya melakukan pengujian terhadap serangan-serangan ketahanan (*Robusness*) contohnya *noise*. Salah satu contoh dari *noise* pada penelitian ini yaitu *noise gaussian*. *Noise gaussian* bisa disebut juga *noise white noise* karena seluruh frekuensinya spektralnya berupa cahaya putih. Besar sedikitnya *noise* ini ditentukan nilai rata rata dan nilai variasi. Tujuan dari *noise* ini digunakan untuk melihat apakah ada yang berubah setelah disisipkan atau masih bisa dibaca atau tidak setelah disisipkan. Teknik pengujian ketahanan ini dipaparkan pada table 3.5 dengan menggunakan nilai PSNR sebagai rujukanya

Tabel 3.5 Pengujian setelah diberi *noise*

No	Citra	Ukuran citra	MSE	PSNR (dB)
1				
2				
3				
4				

## BAB 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat diambil beberapa kesimpulan diantaranya:

1. Hasil PSNR citra ter*watermark* yang bagus didapatkan dari citra yang memiliki gradiasi warna yang tinggi. Hal ini dapat dibuktikan pada tabel 4.9 dan 4.10 dimana ada perbandingan hasil PSNR antara citra ter*watermark* dan citra yang diberi *noise Gaussian*. Pada citra yang terkena *noise* memiliki PSNR yang kecil hal ini dapat diartikan citra yang terkena *noise* tidak layak pakai sesuai rujukan pada tabel 2.1
2. Pada pengujian menggunakan parameter *wavelet* dimana hasil yang didapat pada tiap komponen citra untuk nilai *mean absolute deviation* (MAD) untuk citra rekonstruksi mendekati nilai pada citra asli dapat diartikan citra rekonstruksi sangat bagus dan dapat digunakan untuk proses *watermarking*
3. Citra yang digunakan untuk proses *watermarking* sangat bagus dan layak digunakan untuk proses *watermarking* dengan nilai yang berkisar 50.6-52.9 dB. Hal dapat dibuktikan sesuai pada tabel 4.6 dengan acuan pada tabel 2.1

### 5.2 Saran

Untuk penelitian selanjutnya, penulis menyarankan beberapa hal agar penelitian ini dapat dikembangkan dan lebih bermanfaat. Adapun beberapa saran dari penulis adalah sebagai berikut:

1. Pada penelitian selanjutnya diharapkan dapat mengembangkan sistem *watermarking* yang lainnya seperti audio dan video sehingga dapat bermanfaat kedepannya
2. Penulis mengharapkan adanya perbandingan dengan jenis *wavelet* lainnya agar dapat dikembangkan kedepannya.

**DAFTAR PUSTAKA**

- Alfatwa, Fathony. 2012. *Watermarking Pada Citra Digital Menggunakan Discrete Wavelet Transform*. Bandung. Institute Teknologi Bandung. 10 Mei
- Betri, J. T., E. Suryani., dan A. Aziz. 2014. Video *Watermarking* Untuk Perlindungan Hak Cipta dengan Algoritma Koch Zhao. *Jurnal Itsmart*. 3(2)
- Felyana, M. 2013. *Watermarking Video Menggunakan Transformasi Wavelet Diskrit*. *Jurnal Generic*. 8(1) : 198-208
- Hakim, R. A. 2012. *Analisa Perbandingan Watermarking Image dengan menggunakan Metode Discrete Wavelet Transform*. Skripsi. Depok:Universitas Indonesia.
- Lubis, A. A. 2012. Perancangan *Error Detection System And Error Correction System* Menggunakan Metode Hamming Code Pada Pengiriman Data Text. *Journal Article*. 1(1)
- Munandar, A. 2011 Analisa Psnr, Rasio Kompresi Warna Dan Mse Terhadap Kompresi Image Menggunakan 31 Fungsi Wavelet. *Digital Information & System Conference*. 20 Mei
- Natick, MA, 2015. Matlab Creating Graphical User Interface. The Mathwork, Inc
- Setyawan, F. 2014. Simulasi Kode Hamming, Kode Bch, Dan Kode ReedSolomon untuk Optimalisasi Forward Error Correction. *Skripsi*. Surakarta:Universitas Muhammadiyah Surakarta.
- Sungkawa, I. 2011. Penerapan Ukuran Ketepatan Nilai Ramalan Data Deret Waktu Dalam Seleksi Model Permalan Volume Penjualan. *Jurnal Binus*. 2(2) : 636-645
- Sutarto, M & Suwadi, 2014. Implementasi Encoder dan Decoder BCH Menggunakan DSK TMS320C6416T. *Jurnal Teknik Pomits*. 3(1) : 29-34
- Tyas, A, L. 2011. *Watermarking Citra Digital Berbasis Dwt-Svd Dengan Detektor Non-Blind*. Skripsi. Semarang:Universitas Diponegoro.