



**SKEMA DIGITAL WATERMARKING CITRA MENGGUNAKAN  
METODE DISCRETE WAVELET TRANSFORM DENGAN PENERAPAN  
KUANTISASI ADAPTIF DAN CHINESE REMAINDER THEOREM**

**TESIS**

**PROGRAM MAGISTER TEKNIK ELEKTRO  
MINAT SISTEM KOMUNIKASI DAN INFORMATIKA**

Ditujukan untuk memenuhi persyaratan  
memperoleh gelar Magister Teknik



**NANDA MARTYAN ANGGADIMAS  
NIM. 146060300111016**

**UNIVERSITAS BRAWIJAYA  
FAKULTAS TEKNIK  
MALANG**

**2018**



**DAFTAR ISI**

**KATA PENGANTAR**..... Error! Bookmark not defined.

**DAFTAR ISI**.....i

**DAFTAR GAMBAR**..... Error! Bookmark not defined.

**DAFTAR TABEL** ..... Error! Bookmark not defined.

**BAB I PENDAHULUAN**..... Error! Bookmark not defined.

    1.1 Latar Belakang ..... **Error! Bookmark not defined.**

    1.2 Rumusan Masalah ..... **Error! Bookmark not defined.**

    1.3 Tujuan Penelitian..... **Error! Bookmark not defined.**

    1.4 Manfaat Penelitian..... **Error! Bookmark not defined.**

**BAB II TINJAUAN PUSTAKA** ..... Error! Bookmark not defined.

    2.1 Penelitian Terkait ..... **Error! Bookmark not defined.**

    2.2 Dasar Teori ..... **Error! Bookmark not defined.**

        2.2.1 Digital Watermarking..... **Error! Bookmark not defined.**

            2.2.1.1 Visible Watermark..... **Error! Bookmark not defined.**

            2.2.1.2 Invisible Watermark..... **Error! Bookmark not defined.**

        2.2.2 Kompresi Citra ..... **Error! Bookmark not defined.**

            2.2.2.1 Discrete Wavelet Transform ..... **Error! Bookmark not defined.**

            2.2.2.2 Kuantisasi..... **Error! Bookmark not defined.**

                2.2.2.2.1 Kuantisasi Adaptif Pada DWT..... **Error! Bookmark not defined.**

        2.2.3 Chinese Remainder Theorem ..... **Error! Bookmark not defined.**

        2.2.4 Pengukuran Tingkat Distorsi ..... **Error! Bookmark not defined.**

**BAB III KERANGKA KONSEP PENELITIAN**..... Error! Bookmark not defined.

    3.1 Analisa Permasalahan ..... **Error! Bookmark not defined.**

    3.2 Identifikasi Variabel..... **Error! Bookmark not defined.**

    3.3 Metode yang Relevan..... **Error! Bookmark not defined.**

        3.3.1 Discrete Wavelet Transform (DWT) Haar ..... **Error! Bookmark not defined.**

        3.3.2 Chinese Remainder Theorem (CRT)..... **Error! Bookmark not defined.**

    3.4 Konsep Pengembangan Ilmu..... **Error! Bookmark not defined.**

        3.4.1 Proses Generate dan Embedding Watermark ..... **Error! Bookmark not defined.**

        3.4.2 Proses Ekstraksi, Deteksi dan Pemulihan..... **Error! Bookmark not defined.**

    3.5 Konsep Algoritma ..... **Error! Bookmark not defined.**

        3.5.1 Discrete Wavelet Transform (DWT) Haar ..... **Error! Bookmark not defined.**



3.5.2 Chinese Remainder Theorem (CRT).....	Error! Bookmark not defined.
3.6 Hipotesis.....	Error! Bookmark not defined.
3.7 Alat Penelitian.....	Error! Bookmark not defined.
<b>BAB IV METODE PENELITIAN</b> .....	Error! Bookmark not defined.
4.1 Metode Penelitian.....	Error! Bookmark not defined.
4.2 Skema Proses <i>Generating Watermark</i> .....	Error! Bookmark not defined.
4.2.1 Proses Discrete Wavelet Transform.....	Error! Bookmark not defined.
4.2.2 Proses Kuantisasi Adaptif.....	Error! Bookmark not defined.
4.3 Skema Proses <i>Embedding Watermark</i> .....	Error! Bookmark not defined.
4.3.1 Proses CRT <i>Embedding</i> .....	Error! Bookmark not defined.
4.4 Skema Proses Ekstraksi <i>Watermark</i> .....	Error! Bookmark not defined.
4.4.1 Proses Ekstraksi CRT.....	Error! Bookmark not defined.
4.5 Skema Proses Rekonstruksi.....	Error! Bookmark not defined.
4.5.1 Proses Dekuantisasi.....	Error! Bookmark not defined.
4.6 Proses Rekonstruksi Citra Pemandangan.....	Error! Bookmark not defined.
4.7 Proses Deteksi dan Pemulihan Area Penyuntingan.....	Error! Bookmark not defined.
4.8 Data Penelitian.....	Error! Bookmark not defined.
<b>BAB V HASIL DAN PEMBAHASAN</b> .....	Error! Bookmark not defined.
5.1 <i>Generate Watermark</i> .....	Error! Bookmark not defined.
5.2 <i>Embedding Watermark</i> .....	Error! Bookmark not defined.
5.3 Ekstraksi <i>Watermark</i> .....	Error! Bookmark not defined.
5.4 Deteksi dan Pemulihan Citra.....	Error! Bookmark not defined.
<b>BAB VI KESIMPULAN DAN SARAN</b> .....	Error! Bookmark not defined.
6.1 Kesimpulan.....	Error! Bookmark not defined.
6.2 Saran.....	Error! Bookmark not defined.
<b>DAFTAR PUSTAKA</b> .....	Error! Bookmark not defined.



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan internet yang pesat menyebabkan pertukaran dan distribusi informasi digital semakin mudah. Disamping banyaknya manfaat yang dihasilkan, hal ini juga menimbulkan masalah baru, salah satunya adalah penyalahgunaan data digital yang memiliki hak cipta. Jenis data digital yang paling sering menjadi target adalah citra digital.

Penyalahgunaan citra digital tersebut meliputi penggunaan secara ilegal seperti menggunakan citra yang ada di *photostock* tanpa seijin pemilik, mengubah citra digital untuk merugikan/mempermalukan pihak lain seperti yang saat ini marak di media sosial, penyalahgunaan citra dokumen seperti pemalsuan sertifikat, serta kasus lainnya yang dapat merugikan pemilik hak cipta. Kondisi ini membuat banyak peneliti menciptakan metode-metode baru untuk mencegah permasalahan tersebut, dan metode *watermarking* menjadi salah satu solusi. Digital *watermarking* sendiri merupakan teknik untuk menyembunyikan informasi pada data digital. Sebagai contoh pada data citra digital, digital *watermarking* mampu menyisipkan informasi pada citra tanpa harus merusak citra yang disisipi, sehingga citra terproses dan citra asli tidak akan tampak perbedaannya jika dilihat dengan mata telanjang.

Berdasarkan daya tahan terhadap serangan, skema digital *watermarking* yang paling sering digunakan adalah jenis *robust watermarking* dan *fragile watermarking*. *Robust watermarking* berfokus pada pengamanan *watermark* yang disisipkan pada citra *cover*, sedangkan *fragile watermarking* lebih difokuskan pada sensitifitas terhadap serangan sehingga lebih banyak digunakan untuk metode otentikasi. Tantangan dalam penelitian digital *watermarking* untuk otentikasi citra adalah bagaimana membuat skema yang mampu menggabungkan keunggulan kedua jenis *watermarking* diatas. Karena lebih sulit mendapatkan sensitifitas pada jenis *robust* maka yang menjadi dasar skema harus menggunakan metode berbasis *fragile watermarking*.





menjelaskan skema *digital watermarking* untuk otentikasi citra yang mampu melakukan deteksi jika terjadi perubahan pada citra, serta mampu merekonstruksi citra yang sudah dirubah menjadi citra yang mendekati citra asli. Penggunaan metode CRT untuk proses *embedding watermark* memiliki keunggulan dari segi keamanan karena setidaknya ada dua parameter yang harus diketahui untuk dapat dilakukan pengambilan informasi (bit) yang disisipkan. Penggunaan metode CRT juga mampu menghasilkan citra ber-*watermark* dengan nilai PSNR antara 41-43 dB.

Penelitian lain yang menggunakan metode CRT yakni penelitian dengan metode TLDCR-CRT (Nugraha dkk, 2017). Perbedaan dengan penelitian sebelumnya adalah metode yang digunakan dalam proses *generate watermark* serta pengaplikasian pada citra berwarna. Penggunaan metode TLDCR-CRT mampu memberikan hasil citra ber-*watermark* dengan nilai PSNR 45 dB dan citra ekstraksi dengan PSNR 22-28 dB. Dari kedua penelitian diatas diketahui bahwa metode yang digunakan untuk proses *embedding* sangat berpengaruh terhadap hasil citra ber-*watermark*, sedangkan metode pada proses *generate* akan mempengaruhi hasil citra ekstraksi.

Disamping kelebihannya, kedua penelitian sebelumnya juga memiliki beberapa permasalahan antara lain penggunaan blok 8x8 pada metode DCT dan TLDCR membuat berkurangnya fleksibilitas terhadap ukuran citra yang diproses, selain itu juga dapat mempengaruhi hasil citra ekstraksi dan proses pemulihan. Permasalahan berikutnya adalah penentuan *threshold* yang harus dilakukan manual satu per satu untuk mendapatkan hasil citra pemulihan yang paling maksimal. Untuk mengatasi masalah tersebut maka dibutuhkan metode *generate* yang memiliki tingkat kompresi dan hasil dekompresi yang lebih baik dan tidak terpaku pada penggunaan blok, serta skema yang mampu mengatasi permasalahan penentuan *threshold*.

Metode lain yang biasa digunakan untuk proses *generate watermark* salah satunya adalah Transformasi Wavelet Diskrit (DWT). Keunggulan dari metode ini yakni dari segi rasio kompresi citra yang dihasilkan 19% lebih tinggi serta memiliki *information loss* hampir dua kali lebih rendah jika dibandingkan dengan metode DCT, selain itu metode DWT juga tidak menggunakan *block base* (Anilkumar dkk, 2011).

Untuk lebih memaksimalkan penggunaan metode DWT, maka dapat dilakukan optimasi dengan menerapkan kuantisasi adaptif untuk koefisien DWT 2D (Chen dkk, 2013). Penelitian tersebut menyajikan skema kuantisasi adaptif untuk koefisien DWT 2D dengan melakukan pengembangan dari kuantisasi *uniform* pada JPEG2000 menjadi sebuah skema kuantisasi adaptif, hasilnya dengan rasio kompresi yang sama (CR=56%)



tetapi mampu memiliki nilai PSNR 2,79% - 23% lebih baik jika dibandingkan dengan kuantisasi *uniform* pada metode DWT.

## 1.2 Rumusan Masalah

Penelitian ini berfokus pada rumusan masalah berikut:

1. Bagaimana rancangan skema *digital watermark* yang memanfaatkan kombinasi metode DWT dan CRT
2. Bagaimana analisa hasil skema *digital watermark* yang dirancang untuk proses *embedding* dan ekstraksi
3. Bagaimana analisa hasil skema *digital watermark* yang dirancang untuk proses deteksi dan pemulihan pada citra digital berwarna
4. Bagaimana performa dari skema yang dirancang berdasarkan tingkat *imperceptibility*, *robustness* serta waktu proses dibandingkan dengan kedua penelitian sebelumnya.

## 1.3 Tujuan Penelitian

Membuat sebuah skema *Digital Watermarking* untuk citra digital berwarna menggunakan metode CRT dan DWT melalui proses kuantisasi adaptif yang dapat mendeteksi dan merekonstruksi citra jika terjadi serangan, serta melakukan analisa performa baik dari segi *imperceptibility*, *robustness* dan kecepatan pemrosesan. Evaluasi skema juga akan dilakukan dengan membandingkan hasil skema yang dirancang dengan penelitian metode CRT sebelumnya sehingga didapatkan nilai PSNR dan NC yang lebih baik.

## 1.4 Manfaat Penelitian

Manfaat yang diharapkan dalam penelitian ini adalah sebagai penambah keragaman skema proses *Digital Watermarking* dengan penggabungan metode CRT dan DWT yang melalui proses kuantisasi adaptif, sehingga dapat menjadi pembanding dengan skema *digital watermarking* lainnya.



## BAB II TINJAUAN PUSTAKA

### 2.1 Penelitian Terkait

Penelitian Patra dkk (2012) memaparkan sebuah skema *digital watermarking* dengan metode DCT sebagai proses *watermark generate* dan metode CRT sebagai *watermark embedding*. Memanfaatkan algoritma CRT yang menyatakan sebuah bilangan integer dapat diwakili oleh dua bilangan integer yang bernilai lebih kecil, logika tersebut akan diaplikasikan pada setiap nilai piksel citra dan hasil 2 bilangan perwakilan tersebut akan disimpan dan digunakan sebagai pengaman tambahan, sehingga citra *watermark* yang disisipkan akan sulit untuk dirubah. Dan dari hasil penelitian tersebut menyatakan bahwa proses *embedding* dengan metode CRT menunjukkan perubahan citra yang minimal jika dibandingkan dengan citra awal, begitu juga pada hasil rekonstruksi.

Nugraha dkk (2017) melakukan sebuah penelitian yang berdasarkan penelitian Patra dkk (2012) tentang keunggulan metode CRT yang digabungkan dengan metode TLDCCT atau transformasi DCT secara dua level sebagai proses *generate watermark*. Penelitian tersebut ditujukan untuk proses *digital watermarking* pada citra berwarna, dan dapat melakukan rekonstruksi citra tanpa memerlukan citra awal. Hasil penelitian tersebut menyatakan bahwa penggunaan metode TLDCCT memberikan tingkat kompresi yang baik sehingga perubahan citra hasil *watermark* dari citra awal dapat diminimalisir. Hal ini menjadi poin penting karena pada citra berwarna terdapat tiga channel warna, maka setiap perubahan yang signifikan pada tiap channel warna akan menghasilkan perubahan yang besar dari citra awal. Penelitian tersebut memperoleh nilai rata-rata PSNR = 45 dB, dan pada citra *recovery* nilai PSNR = 28 dB.

Katharotiya dkk (2011) melakukan penelitian yang membandingkan kompresi citra menggunakan metode DCT dan DWT. Hasilnya menunjukkan kompresi citra menggunakan DWT memiliki rasio kompresi sekitar 19% lebih tinggi serta *information loss* dua kali lebih kecil dibandingkan dengan metode DCT.

Chen dkk (2013) menjelaskan sebuah skema metode kuantisasi adaptif dengan memodifikasi metode kuantisasi yang diterapkan pada JPEG2000. Analisa statistik yang dilakukan pada kuantisasi adaptif penelitian tersebut memanfaatkan nilai median dari

distribusi nilai koefisien DWT. Hasil penerapan kuantisasi adaptif tersebut, dengan perbandingan rasio kompresi yang sama (CR=56%), didapatkan nilai rata-rata PSNR=34dB atau berkisar 2,79% - 23% lebih baik dibandingkan dengan kuantisasi *uniform* (Chen dkk, 2013).

Beberapa penelitian lain tentang skema *digital watermarking* disebutkan pada tabel 2.1 berikut:

No	Peneliti	Obyek	Metode	Hasil
1	Nguyen dkk.	Citra <i>greyscale</i>	Proses <i>embedding</i> dilakukan pada domain wavelet atau langsung diterapkan pada koefisien DWT untuk mendeteksi <i>editing</i> .	Hasilnya nilai rata-rata PSNR=83 dB, namun proses otentikasi masih membutuhkan citra awal.
2	Vaidya dkk	Citra <i>greyscale</i>	Proses <i>embedding</i> melibatkan metode <i>Bhattacharyya distance</i> dan Kurtosis pada citra <i>greyscale</i> .	Hasilnya, tahan terhadap serangan yang diberikan pada citra hasil <i>watermark</i> , dengan nilai rata-rata PSNR=33,53 dB untuk citra hasil <i>watermarking</i> .
3	Patra dkk	Citra <i>greyscale</i>	Membandingkan antara proses <i>embedding</i> dengan menggunakan metode CRT dan metode SVD	Hasilnya, metode CRT miliki nilai PSNR=60 dB atau 33% sampai dengan 50% lebih baik dibandingkan menggunakan skema SVD, metode CRT juga lebih tahan terhadap serangan.
4	Haribabu dkk	Citra berwarna	Skema <i>digital watermarking</i> dengan merubah channel warna citra dari RGB menjadi HIS dan proses <i>embedding</i> dilakukan pada domain wavelet.	nilai rata-rata PSNR=59dB, serta tahan terhadap beberapa jenis serangan, namun proses ekstraksi citra ter- <i>watermark</i> masih membutuhkan citra awal sebagai pembanding.

Tabel 2.1 Penelitian terkait

## 2.2 Dasar Teori

### 2.2.1 Digital Watermarking

*Watermark* (tanda air) sudah digunakan mulai abad 13 di Italia yang kemudian menyebar dengan cepat ke seluruh Eropa. Tanda air tersebut digunakan sebagai penanda perusahaan pembuat kertas untuk mencegah pemalsuan. Seiring perkembangan dari teknik tersebut menginspirasi pembuatan *Digital watermarking*.

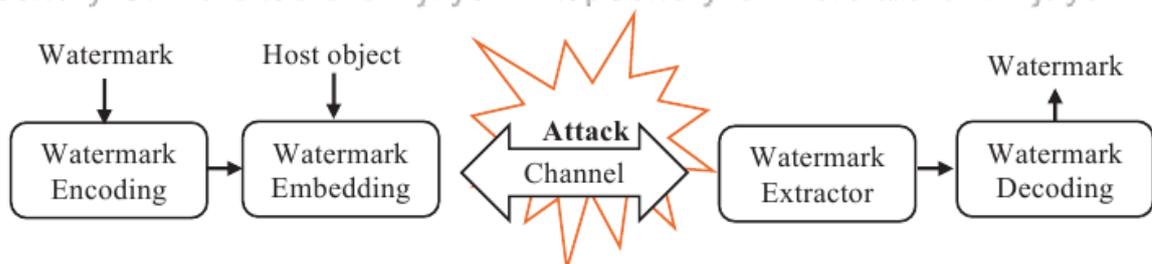
*Digital watermarking* sendiri merupakan suatu teknik untuk menyembunyikan pesan pada sinyal digital, baik citra, audio, maupun video (Cox dkk, 2007). Secara konsep *digital watermarking* sama dengan steganografi, perbedaannya hanya pada tujuan penggunaan, jika steganografi digunakan untuk menyembunyikan pesan yang tidak memiliki arti apapun terhadap konten digital yang ditumpangi, maka pada *digital watermarking* pesan yang disembunyikan memiliki arti terhadap konten digital (*host*) yang ditumpangi, misal logo pembuat pada sebuah foto. Secara garis besar *digital watermarking* dibagi menjadi dua, yakni *visible watermark* dan *invisible watermark*.

### 2.2.1.1 Visible Watermark

*Visible watermark* merupakan proses penambahan pesan berupa logo atau gambar pada konten digital baik citra maupun video digital yang dapat dilihat oleh mata telanjang. Tanda yang disisipkan akan disesuaikan tingkat kecerahannya sehingga pesan dapat terlihat. Metode ini biasanya digunakan sebagai penanda hak cipta pemilik atau pembuat konten digital. Hanya saja untuk beberapa kasus, tanda air ini bisa mengganggu pengguna konten tersebut, karena dapat mengurangi nilai estetika pada konten digital.

### 2.2.1.2 Invisible Watermark

Metode *invisible watermark* memiliki tujuan yang sama dengan *visible watermark* yakni mempertahankan tanda atau pesan yang disisipkan pada konten digital (objek *host*) sehingga bisa digunakan untuk mencegah klaim kepemilikan konten oleh pihak lain maupun mendeteksi apabila ada perubahan pada konten. Berbeda dengan *visible watermark*, pada metode *invisible watermark*, tanda atau pesan yang disisipkan tidak dapat dilihat dengan mata manusia, hal ini dikarenakan pesan yang disisipkan akan ditransformasi pada domain frekuensi kemudian nilai bit yang dihasilkan akan diacak dan langsung ditanam pada konten digital (objek *host*). Proses pengacakan melibatkan sebuah *key*, tujuannya agar dapat mengetahui posisi bit yang sudah diacak agar bisa diambil untuk menghasilkan kembali pesan yang disisipkan.



Gambar 2.1 Konsep digital watermarking

Sumber: Nematollahi dkk (2017:3)



Pada gambar 2.1 dijelaskan bahwa Proses *digital watermarking* terdiri dari proses *embedding watermark* untuk menyisipkan pesan pada konten digital dan proses *extracting watermark* untuk memperoleh kembali pesan yang disisipkan. Proses *embedding watermark* membutuhkan tiga komponen yakni objek induk (*host*), pesan yang disisipkan, serta penambahan *key*. Pesan yang disisipkan (*watermark*) akan melalui proses *encoding/generate watermark*, tujuannya adalah untuk mengubah data pesan menjadi bit-bit biner untuk disisipkan pada objek *host*. Dalam proses *generate watermark* terdapat proses kompresi data yang bertujuan agar data pesan yang disisipkan tidak membebani objek *host* sehingga objek hasil proses *watermarking* tidak mengalami banyak kerusakan.

Untuk memperoleh pesan yang disisipkan dibutuhkan proses deteksi atau *extracting watermark*. Proses ini membutuhkan tiga komponen yakni objek hasil proses *watermark*, objek *watermark* (pesan) atau bisa juga objek induk (*host*) serta *key* yang dipakai dalam proses *embedding*. Berdasarkan komponen yang dibutuhkan dalam proses *recovery watermark* diatas, maka *digital watermarking* dapat dibedakan menjadi tiga, yaitu:

a) *Non-Blind Watermarking (Private Watermarking)*

Pada model ini dibutuhkan objek *host* (data asli) atau objek *watermark* (data pesan) untuk membandingkan atau menunjukkan apakah dalam objek yang diproses terdapat *watermark* atau tidak.

b) *Semi-Blind Watermarking (Semi Private Watermarking)*

Pada jenis skema ini tidak dibutuhkan data asli, akan tetapi hasil keluaran hanya akan menjelaskan apakah didalam objek yang diproses mengandung *watermark* atau tidak.

c) *Blind Watermarking (Public Watermarking)*

Pada jenis skema ini tidak membutuhkan data original baik objek *host* ataupun objek *watermark*. Model ini merupakan model yang hingga saat ini masih menjadi tantangan dalam penelitian *digital watermarking*.

Jenis serangan yang terjadi (*Attack*) yang disebut pada gambar 2.1 adalah manipulasi yang dilakukan pada objek hasil proses *watermarking*. Contoh manipulasi yang dimaksud antara lain proses *editing* baik *cropping*, *rotating* atau merubah sebagian dengan objek lain (untuk citra dan video digital), pemberian *noise*, maupun kompresi data pada objek hasil *watermarking*. Berdasarkan tingkat ketahanan (*robustness*) *watermark* yang disisipkan terhadap serangan yang terjadi, terdapat tiga jenis model *digital watermarking* yang ketiganya memiliki tujuan dan aplikasi penggunaan yang berbeda-beda (Nematollahi dkk, 2017). Ketiga jenis model tersebut antara lain:



a) *Robust Digital Watermarking*

Model ini difokuskan agar *watermark* yang disisipkan dalam data *host* memiliki daya tahan yang kuat terhadap berbagai bentuk manipulasi (*attack*). Contoh penggunaan model ini adalah untuk proses perlindungan hak cipta terhadap data *host*.

b) *Semi-Fragile Digital Watermarking*

Model ini menggabungkan kelebihan model *robust digital watermarking* dari segi ketahanan, akan tetapi terbatas pada jenis serangan yang dilegitimasi (*friendly attack*) yang masih masuk dalam toleransi. Serta kelebihan model *fragile digital watermarking* yang memiliki sensitifitas terhadap serangan yang tidak bisa dilegitimasi (*malicious attack*).

c) *Fragile Digital Watermarking*

Pada skema ini akan menitik beratkan pada kemampuan sensitifitas yang baik terhadap segala perubahan yang terjadi pada *watermark* yang disisipkan. Dalam *digital watermarking*, untuk mendapatkan tingkat *fragility* akan lebih susah daripada mendapatkan tingkat *robustness*. Contoh penggunaan model ini adalah sebagai sistem otentikasi data digital yang tidak memperbolehkan adanya pengubahan sedikit pun terhadap data sudah melalui proses *watermarking*.

### 2.2.2 Kompresi Citra

Salah satu tantangan pengaplikasian *digital watermarking* pada citra digital adalah bagaimana menyisipkan *watermark* agar tidak membebani citra induk (*host*) yang ditumpanginya sehingga mendapatkan hasil proses *watermarking* yang tidak berbeda dari citra induk semula, atau bisa disebut juga memenuhi konsep *imperceptibility*. Untuk itulah pada proses *generate watermark* dibutuhkan proses kompresi citra. Citra *watermark* akan dikompres terlebih dahulu baru kemudian disisipkan pada citra induk.

Kompresi citra sendiri masuk dalam bidang kompresi data yang merupakan ilmu atau teknik untuk membuat informasi menjadi bentuk yang lebih padat dan ringkas (Sayood, 2006). Secara umum proses kompresi citra dilakukan dengan cara mengidentifikasi dan mengurangi duplikasi data didalam citra, sehingga representasi informasi citra akan lebih sedikit dibandingkan representasi citra awal. Menurut Sutoyo dkk(2009:166) terdapat dua jenis teknik dalam melakukan kompresi citra, yakni:



## 1. *Lossless Compression*

Kompresi *lossless* atau *lossless compression* merupakan salah satu jenis kompresi citra yang menghasilkan citra dekompresi memiliki kualitas yang hampir sama jika dibandingkan dengan citra aslinya atau bisa dibilang hanya sedikit informasi yang hilang dari citra asli. Akan tetapi rasio kompresi citra metode ini sangat rendah. Banyak aplikasi yang memerlukan kompresi tanpa cacat, seperti pada aplikasi radiografi, kompresi citra hasil diagnosa medis atau citra satelit yang apabila kehilangan informasi citra sekecil apapun akan menyebabkan hasil yang tidak diharapkan.

## 2. *Lossy Compression*

Model *lossy compression* adalah model kompresi citra yang menghasilkan citra dekompresi yang tidak sama dengan citra aslinya karena ada informasi yang hilang, akan tetapi masih dapat ditolerir oleh persepsi mata manusia. Metode ini menghasilkan rasio kompresi yang lebih tinggi daripada metode *lossless*. Contoh metode pada *lossy compression* adalah *transform coding* seperti transformasi Fourier, Wavelet dan lainnya.

Dalam penelitian ini akan menggunakan metode *lossy compression* agar mendapatkan nilai rasio kompresi yang lebih tinggi. Proses yang digunakan adalah proses *transform coding*, dimana didalamnya terdapat proses transformasi wavelet diskrit dan proses kuantisasi.

### 2.2.2.1 Discrete Wavelet Transform

Transformasi wavelet diskret (*discrete wavelet transform/DWT*) merupakan salah satu metode yang sering dipakai dan sangat baik digunakan untuk representasi analisis sinyal diskret. Kelebihan dari DWT adalah pada saat yang bersamaan dapat memberikan informasi frekuensi dan informasi temporal, berbeda dengan DFT dan DCT yang hanya memberikan informasi frekuensi. Sehingga DWT lebih sering digunakan untuk analisis *time-frequency* dari sebuah sinyal (Madenda, 2015:235).

DWT pertama kali diperkenalkan oleh seorang ahli matematika Hungaria bernama Alfred Haar. Untuk sederetan data sinyal masukan yang direpresentasikan dalam bilangan numerik  $2^n$ , transformasi wavelet Haar menyatakan bahwa setiap pasang bilangan data masukan itu dapat dinyatakan dalam satu nilai selisih dan satu nilai jumlah antara keduanya. Bagian nilai jumlah dinyatakan sebagai *koefisien aproksimasi* yang berisikan informasi global atau komponen frekuensi rendah yang terdapat dalam sebuah citra,

sedangkan bagian nilai selisih dikenal sebagai *koefisien wavelet* yang berisikan informasi detail atau informasi frekuensi tinggi. Koefisien frekuensi tinggi inilah yang akan dikompresi sedemikian rupa sehingga dapat diperoleh rasio kompresi yang tinggi dan kualitas citra yang masih baik.

Ide dasar wavelet Haar tersebut, secara sederhana dapat dituangkan dalam formulasi matematika 1-D berikut. Andaikan sederatan  $N$  data  $X = (x_1, x_2, \dots, x_n)$  dengan  $N=2^i$  dan  $i > 1, i \in \mathbb{Z}$ . Deretan nilai jumlah (bersimbol  $L$ ) dan deretan nilai selisih (bersimbol  $H$ ) dari data  $X$  dapat dituliskan sebagai berikut:

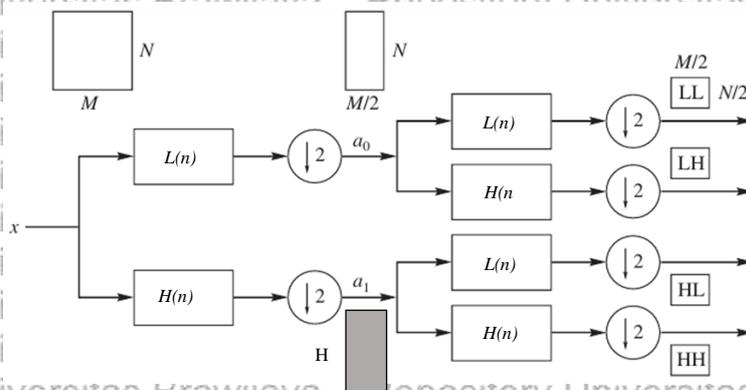
$$L(n) = \frac{x(2n-1)+x(2n)}{2}, \quad n = 1, \dots, N/2 \quad (2.1)$$

$$H(n) = \frac{x(2n-1)-x(2n)}{2}, \quad n = 1, \dots, N/2 \quad (2.2)$$

Dengan demikian, transformasi wavelet Haar terhadap  $X$  dapat dituliskan sebagai berikut:

$$X = (x_1, x_2, \dots, x_n) \rightarrow (L|H) = (l_1, l_2, \dots, l_{N/2} | h_1, h_2, \dots, h_{N/2}) \quad (2.3)$$

Untuk proses transformasi wavelet 2-D, struktur proses transformasinya dapat dilihat di Gambar 2.2. Pada DWT Haar, proses transformasi dimulai dari perhitungan secara bersamaan terhadap komponen citra asli, yakni komponen frekuensi rendah dengan fungsi *low-pass filter*  $L(n)$  dan komponen frekuensi tinggi dengan fungsi *high-pass filter*  $H(n)$ . Proses ini dilakukan pada arah horizontal antara dua piksel atau lebih, dimulai pada baris pertama dari kolom pertama hingga kolom terakhir, kemudian dilanjutkan ke baris berikutnya hingga baris terakhir. Simbol operator *downsampling*  $\downarrow 2$  bermakna bahwa untuk setiap dua kolom citra asli akan tinggal menghasilkan satu kolom setelah proses. Dengan demikian keluaran dari filter  $L(n)$  adalah citra  $L_1$  yang berisikan informasi global yang mirip dengan citra aslinya namun jumlah ukuran kolomnya menjadi setengahnya, sedangkan keluaran dari filter  $H(n)$  adalah citra  $H_1$  yang mengandung informasi detail dengan jumlah kolomnya sama dengan jumlah kolom citra keluaran filter  $L(n)$ .

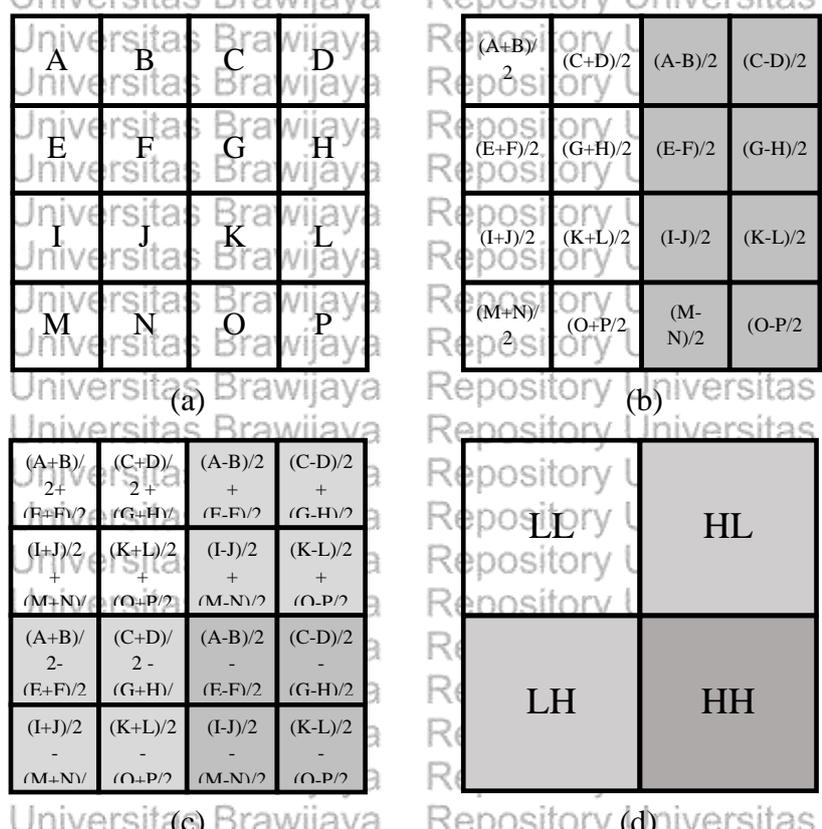


Gambar 2.2 Struktur proses transformasi wavelet pada citra 2-D

Sumber: Sayood (2012:532)

Tahap selanjutnya adalah citra  $L_1$  dan  $H_1$  masing-masing diproses kembali dengan menggunakan filter  $L(n)$  dan  $H(n)$  yang sama. Proses ini dilakukan pada arah vertikal antara dua piksel atau lebih, mulai dari baris pertama hingga baris terakhir pada kolom pertama dan selanjutnya kolom berikutnya hingga kolom terakhir. Secara sederhana proses perhitungannya dapat dilihat pada contoh di Gambar 2.3.

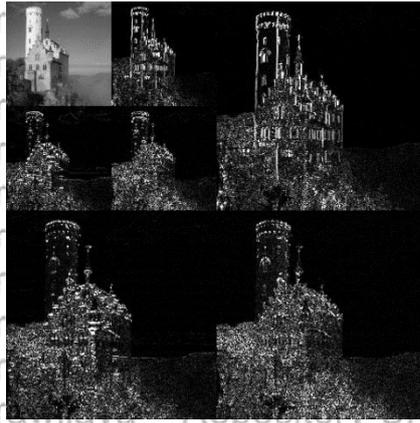
Dalam DWT, Gambar 2.2 dan Gambar 2.3 merupakan contoh dari transformasi wavelet skala 1. Hasil dari proses DWT skala 1 berupa nilai koefisien subband LL, HL, LH dan HH. Dalam beberapa aplikasi seringkali digunakan DWT multi skala, yaitu dengan melakukan proses transformasi ulang pada subband tertentu atau yang paling sering pada subband LL. Tujuan dari proses DWT multi skala ini adalah untuk mendapatkan nilai koefisien transformasi yang lebih kompak, sehingga berdampak pada rasio kompresi yang lebih tinggi. Akan tetapi karena banyak informasi detail (frekuensi tinggi) yang tidak digunakan, penggunaan DWT multi skala juga dapat mengakibatkan citra mengalami lebih banyak kerusakan. Contoh peengaplikasian DWT multi skala dapat dilihat pada Gambar 2.4.



Gambar 2.3 Proses DWT Haar  
 (a) Data asli; (b) Proses horizontal; (c) Proses vertikal; (d) 4 Subband hasil DWT

Sumber: Chen dkk (2013:3)

Dalam gambar 2.4 menunjukkan proses DWT skala 2 pada sebuah citra yang menghasilkan 7 subband.



Gambar 2.4 Hasil proses DWT skala 2

Sumber: Nematollahi dkk (2017)

Salah satu kelebihan metode DWT dibandingkan metode lainnya adalah jumlah variasi skema yang dapat digunakan. Pemilihan subband yang akan digunakan akan menghasilkan rasio kompresi dan kualitas citra yang berbeda. Kelebihan lainnya adalah pada proses kuantisasi, jumlah rentang bit yang digunakan dapat diatur untuk masing-masing subband yang dipilih.

### 2.2.2.2 Kuantisasi

Dalam penggunaan kompresi *lossy*, untuk mengurangi representasi seluruh data akan diwakilkan menjadi sekelompok kecil representasi data baru (*codeword*). Hal ini bertujuan untuk menghilangkan informasi yang kurang signifikan. Proses representasi menjadi sekelompok kecil data inilah yang disebut sebagai proses kuantisasi (Sayood, 2012:252).

Skema proses kuantisasi memegang peran penting dalam menghasilkan rasio kompresi dan memperkirakan banyaknya informasi yang hilang. Secara umum proses kuantisasi dapat dikelompokkan menjadi tiga, yaitu:

#### a) Kuantisasi *Uniform*

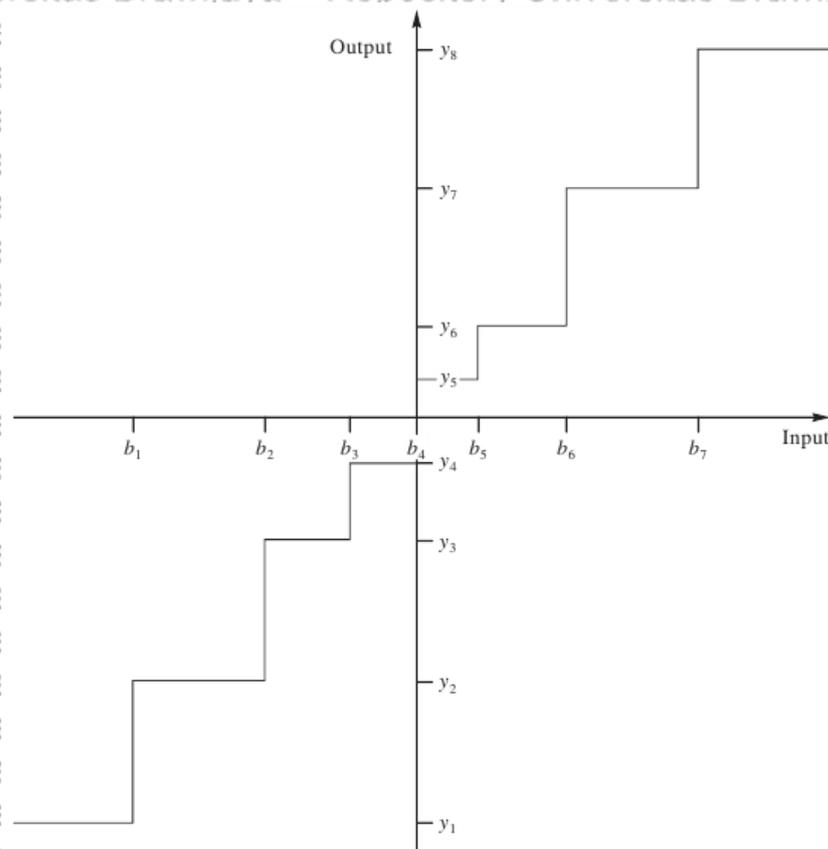
Jenis kuantisasi ini merupakan salah satu skema yang paling sering digunakan dalam proses kompresi citra. Pada kuantisasi *uniform*, data input akan dibagi menjadi interval/*step size* ( $\Delta$ ) yang sama.



Pada Gambar 2.6 (b), apabila menggunakan interval 1 bit per piksel artinya nilai intensitas pada *range* [0,255] akan dibagi menjadi 2 interval yakni [0,127] dan [128,255], oleh karena itu citra asli akan diwakili oleh 2 nilai derajat keabuan.

b) Kuantisasi *Non-Uniform*

Kuantisasi *uniform* memiliki kelemahan apabila data input memiliki distribusi nilai dengan ciri tertentu. Dengan kondisi tersebut pemakaian kuantisasi *uniform* akan mengakibatkan semakin besar nilai *error* yang terjadi, untuk itulah perlu digunakan kuantisasi *Non-uniform*. Salah satu cara yang digunakan dalam kuantisasi *non-uniform* adalah dengan memperhatikan nilai *pdf* (*Probability Density Function*), setelah itu bisa dianalisa distribusi nilai data input dimana probabilitas yang tinggi akan diberikan interval yang lebih sempit, sedangkan pada nilai dengan probabilitas rendah diberikan interval yang lebih lebar. Hal ini bisa dilihat pada Gambar 2.7 tentang hubungan input dan output pada kuantisasi *non-uniform*.

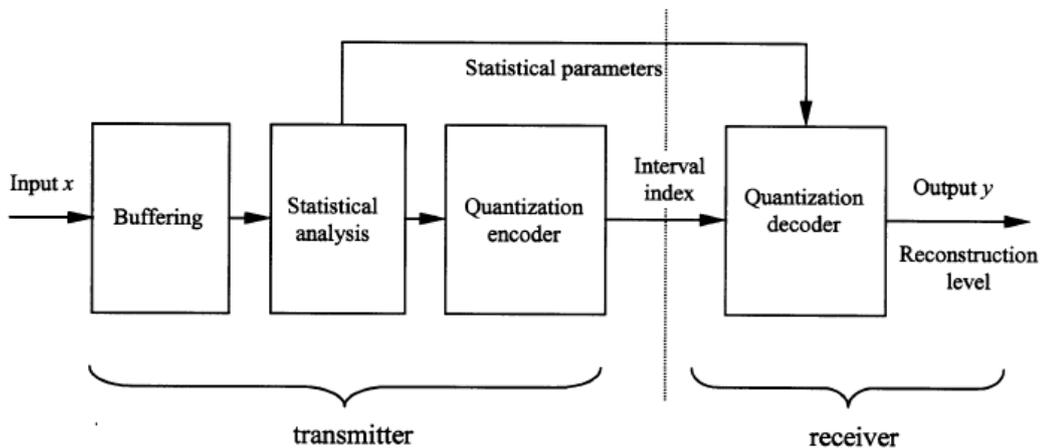


Gambar 2.0.7 Hubungan input dan ouput pada kuantisasi Non-Uniform  
*Sumber: Sayood (2012:277)*

c) Kuantisasi Adaptif

Kuantisasi *non-uniform* juga masih memiliki kelemahan, dan salah satu kelemahan pada kuantisasi tersebut adalah distribusi interval yang digunakan bersifat statis atau bisa dikatakan hanya untuk distribusi dengan ciri nilai tertentu. Apabila terjadi perbedaan nilai *pdf* maupun nilai *variance* pada data input, maka akan terjadi *mismatch* sehingga menimbulkan nilai *error* yang besar. Oleh karena itu penggunaan kuantisasi adaptif merupakan solusi yang paling baik untuk mengatasi variasi pada data input.

Kuantisasi adaptif melibatkan perhitungan statistik untuk menganalisa distribusi data input, sehingga dapat diputuskan model *quantizer* yang paling cocok digunakan. Dari segi adaptasi terhadap data input, kuantisasi adaptif memiliki dua jenis skema yang paling umum digunakan yakni *Forward Adaptive Quantization* dan *Backward Adaptive Quantization*. Skema *Forward Adaptive Quantization* dapat dilihat pada Gambar 2.8.

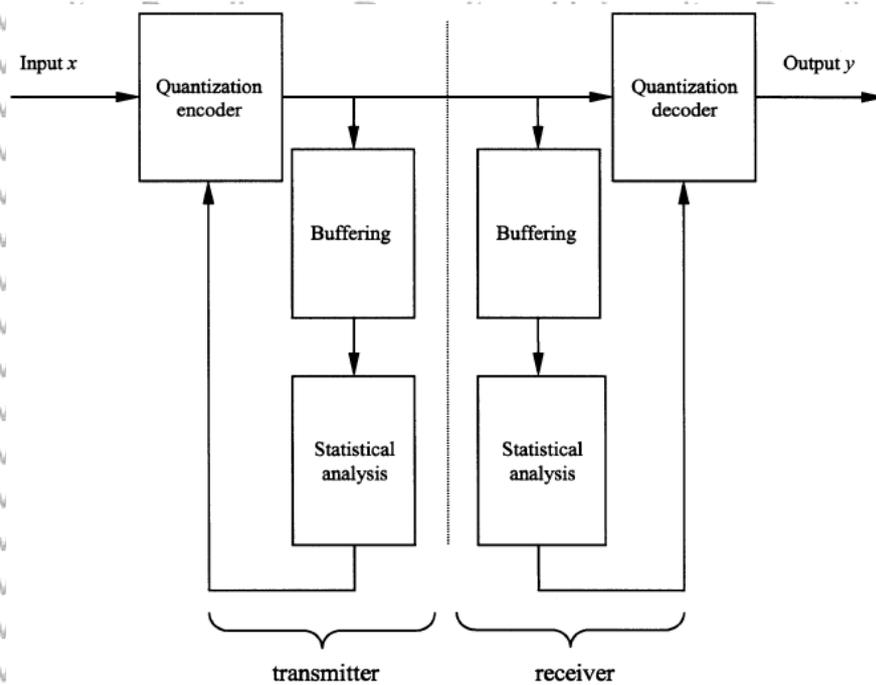


Gambar 2.8 *Forward Adaptive Quantization*

Sumber: Shi dkk (2000:67)

Salah satu poin yang perlu ditekankan pada proses *Forward Adaptive Quantization* adalah data parameter yang digunakan pada analisa statistik nilai input harus disimpan sebagai informasi tambahan untuk digunakan pada proses dekuantisasi. Pemilihan ukuran blok citra akan sangat berdampak pada besarnya informasi tambahan ini. Semakin kecil blok citra yang digunakan maka semakin besar informasi tambahan yang disimpan, akan tetapi adaptasi terhadap nilai input akan semakin baik. Sebaliknya semakin besar blok citra maka semakin sedikit informasi tambahan yang disimpan, tetapi adaptasi dan sensitifitas terhadap nilai input akan semakin berkurang.

Skema lain dari kuantisasi adaptif adalah *Backward Adaptive Quantization*. Perbedaan dengan skema sebelumnya yakni pada *Backward Adaptive Quantization* tidak membutuhkan informasi tambahan yang perlu disimpan, karena proses analisa statistik berada di kedua sisi *encoder* dan *decoder*. Namun skema ini memiliki kekurangan karena analisa statistik pada sisi *decoder* hanya memperoleh data hasil (output) kuantisasi tanpa mengetahui informasi data input original. Hal inilah yang dapat menyebabkan kemungkinan *noise* pada analisa statistik disisi *decoder*.



Gambar 2.9 *Backward Adaptive Quantization*  
 Sumber: Shi dkk (2000:68)

#### 2.2.2.2.1 Kuantisasi Adaptif Pada DWT

Penggunaan kuantisasi adaptif pada proses kompresi citra 2-D sangat berpengaruh terhadap kualitas citra hasil kompresi, hal ini dibuktikan pada penelitian sebelumnya (Chen dkk, 2013). Dalam penelitian tersebut didapatkan nilai PSNR yang lebih baik dengan tetap mempertahankan rasio kompresi yang sama atau bahkan lebih baik dibandingkan dengan penggunaan kuantisasi *uniform*. Jenis kuantisasi adaptif dalam penelitian tersebut adalah *Forward Adaptive Quantization*, artinya terdapat informasi tambahan yang harus disimpan agar bisa dilakukan proses dekuantisasi.



Proses kuantisasi yang dilakukan cukup sederhana, yaitu memodifikasi kuantisasi *uniform* pada kompresi JPEG2000 dengan menerapkan nilai *median* sebagai parameter statistik untuk menganalisa distribusi nilai input.

Misal diketahui sebuah koefisien subband 4x4 dengan distribusi nilai seperti pada Gambar 2.10. Berikut adalah langkah proses kuantisasi adaptif penelitian tersebut:

#### Step I – Hitung nilai median data input

Proses kuantisasi akan diawali dengan mencari nilai median data tersebut. Didapatkan

$$\text{median} = \frac{177+180}{2} = 179.$$

255	210	198	177
155	25	200	0
153	199	174	180
150	200	45	207

Gambar 2.10 Contoh data koefisien subband 4x4

Sumber: Chen dkk (2013:4)

#### Step II – Kurangi setiap data input dengan nilai median

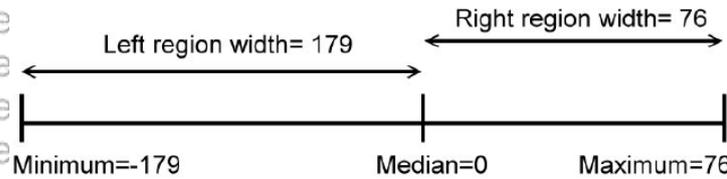
Setelah didapatkan nilai median proses berikutnya adalah mengurangi semua data input dengan nilai median. Hasil pengurangan tersebut akan menjadi data pengganti untuk langkah selanjutnya.

76	31	19	-2
-24	-154	21	-179
-26	20	-5	1
-29	21	-134	28

Gambar 2.11 Data pengganti (hasil Step II)

Sumber: Chen dkk (2013:5)

Dari proses tersebut, distribusi nilai akan membentuk dua wilayah distribusi asimetris, sisi kiri dengan bagian distribusi nilai (-179,0) dengan lebar 179 dan bagian kanan median dengan distribusi (0,76) yang memiliki lebar 76, seperti yang bisa dilihat pada Gambar 2.12.



Gambar 2.12 Dua wilayah kuantisasi asimetris

Sumber: Chen dkk (2013:5)

Penggunaan nilai median berfungsi untuk memetakan distribusi nilai input, dengan mengutamakan wilayah manakah yang memiliki rentang distribusi nilai input yang lebih kecil, wilayah sisi kanan (nilai input > median) ataukah wilayah sisi kiri (nilai input < median), dengan begitu interval kuantisasi akan dibuat lebih sempit pada wilayah distribusi nilai input yang lebih kecil tersebut.

**Step III** – Penentuan interval kuantisasi

Penentuan interval kuantisasi disesuaikan dengan kebutuhan rasio kompresi, semakin kecil interval kuantisasi artinya semakin besar rasio kompresi, akan tetapi semakin besar pula nilai error yang terjadi sehingga citra semakin banyak kerusakan saat direkonstruksi.

Sebaliknya semakin besar interval kuantisasi yang digunakan, artinya semakin kecil rasio kompresi tetapi semakin kecil nilai error. Misal dalam contoh ini digunakan interval 10, artinya kedua sisi wilayah distribusi pada Gambar 2.12 akan memiliki masing-masing 10 interval. Dengan begitu akan didapatkan nilai lebar interval kuantisasi sisi kiri ( $\Delta b_L$ ) = 17,9 dan nilai lebar interval kuantisasi sisi kanan ( $\Delta b_R$ ) = 7,6.

**Step IV** – Perhitungan nilai kuantisasi

Persamaan untuk menghitung nilai kuantisasi pada JPEG2000 (kuantisasi uniform) adalah

$$Q(i,j) = \text{sign}(H(i,j)) \left\lfloor \frac{|H(i,j) - \omega|}{\Delta b} \right\rfloor \quad (2.4)$$

Dengan  $Q(i,j)$  adalah nilai hasil kuantisasi pada posisi  $(i,j)$ ,  $H(i,j)$  merupakan koefisien DWT pada posisi  $(i,j)$ ,  $\Delta b$  adalah lebar interval kuantisasi, serta  $\omega$  adalah rata-rata dari nilai maksimum dan minimum koefisien DWT. Karena penggunaan median yang menghasilkan distribusi asimetris, maka persamaan 2.4 disesuaikan menjadi:

$$Q'(i,j) = \text{sign}(H'(i,j)) \left\lfloor \frac{|H'(i,j) - \omega|}{\Delta b_L} \right\rfloor \quad (2.5)$$

$$Q'(i,j) = \text{sign}(H'(i,j)) \left\lfloor \frac{|H'(i,j) - \omega|}{\Delta b_R} \right\rfloor \quad (2.6)$$

Dengan  $\Delta b_L$  dan  $\Delta b_R$  masing-masing mewakili interval sisi kiri dan kanan, serta  $\omega$  = median. Sehingga koefisien pada Gambar 2.11 jika diproses akan menghasilkan nilai kuantisasi seperti yang ditunjukkan dalam Gambar 2.13.

10	4	3	0
-1	-9	3	-10
-1	3	0	0
-2	3	-7	4

Gambar 2.13 Nilai hasil proses kuantisasi

Sumber: Chen dkk (2013:6)

Untuk proses dekuantisasi menggunakan persamaan berikut:

$$R_{Q'(i,j)} = \begin{cases} (Q'(i,j) + r)\Delta b + \omega, Q'(i,j) > 0 \\ (Q'(i,j) - r)\Delta b + \omega, Q'(i,j) < 0 \\ \omega, \text{lainnya} \end{cases} \quad (2.7)$$

Dengan  $R_{Q'(i,j)}$  merupakan koefisien rekonstruksi dari  $Q'(i,j)$ , dan  $r$  adalah parameter opsional.

### 2.2.3 Chinese Remainder Theorem

*Chinese Remainder Theorem* adalah metode yang memanfaatkan konsep kongruensi simultan yang merupakan sekelompok relasi kongruensi yang memiliki hubungan satu sama lain pada variabel yang sama tapi dengan moduli yang berbeda.

Dalam pemakaian terhadap sistem *digital watermarking*, metode ini menjelaskan bahwa sebuah bilangan integer dapat direpresentasikan dengan dua bilangan yang lebih kecil. Apabila terdapat dua bilangan integer  $S1$  dan  $S2$  dengan  $S=S1*S2$ , maka sebuah bilangan  $Z$  berada pada rentang antara 0 sampai  $S-1$  ( $0 \leq Z \leq S-1$ ), dengan begitu berapa pun nilai  $Z$  dapat direpresentasikan dengan sepasang integer  $(R1, R2)$ , dimana  $R1 < S1$  dan  $R2 < S2$ . Nilai  $R1$  dan  $R2$  dapat diketahui dengan melibatkan operasi modulo yang terwujud dalam relasi kongruensi  $Z \equiv R1 \pmod{S1}$  dan  $Z \equiv R2 \pmod{S2}$  (Aditya dkk, 2017).

Sebagai contoh, apabila  $S1 = 6$  dan  $S2 = 11$ , sehingga nilai  $S = S1*S2 = 66$ , maka bilangan berapapun di antara 0 dan 65 dapat direpresentasikan oleh nilai  $(R1, R2)$ , dengan nilai  $R1$  berada pada rentang  $0 \leq R1 \leq 5$ , sedangkan nilai  $R2$  berada pada rentang  $0 \leq R2 \leq 10$ . Apabila nilai integer yang dimaksud  $Z = 52$ , maka diperoleh relasi kongruensi  $52 \equiv R1 \pmod{6}$  dan  $52 \equiv R2 \pmod{11}$ , sehingga  $R1 = 4$ ,  $R2 = 8$ . Maka, nilai  $Z = 52$  dapat diwakili dengan  $(4,8)$ .



### 2.2.4 Pengukuran Tingkat Distorsi

Untuk mengukur kualitas *digital watermarking* salah satunya adalah mengukur tingkat *imperceptibility* yaitu dengan membandingkan antara citra awal/asli dan citra hasil proses *watermarking*. Metriks pengukuran tingkat distorsi yang sering digunakan adalah MSE (*Mean Square Error*) dan PSNR (*Peak Signal to Noise Ratio*). Metriks ini mengukur setiap perubahan yang terjadi pada citra. Berdasarkan (Katzenbeisser dkk, 2000:112) dan (Patra dkk, 2010) persamaan untuk mendapatkan nilai MSE dan PSNR adalah:

$$MSE = \frac{1}{M \times N} \sum_{x=1}^M \sum_{y=1}^N [P_{x,y} - P'_{x,y}]^2 \quad (2.8)$$

$$PSNR = 10 \log_{10} \frac{(\max P_{x,y})^2}{MSE} \quad (2.9)$$

Dengan  $P_{x,y}$  adalah nilai intensitas piksel citra pada posisi  $x,y$ ,  $P'_{x,y}$  merupakan nilai intensitas piksel citra setelah proses *watermarking* pada posisi  $x,y$ , serta  $M$  dan  $N$  adalah ukuran panjang dan lebar dari kedua citra yang dibandingkan.

Sedangkan metriks untuk mengukur tingkat *robustness* menggunakan parameter *Normalized Correlation* (NC) (Su dkk, 2013). Persamaan untuk mencari nilai NC ditunjukkan pada persamaan (2.10).

$$NC = \frac{\sum_{x=1}^M \sum_{y=1}^N P_{x,y} - P'_{x,y}}{\sqrt{\sum_{x=1}^M \sum_{y=1}^N [P_{x,y}]^2} \sqrt{\sum_{x=1}^M \sum_{y=1}^N [P'_{x,y}]^2}} \quad (2.10)$$





## BAB III KERANGKA KONSEP PENELITIAN

### 3.1 Analisa Permasalahan

Secara umum parameter yang dapat mempengaruhi kualitas digital *watermarking* pada sebuah citra adalah metode yang dipakai dalam proses *generate* dan *embedding watermark*. Pada proses *generate watermark* bertujuan untuk mendapatkan bit *watermark*, metode kompresi citra yang menjadi salah satu proses di dalam proses *generate* sangat berperan besar untuk mendapatkan hasil ekstraksi digital *watermarking* dengan kualitas citra yang baik, terlebih untuk pengaplikasian pada citra berwarna yang memiliki tiga *channel* warna. Oleh karena itu skema pemilihan subband yang digunakan dalam transformasi DWT akan berdampak pada citra ekstraksi.

Pada proses *embedding watermark*, metode CRT memiliki keunggulan dalam segi keamanan serta mampu menghasilkan penggabungan bit *watermark* dengan hanya sedikit perubahan yang terjadi pada citra induk. Pada proses ini hal yang perlu diperhatikan adalah jumlah bit yang akan disisipkan, karena semakin besar informasi atau bit *watermark* yang disisipkan pada citra induk maka semakin berkurang kualitas citra hasil *watermarking*, akan tetapi citra ekstraksi akan semakin baik.

### 3.2 Identifikasi Variabel

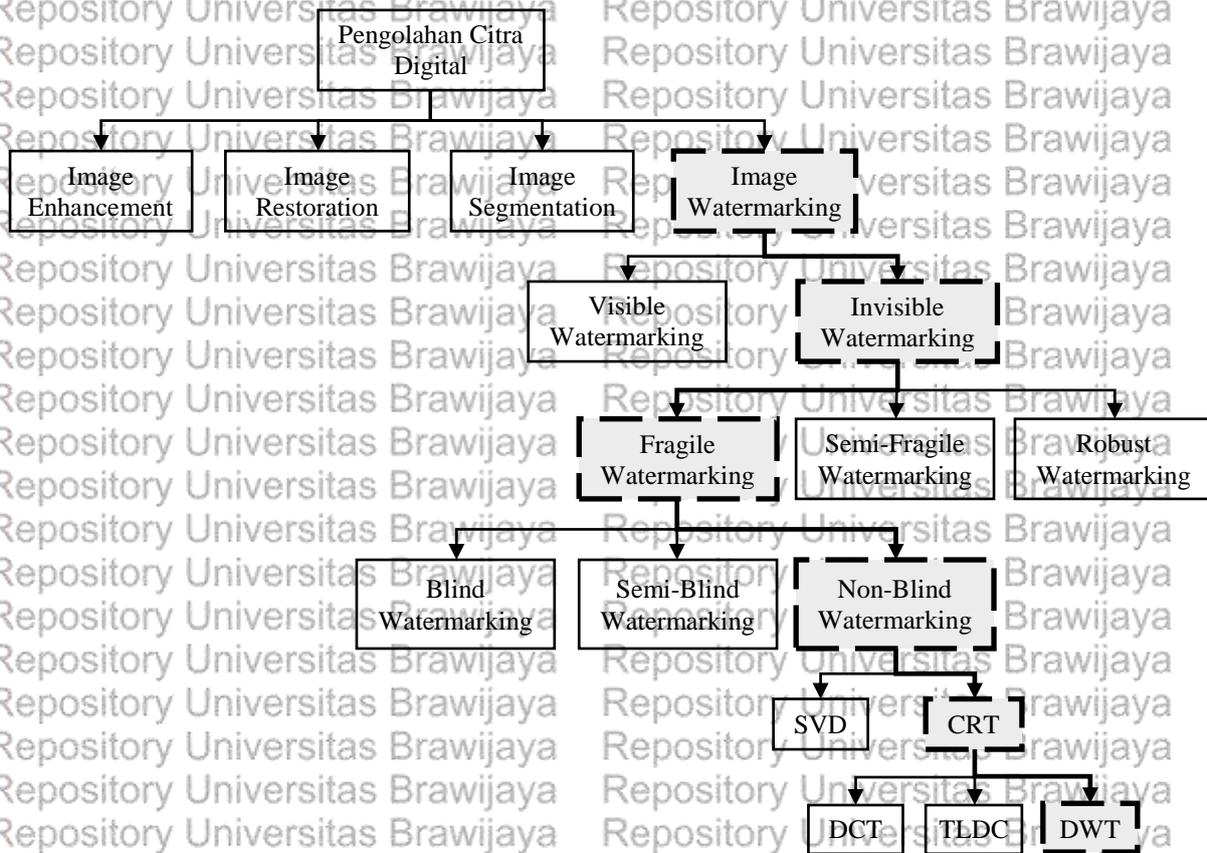
Berdasarkan permasalahan yang sudah dijelaskan maka variabel yang terlibat dalam penelitian ini adalah sebagai berikut:

1. Subband yang digunakan, bernilai LL, LH, HL atau HH.
2. Nilai maksimal subband: bernilai angka.
3. Nilai minimal subband: bernilai angka.
4. Nilai median subband: bernilai angka.
5. Interval kuantiasi: bernilai angka.
6. Nilai  $S_1$  dan  $S_2$  pada proses CRT: bernilai angka.
7. Nilai PSNR: bernilai angka.
8. Nilai NC: bernilai angka.

Variabel yang disebut diatas merupakan variabel yang digunakan dalam proses DWT dan proses CRT. Sedangkan untuk poin 7 dan 8 merupakan parameter yang digunakan untuk menguji kualitas hasil penelitian.

### 3.3 Metode yang Relevan

Dari penjabaran masalah dan variabel yang digunakan, maka berdasarkan tinjauan pustaka di bab sebelumnya, selanjutnya akan dibahas metode yang dapat digunakan dalam upaya untuk pemecahan masalah tersebut.



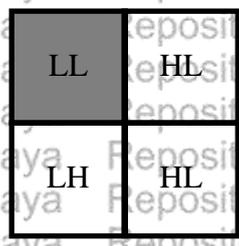
Gambar 3.1 Hirarki penelitian dalam metode digital watermarking

#### 3.3.1 Discrete Wavelet Transform (DWT) Haar

Pada proses *generate watermark*, metode yang digunakan adalah metode DWT Haar. Metode ini dipilih karena tidak menggunakan *block based* serta memiliki hasil kompresi citra yang lebih baik dari metode TLDC dan DCT, hal ini sesuai dengan yang sudah dijelaskan dalam Bab 1 dan 2.

Pemilihan subband yang digunakan dalam proses DWT sangat menentukan hasil penelitian. Oleh karena itu, untuk memberikan tingkat kompresi yang baik dengan tetap menjaga kualitas citra hasil kompresi maka subband yang dipilih dalam penelitian ini

adalah subband LL. Subband HL, LH dan HH tidak digunakan, hal ini dilakukan karena subband yang paling banyak menyimpan informasi pada citra adalah subband LL. Skema subband DWT yang digunakan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Skema subband DWT yang digunakan dalam penelitian

### 3.3.2 Chinese Remainder Theorem (CRT)

Metode CRT digunakan untuk melakukan proses *embedding watermark*. Metode ini dipilih karena memiliki tingkat keamanan yang baik, karena untuk dapat menjalankan proses ekstraksi, metode ini mengharuskan pengguna untuk mengetahui nilai S1 dan S2 (sesuai yang dijelaskan dalam subbab 2.2.3).

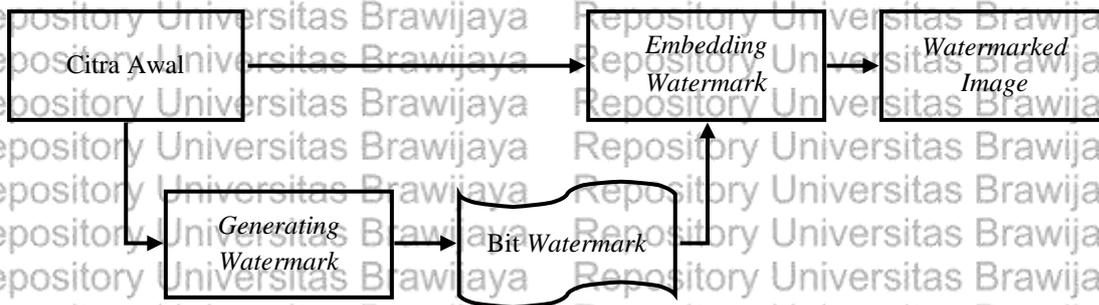
Selain memiliki tingkat keamanan yang baik, metode ini juga mampu menghasilkan citra embedding dengan kualitas yang baik pula. Hal ini dapat dilakukan dengan mengendalikan jumlah bit yang disisipkan, dengan begitu perubahan yang terjadi pada citra akibat proses *embedding* akan sangat minimal, sehingga citra hasil proses watermarking akan memiliki nilai PSNR yang tinggi. Terakhir, keuntungan lain menggunakan metode ini adalah memiliki proses perhitungan yang sederhana, yakni hanya dengan operasi modulus, hal ini mampu meningkatkan waktu proses *embedding*.

### 3.4 Konsep Pengembangan Ilmu

Dari penjabaran metode yang relevan diatas, maka dapat dibuat konsep solusi untuk mengintegrasikan metode yang digunakan.

#### 3.4.1 Proses Generate dan Embedding Watermark

Berbeda dengan skema *digital watermarking* untuk perlindungan terhadap hak cipta, yang kebanyakan menyisipkan logo pihak pembuat citra, atau bisa dikatakan citra induk dan citra *watermark* merupakan citra yang berbeda. Pada skema *digital watermarking* untuk otentikasi, citra induk dan citra *watermark* merupakan citra yang sama. Secara garis besar skema proses *digital watermarking* dapat dilihat pada Gambar 3.3.



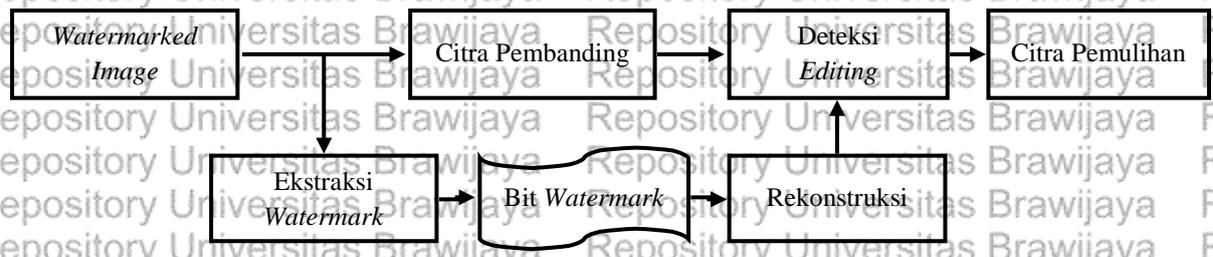
Gambar 3.3 Proses generate dan embedding watermark

Sesuai yang dijelaskan sebelumnya, citra awal akan digandakan untuk proses *generating* dan *embedding watermark*. Citra awal untuk proses *generate* difungsikan sebagai citra *watermark*, sedangkan citra awal untuk proses *embedding* berfungsi sebagai citra induk. Citra *watermark* akan diproses dengan metode DWT, dan dari proses tersebut akan menghasilkan bit-bit *watermark*. Bit *watermark* inilah yang akan disisipkan pada citra induk dengan metode CRT dalam proses *embedding*. Hasil dari proses *embedding* adalah citra ter-*watermark*.

### 3.4.2 Proses Ekstraksi, Deteksi dan Pemulihan

Untuk melakukan proses deteksi *editing*, akan dilakukan ekstraksi pada citra ter-*watermark* terlebih dahulu guna mendapatkan kembali bit *watermark* yang disisipkan. Proses ekstraksi dilakukan dengan mengambil setiap piksel citra yang mengandung *watermark* kemudian dilakukan perhitungan dengan metode CRT seperti pada proses *embedding*, hasilnya dari proses ini adalah bit *watermark* yang disisipkan. Bit-bit *watermark* yang sudah didapatkan tersebut akan direkonstruksi dan dibandingkan dengan citra pembanding sehingga diketahui area yang mengalami penyuntingan. Citra pembanding disini adalah citra hasil rekonstruksi dari bit *watermark* pada proses *generate watermark* yang disimpan dalam file kunci.

Area yang terdeteksi terdapat proses penyuntingan akan diproses untuk sebisa mungkin dikembalikan menjadi citra awal, hasil dari proses ini disebut dengan citra pemulihan. Proses pemulihan dilakukan dengan dengan cara menjadikan citra ter-*watermark* sebagai citra utama dan area yang terdeteksi terdapat penyuntingan akan diganti dengan nilai intensitas citra pembanding. Secara umum alur prosesnya dapat dilihat di Gambar 3.4.



Gambar 3.4 Skema proses deteksi penyuntingan

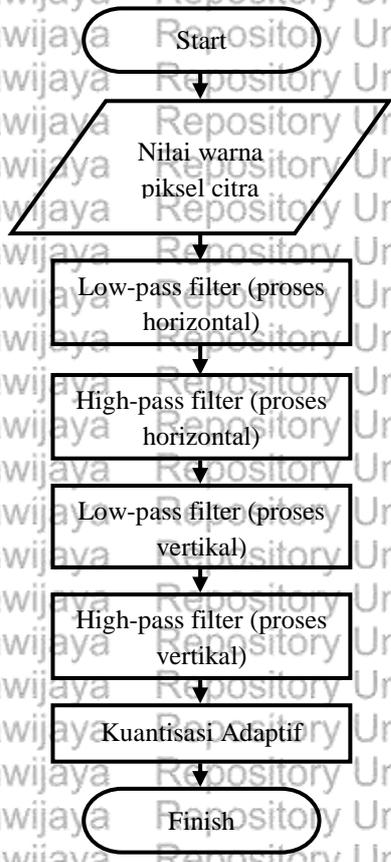
### 3.5 Konsep Algoritma

Konsep algoritma dalam penelitian ini dapat diuraikan sebagai berikut:

#### 3.5.1 Discrete Wavelet Transform (DWT) Haar

Berikut adalah urutan proses metode DWT Haar:

1. Nilai masukan berupa citra dengan ukuran yang habis dibagi 2, baik lebar maupun tinggi.
2. Proses horizontal, proses citra dengan *low-pass filter* dan *high-pass filter* (lihat Gambar 2.3 (b)).
3. Proses vertikal, lakukan pemrosesan citra dengan *low-pass filter* dan *high-pass filter* (lihat Gambar 2.3 (c)).
4. Lakukan kuantisasi adaptif DWT

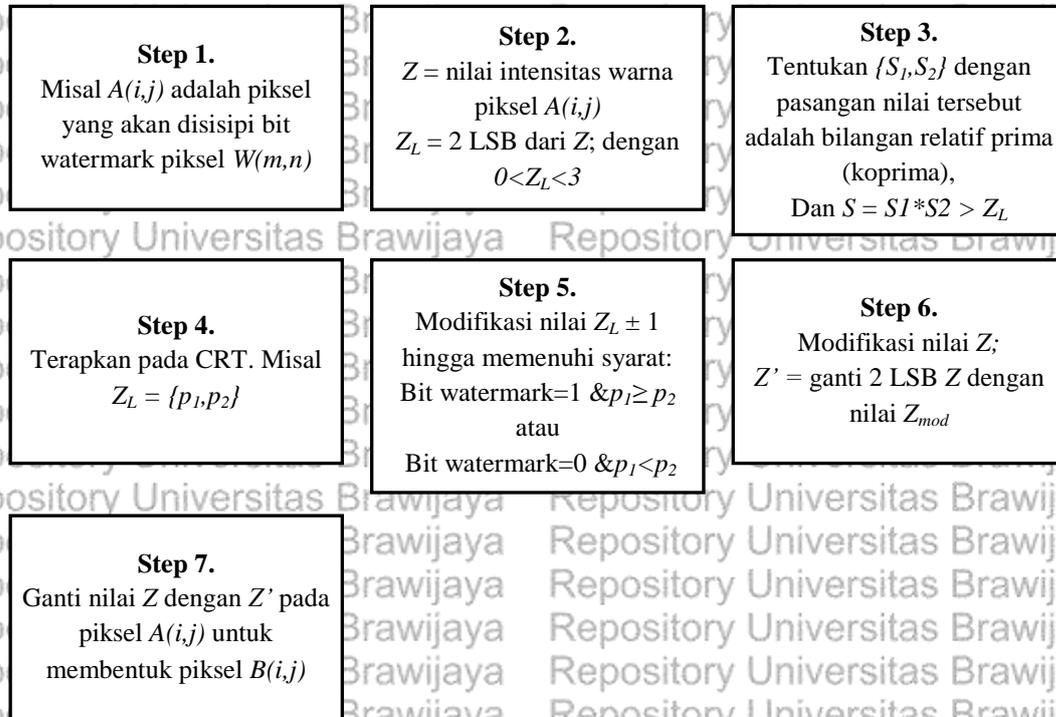


Gambar 3.5 Konsep metode DWT Haar



### 3.5.2 Chinese Remainder Theorem (CRT)

Untuk proses *embedding watermark* dan citra induk, langkah penggabungan menggunakan metode CRT yang sudah disesuaikan dengan metode DWT ditunjukkan pada Gambar 3.6.



Gambar 3.6 Konsep metode CRT dalam proses *embedding watermark*

Proses penggabungan pada Gambar 3.6 algoritma CRT digunakan untuk merubah 2 bit LSB pada nilai piksel citra induk yang akan disisipi bit *watermark*. Dari penggunaan metode ini dan kombinasi bit *watermark* pada tiap yang disisipkan hanya 1 bit, maka nilai intensitas warna piksel hanya berubah maksimal 1 satuan.

### 3.6 Hipotesis

Berdasarkan perancangan skema *digital watermarking* yang dilakukan, maka model ini memiliki kelebihan untuk dapat memproses citra berwarna dengan ukuran yang lebih fleksibel tanpa harus terbatas dengan *block base*, dan dengan menggunakan kuantisasi adaptif kualitas citra rekonstruksi akan lebih baik daripada metode kuantisasi *uniform* yang biasa dilakukan pada proses *digital watermarking*. Pengaplikasian metode CRT juga memiliki kelebihan dari sisi keamanan.



### 3.7 Alat Penelitian

Peralatan yang digunakan dalam pengujian penelitian ini baik *hardware* maupun *software* antara lain:

1. Laptop dengan spesifikasi prosesor i5-4200U 1.60 GHz, RAM 8 GB.
2. Bahasa pemrograman yang digunakan adalah Java JDK 1.8, dan GUI memanfaatkan Java Swing.
3. *Integrated Development Environment* (IDE) yang digunakan adalah Netbeans 8.2.
4. *Library* tambahan yang digunakan adalah OpenCV 3.2.0. Digunakan untuk melakukan operasi matriks pada citra.
5. Proses penyuntingan menggunakan *software* Adobe Photoshop CC.
6. OS yang digunakan adalah Windows 8.

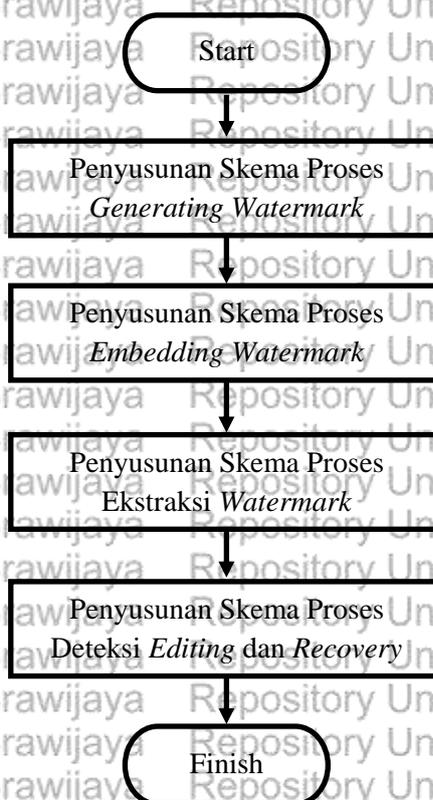




## BAB IV METODE PENELITIAN

### 4.1 Metode Penelitian

Metodologi yang dilakukan dalam penelitian tentang skema *digital watermarking* ini melalui tahapan-tahapan seperti yang ditunjukkan pada Gambar 4.1.



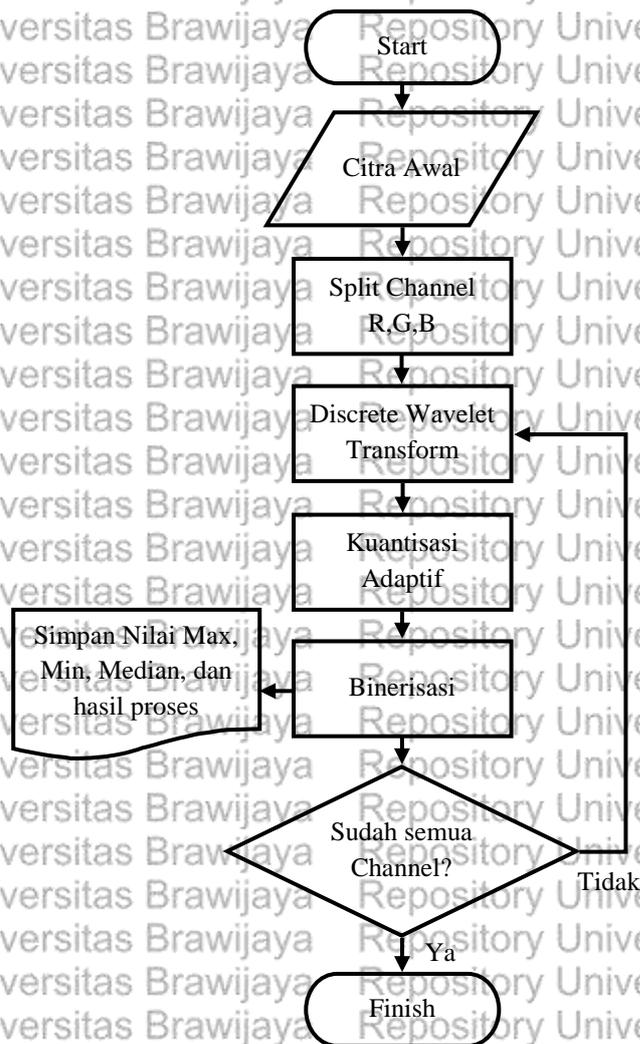
Gambar 4.1 Metodologi penelitian

### 4.2 Skema Proses *Generating Watermark*

Proses *generating watermark* melibatkan suatu proses kompresi citra dengan metode *transform coding*, yang terdiri dari *Discrete Wavelet Transform* dan kuantisasi adaptif.

Proses dilakukan untuk mendapatkan bit *watermark* yang akan disisipkan pada citra induk.

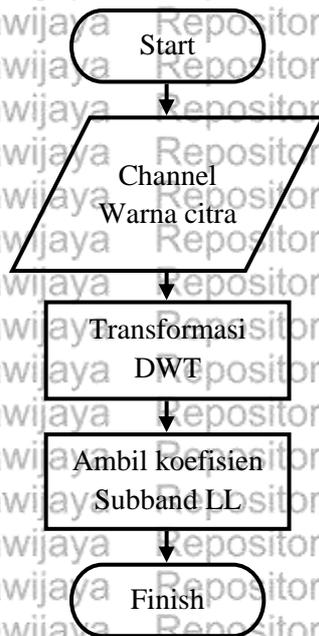
Proses dimulai dengan membagi citra berdasarkan *channel* warna, setelah itu dilakukan proses DWT dan kuantisasi. Hasil dari kuantisasi akan direpresentasikan dengan nilai biner melalui proses binerisasi. Alur proses yang dirancang dapat dilihat dalam Gambar 4.2.



Gambar 4.2 Alur proses generating watermark

#### 4.2.1 Proses Discrete Wavelet Transform

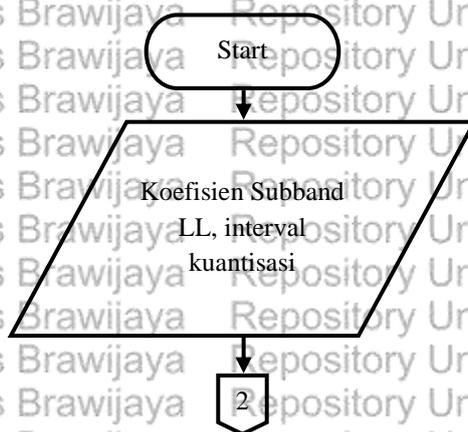
Tahap kompresi citra *watermark* dimulai dengan proses transformasi DWT. Seperti yang sudah dijelaskan dalam sub-bab 3.3.1, transformasi akan dilakukan dalam skala 1 untuk mendapatkan subband LL. Alur proses transformasi dan pemilihan subband dapat dilihat pada Gambar 4.3.

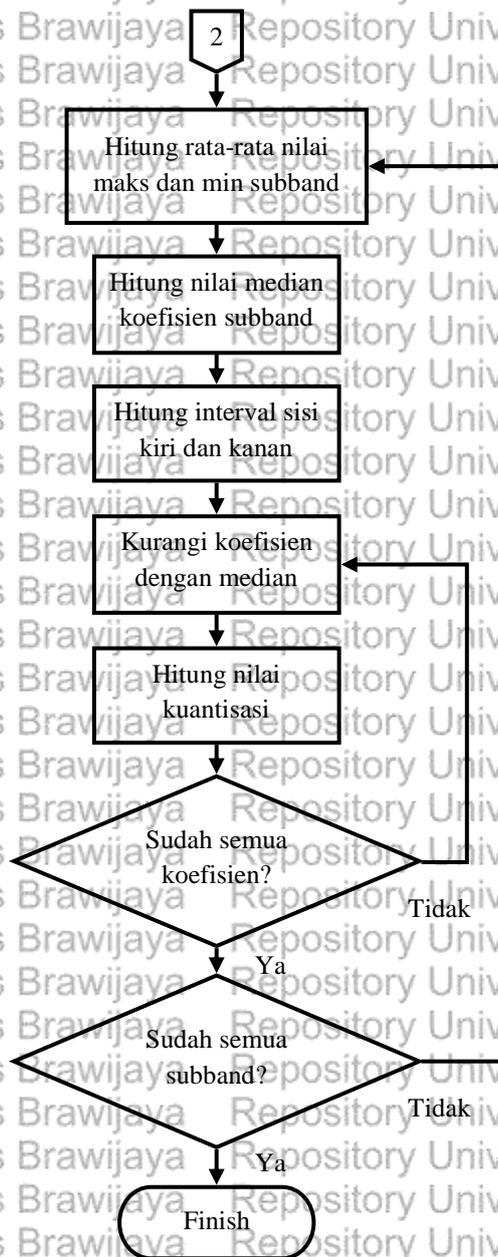


Gambar 4.3 Alur proses Discrete Wavelet Transform

#### 4.2.2 Proses Kuantisasi Adaptif

Proses kuantisasi adaptif berfungsi untuk merepresentasikan koefisien transformasi menjadi nilai yang lebih kecil. Parameter yang akan disimpan dalam proses ini adalah nilai interval kuantisasi, nilai interval sisi kiri dan kanan, nilai maksimum dan minimum, serta nilai median pada masing-masing subband *watermark*. Alur proses kuantisasi adaptif dapat dilihat pada Gambar 4.4 sebagai berikut:



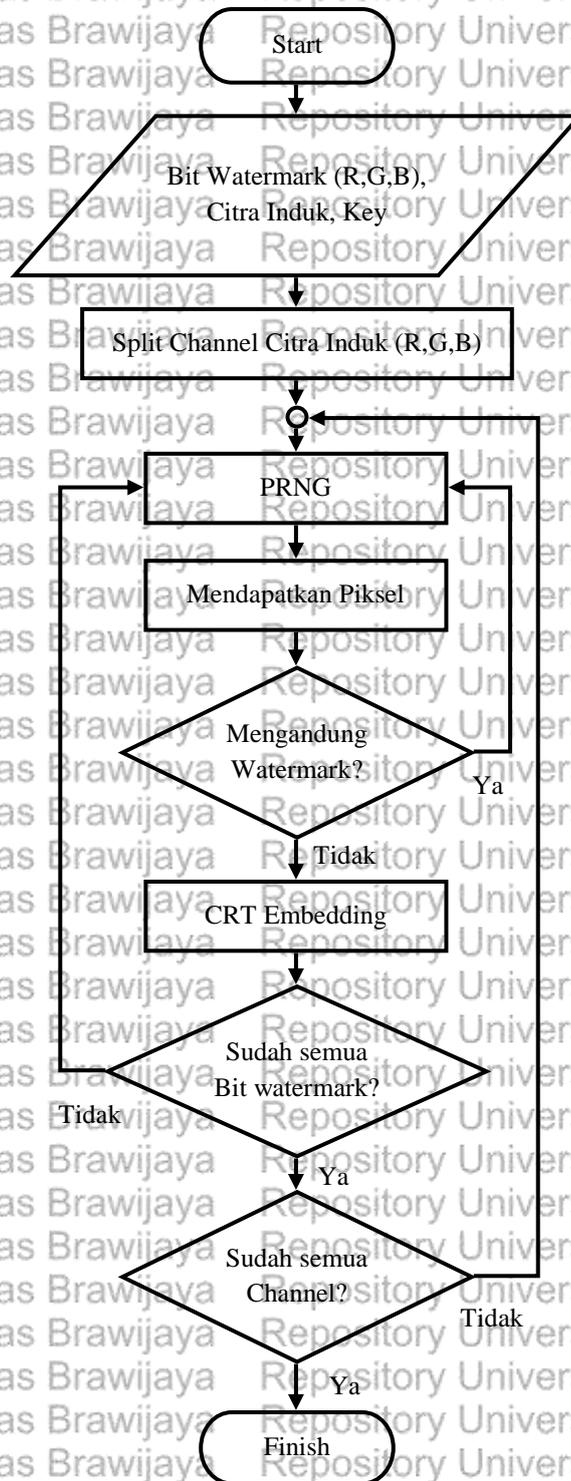


Gambar 4.4 Alur proses kuantisasi adaptif

### 4.3 Skema Proses *Embedding Watermark*

Bit *watermark* yang sudah didapatkan dari proses *generate watermark* selanjutnya akan disisipkan pada citra induk melalui proses *embedding watermark*. Sama halnya dengan proses sebelumnya, proses *embedding* dilakukan pada tiga channel warna *red*, *green* dan *blue*, artinya bit *watermark* akan disisipkan pada citra induk dengan *channel* warna yang sama. Proses ini melibatkan sebuah nilai *Key* yang bertugas sebagai *seed* pada proses random PRNG sehingga ketika proses ekstraksi dapat di dapatkan kembali nilai random yang sama. Proses random berfungsi untuk mengacak posisi piksel citra induk yang akan disisipi bit *watermark*. Dengan metode PRNG, bit *watermark* yang

sudah teracak pada piksel citra induk dapat disusun kembali pada urutan semula sehingga dapat dilakukan proses rekonstruksi citra. Alur proses *embedding watermark* secara lengkap dapat dilihat di Gambar 4.5.



Gambar 4.5 Alur proses embedding watermark





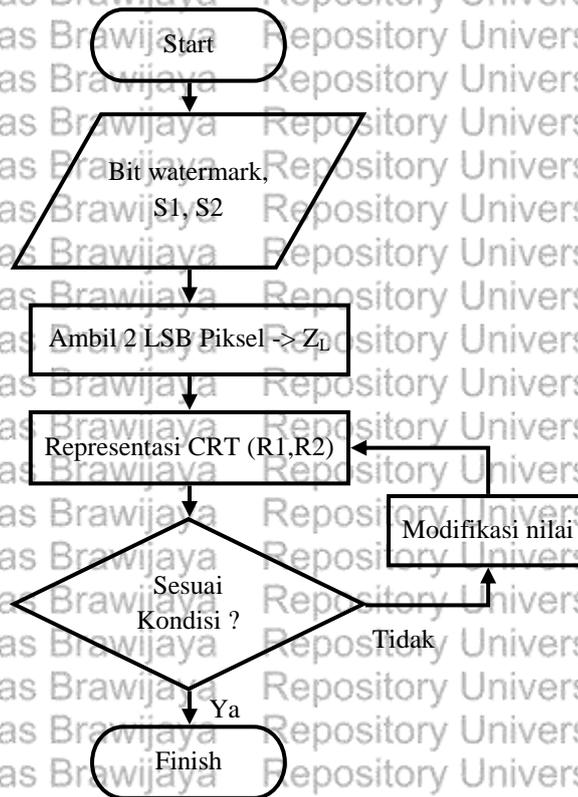
### 4.3.1 Proses CRT Embedding

Setelah mendapatkan lokasi piksel yang akan disisipi bit *watermark*, maka dilakukan proses CRT *embedding*. Selain data piksel, proses ini membutuhkan nilai S1 dan S2 yang merupakan koefisien tetap yang sudah diatur sejak awal. Dari nilai intensitas piksel, diambil 2 LSB (*Least Significant Bit*) dan disimpan di variabel  $Z_L$  yang kemudian bersama dengan nilai S1 dan S2 dilakukan perhitungan untuk mencari nilai representasi R1 dan R2. Proses selanjutnya adalah membandingkan antara nilai bit *watermark* dengan nilai R1 dan R2 sesuai dengan kondisi berikut:

$$\text{if Watermark bit} == 1 \ \&\& \ R1 \geq R2$$

$$\text{if Watermark bit} == 0 \ \&\& \ R1 < R2$$

Jika kondisi tersebut terpenuhi maka proses akan dilanjutkan ke piksel selanjutnya, dan apabila tidak terpenuhi maka akan dilakukan penyesuaian pada nilai intensitas piksel dengan mengurangi atau menambahkan 1 nilai satuan. Alur proses secara lengkap dapat dilihat dalam Gambar 4.6 berikut.

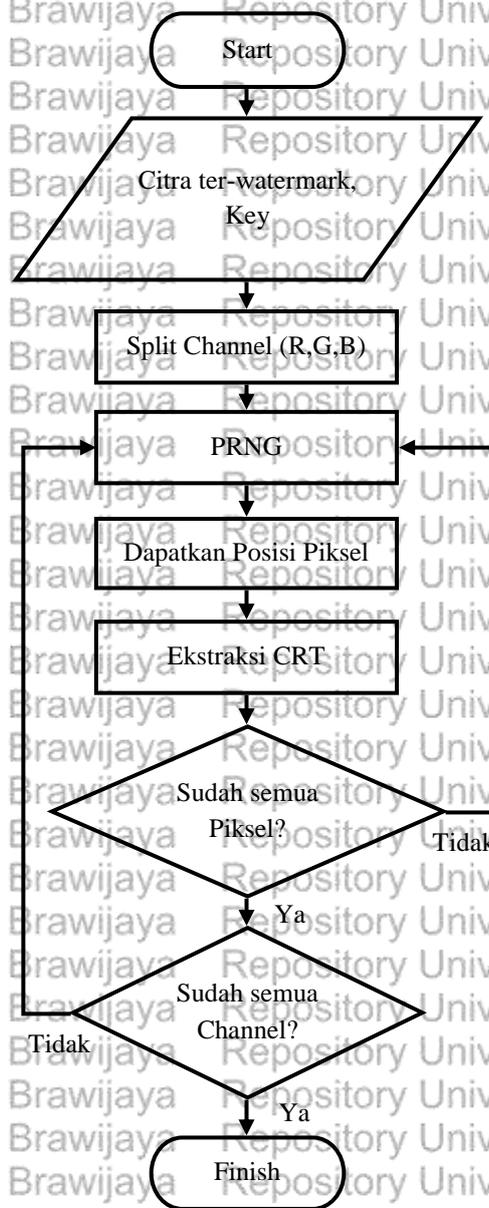


Gambar 4.6 Alur proses CRT embedding



#### 4.4 Skema Proses Ekstraksi Watermark

Tahapan penelitian selanjutnya adalah penyusunan skema ekstraksi *watermark*. Proses ini membutuhkan citra ter-*watermark*, *Key* dan koefisien  $S_1$  dan  $S_2$ . Proses ekstraksi dimulai dengan memisahkan *channel* warna RGB pada citra, kemudian dengan memanfaatkan *key* yang sama pada proses *embedding*, dengan PRNG akan didapatkan nilai acak yang sama, sehingga dapat diketahui lokasi piksel yang mengandung *watermark*. Urutan piksel tersebut akan melalui proses ekstraksi CRT untuk mendapatkan bit *watermark* yang disisipkan pada piksel tersebut. Alur proses ekstraksi *watermark* dijabarkan dalam Gambar 4.7.

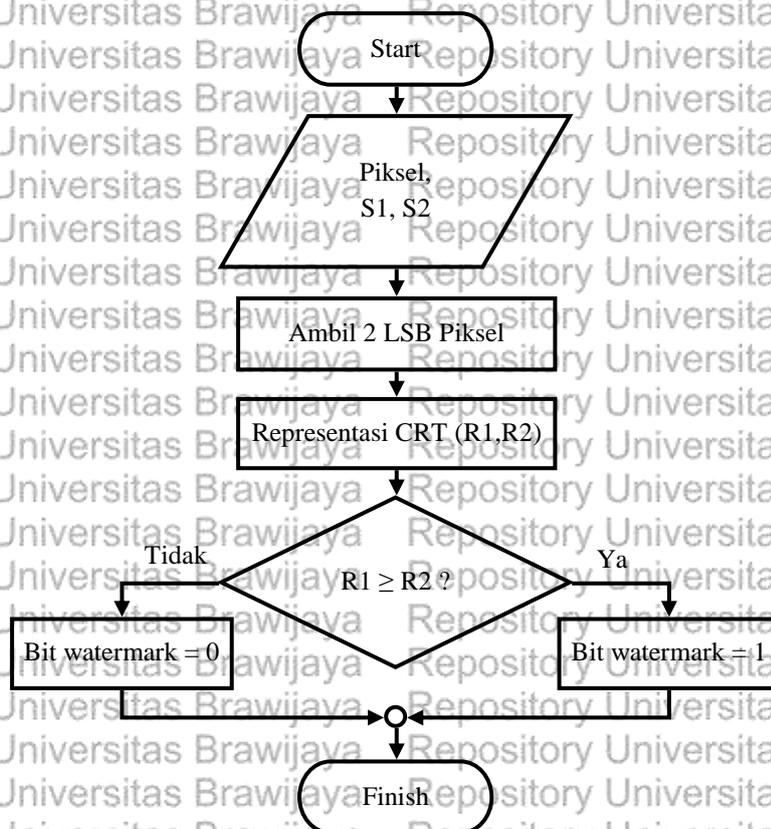


Gambar 4.7 Skema alur proses ekstraksi watermark



#### 4.4.1 Proses Ekstraksi CRT

Sama halnya pada proses CRT *embedding*, pada proses ekstraksi dibutuhkan data piksel dan koefisien S1 dan S2. Dari data piksel diperoleh 2 LSB nilai intensitas warna, kemudian dengan mengkombinasikan nilai S1, S2 dan 2 LSB tersebut dihitung nilai representasi CRT untuk mendapatkan variabel R1 dan R2. Setelah itu nilai R1 dan R2 akan dibandingkan apakah nilai R1 sama atau lebih besar dari R2, jika kondisi tersebut terpenuhi maka ditetapkan bahwa nilai bit *watermark* adalah '1', dan jika tidak maka ditetapkan bahwa nilai bit *watermark* yang disisipkan pada piksel tersebut adalah '0'. Untuk lebih lengkapnya, alur proses ekstraksi CRT dapat dilihat pada Gambar 4.8.

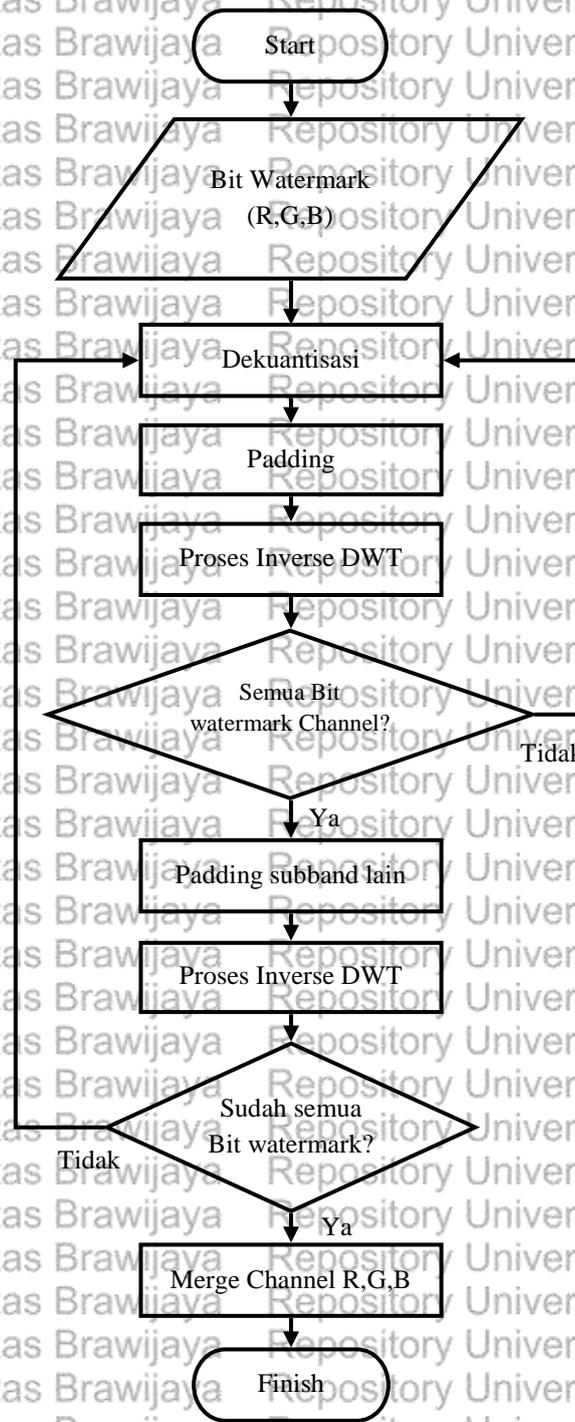


Gambar 4.8 Alur proses ekstraksi CRT

#### 4.5 Skema Proses Rekonstruksi

Setelah mendapatkan bit *watermark* pada proses ekstraksi, proses selanjutnya adalah merekonstruksi dari bit *watermark* tersebut menjadi citra awal. Hasil yang diperoleh dari proses rekonstruksi berupa bit *watermark* yang sudah tersusun berdasarkan subband sesuai *channel* masing-masing. Proses rekonstruksi dimulai dengan tahapan dekuantisasi yang bertujuan untuk merubah bit *watermark* menjadi nilai koefisien transformasi. Kemudian hasil proses tersebut akan melalui proses *padding* untuk menambal koefisien subband yang dihilangkan (LH, HL dan HH) saat proses transformasi DWT skala 1.

Proses *padding* pada subband tersebut diberikan nilai '0' dengan ukuran masing-masing subband adalah 1/4 ukuran citra ter-*watermark*. Setelah itu dilakukan inverse transformasi DWT untuk mendapatkan nilai koefisien keseluruhan. Proses tersebut akan diulang untuk *channel* warna yang lain, dan setelah semua *channel* terproses, maka akan dilakukan *merge* pada ketiga *channel* warna tersebut, sehingga diperoleh citra rekonstruksi. Alur proses rekonstruksi dijelaskan pada Gambar 4.9.

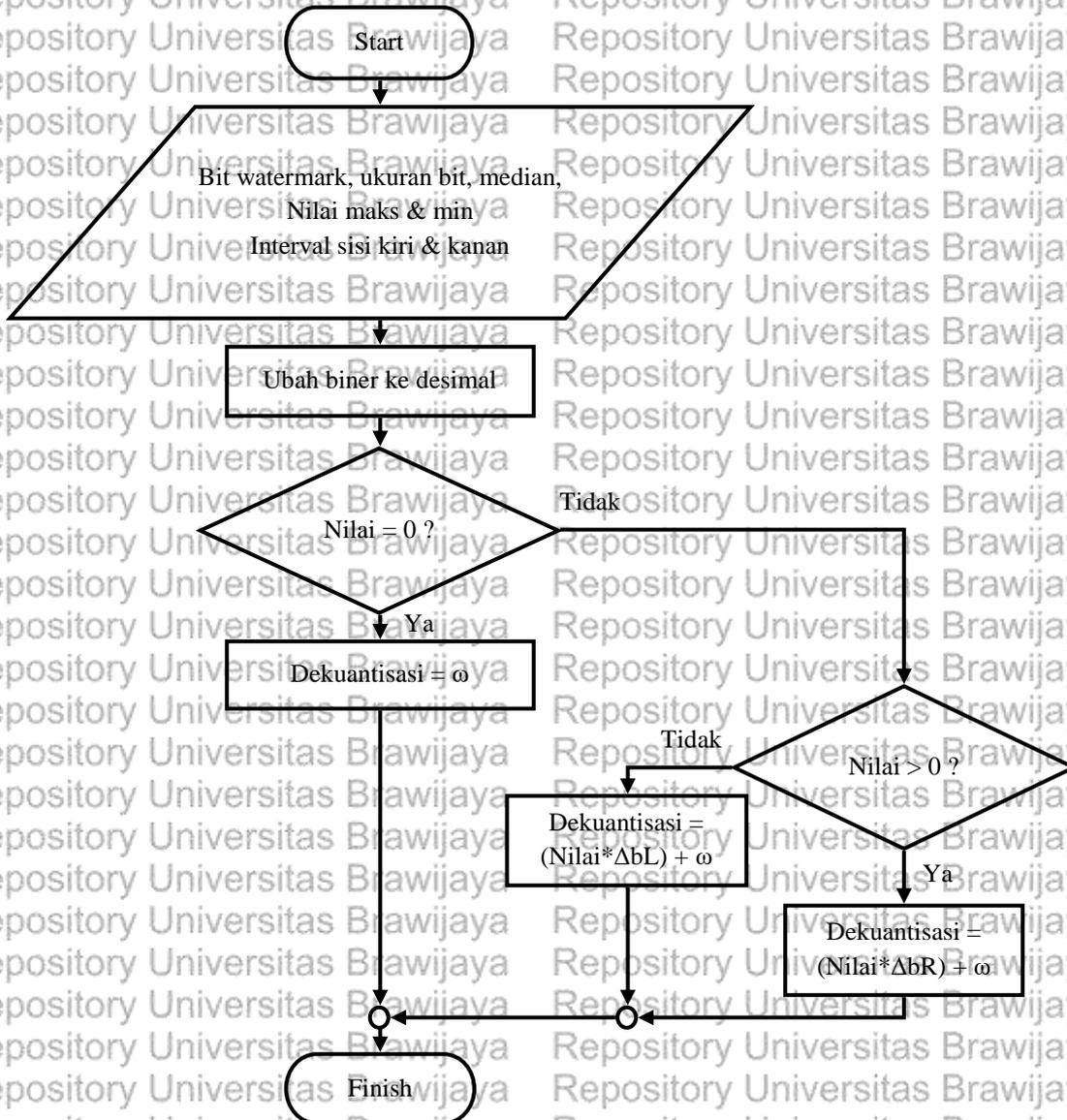


Gambar 4.9 Alur proses rekonstruksi



### 4.5.1 Proses Dekuantisasi

Proses dekuantisasi membutuhkan beberapa input yakni ukuran bit *watermark* (interval kuantisasi), interval sisi kiri dan kanan, nilai maksimum dan minimum, serta median dari masing-masing subband yang dijadikan *watermark*. Proses dimulai dengan mengubah bit *watermark* menjadi koefisien kuantisasi, setelah itu akan dicek dan diproses sesuai Persamaan 2.7.



Gambar 4.10 Alur proses dekuantisasi

### 4.6 Proses Rekonstruksi Citra Pembanding

Untuk membuat citra pembanding dimulai dengan pengambilan bit *watermark* yang disimpan di *file* kunci. Bit *watermark* tersebut akan direkonstruksi sesuai alur pada sub-bab 4.7. Hasil rekonstruksi inilah yang menghasilkan citra pembanding.

#### 4.7 Proses Deteksi dan Pemulihan Area Penyuntingan

Proses ini dilakukan dengan membandingkan citra pembandingan dan citra rekonstruksi, yakni dengan membandingkan nilai *error* pada setiap piksel kedua citra tersebut. Nilai *error* didapatkan dengan memanfaatkan parameter MSE. Untuk mencari nilai *error* ditunjukkan pada persamaan berikut:

$$\theta_{i,j} = [x(r,l) - x(p,l)]^2 \quad (4.1)$$

Variabel  $x(r,l)$  dalam persamaan tersebut mewakili nilai piksel dari citra rekonstruksi, sedangkan  $x(p,l)$  mewakili nilai piksel dari citra pembandingan.

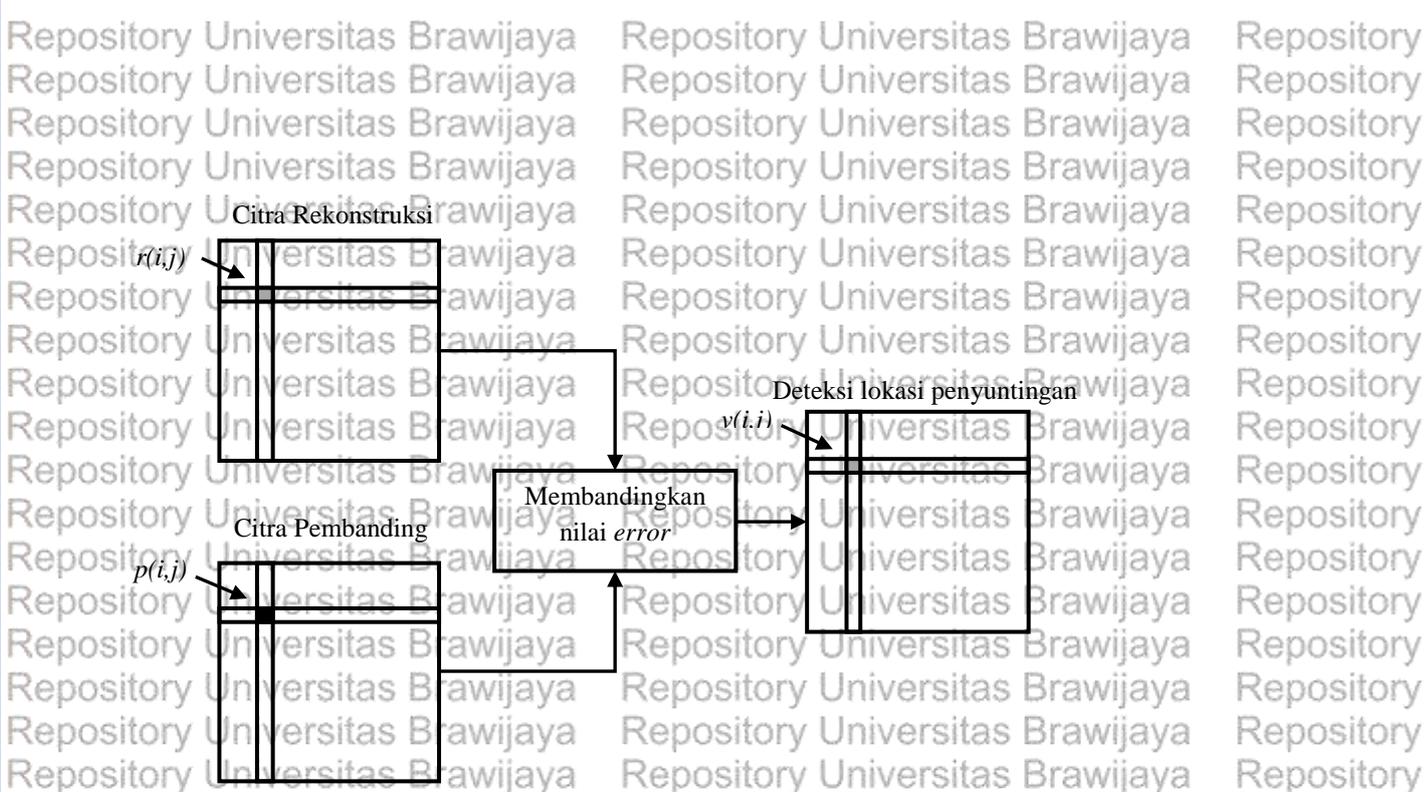


Gambar 4.11 Pembentukan citra pemulihan

Setelah mendapatkan citra pembandingan dan citra rekonstruksi maka akan dilakukan pengecekan untuk mendeteksi apakah sudah terjadi penyuntingan atau tidak. Kedua citra tersebut akan dicek dengan cara yang ditunjukkan pada Gambar 4.12. Secara teori karena pada metode CRT hanya akan memodifikasi dengan menambah atau mengurangi 1 nilai intensitas warna (lihat subbab 3.3.1), maka batas nilai *error* yang ditolerir untuk proses pendeteksian jika mengacu pada persamaan (4.1) adalah  $1^2$  atau bernilai 1.

Apabila sudah didapatkan lokasi piksel yang sudah mengalami penyuntingan maka proses selanjutnya adalah penggabungan antara citra ter-watermark dan citra pembandingan.

Nilai intensitas piksel yang sudah mengalami penyuntingan akan digantikan dengan nilai intensitas piksel citra pembandingan, dengan begitu hasil citra pemulihan akan memiliki PSNR yang lebih baik atau dengan kualitas yang mendekati citra asli.



Gambar 4.12 Deteksi lokasi penyuntingan

#### 4.8 Data Penelitian

Data yang digunakan pada penelitian ini adalah 21 citra berwarna dengan rincian 11 citra klasik yang biasa digunakan dalam penelitian citra digital, 9 citra dokumen dengan variasi warna latar, serta 1 citra dengan ukuran besar.

Tabel 4.1 Data penelitian

No.	Nama Data	Ukuran
1	Airplane	512x512
2	Baboon	500x480
3	Lenna	512x512
4	Pepper	512x512
5	Barbara	720x576
6	Boats	787x576
7	Cable Car	512x480
8	Gold Hill	720x576
9	Monarch	768x512
10	Sail Boat	512x512
11	Zelda	787x576
12	Cek Bank CIMB	800x406
13	Cek Bank Danamon	1400x544
14	Cek Bank BCA	1050x408
15	Kartu Kredit	634x403
16	KTP	984x641
17	Paspor	800x561
18	Sertifikat Harvard	756x580
19	Sertifikat PAP UGM	1200x654
20	Sertifikat TPA	787x576
21	Boardwalk	2000x1335



Pemilihan data pembanding di Tabel 4.1 didasarkan pada data uji pada penelitian sebelumnya. Sedangkan data uji citra dokumen meliputi citra dokumen sertifikat dengan variasi detail yang berbeda seperti sertifikat dengan atau tanpa foto serta sertifikat dengan warna latar polos atau yang terdiri dari beberapa warna.

Data pengujian akan dicoba dengan beberapa parameter perubahan yakni melakukan proses penyuntingan pada data hasil proses *watermarking* dengan ukuran area penyuntingan yang bervariasi. Selain itu akan dilakukan pengujian terhadap daya tahan terhadap jenis serangan lain seperti kompresi JPEG citra.



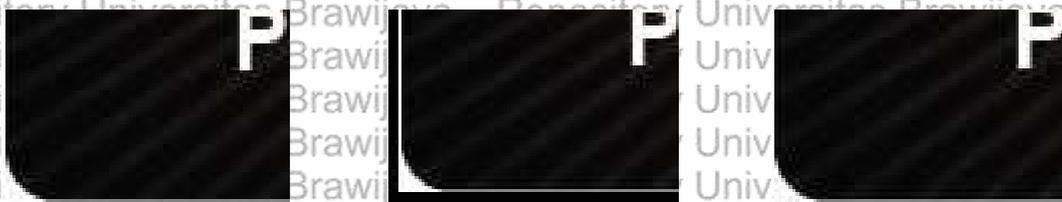
## BAB V HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas hasil penelitian dari skema yang dirancang pada bab sebelumnya. Pembahasan hasil pengujian meliputi hasil pada setiap proses mulai dari proses *generate*, *embedding watermark*, proses ekstraksi dan terakhir pengujian daya tahan skema yang dirancang terhadap beberapa jenis serangan.

Parameter yang digunakan untuk menilai kualitas pada skema yang dirancang antara lain nilai PSNR yang digunakan untuk mengukur kemiripan antara citra awal dan citra hasil pemrosesan, nilai NC untuk mengukur kualitas daya tahan citra hasil proses *watermarking* terhadap serangan, dan waktu yang dibutuhkan pada tiap proses.

### 5.1 *Generate Watermark*

Proses *generate* dimulai setelah terlebih dahulu melakukan *preprocessing* pada citra asli. Citra akan dicek terlebih dahulu untuk mengetahui apakah ukuran lebar dan tinggi citra habis dibagi 2, jika tidak maka akan dilakukan penambalan (*padding*) dengan memberikan nilai intensitas warna '0' atau warna hitam. Contoh bagian hasil *preprocessing* pada citra berukuran 634x403 ditunjukkan pada Gambar 5.1.



(a) Citra Asli

(b) Preprocessing TLDCCT

(c) Preprocessing DWT

Gambar 5.1 Perbesaran citra pada area penambalan (*padding*)

Pada Gambar 5.1 terlihat bahwa proses penambalan dengan metode yang bergantung *block based 8x8* seperti metode TLDCCT dan DCT pada penelitian sebelumnya memiliki area penambalan yang lebih besar dibandingkan dengan metode DWT, hal ini dikarenakan ukuran citra pada tinggi dan lebarnya harus habis dibagi dengan 8. Jika melihat ukuran citra dapat dihitung bahwa penambalan yang terjadi pada metode TLDCCT totalnya adalah 5 piksel untuk tinggi dan 6 piksel untuk lebarnya, perlu diketahui pada Gambar 5.1(b) total penambalan 5 dan 6 piksel tersebut terbagi rata pada bagian atas dan bawah serta sisi kanan

dan kiri citra. Sedangkan pada metode DWT hanya menambahkan 1 piksel pada tinggi citra, yang terletak pada bagian bawah citra.

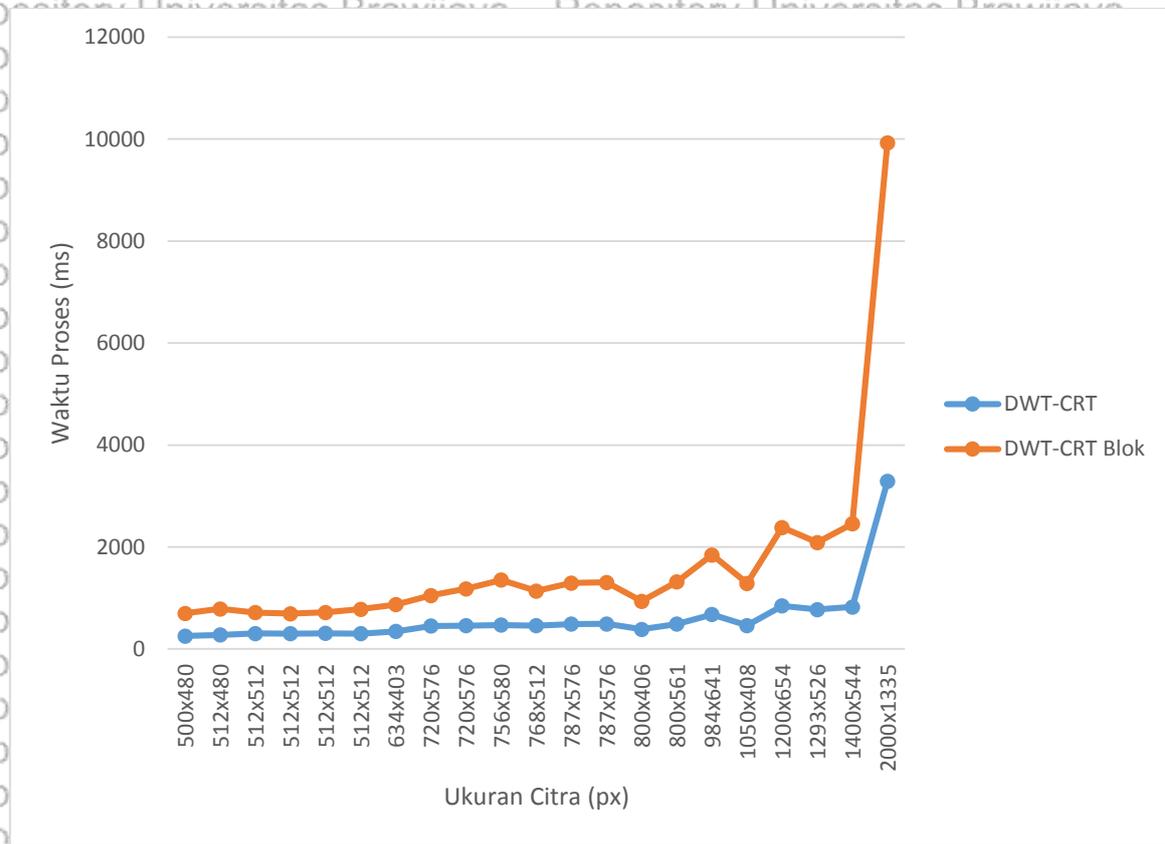
Setelah citra melewati proses *preprocessing* maka akan dilanjutkan dengan proses *generate watermark* sesuai dengan skema pada sub bab 4.2. Proses diawali dengan membagi citra menjadi tiap bagian channel warna yakni *Red*, *Green* dan *Blue*. Masing-masing channel warna akan diproses dengan transformasi DWT dan diambil subband LL untuk dilakukan proses kuantisasi adaptif. Koefisien yang didapat dari proses kuantisasi akan diubah menjadi bilangan biner, hasil inilah yang disebut dengan bit *watermark*. Hasil pengujian proses eksekusi *generate watermark* bisa dilihat pada Tabel 5.1.

Tabel 5.1 Waktu proses *generate watermark*

No.	Nama Citra	Ukuran (px)	Bit Kuantisasi	DWT-CRT (ms)	DWT-CRT Blok (ms)
1	Baboon	500x480	4 bit	253	700
2	Cable Car	512x480	4 bit	276	788
3	Airplane	512x512	4 bit	304	716
4	Lenna	512x512	4 bit	301	694
5	Pepper	512x512	4 bit	307	719
6	Sail Boat	512x512	4 bit	301	780
7	Kartu Kredit	634x403	4 bit	344	872
8	Barbara	720x576	4 bit	452	1049
9	Goldhill	720x576	4 bit	457	1180
10	Sertifikat Harvard	756x580	4 bit	469	1353
11	Monarch	768x512	4 bit	458	1136
12	Boats	787x576	4 bit	488	1294
13	Zelda	787x576	4 bit	491	1306
14	Cek Bank CIMB	800x406	4 bit	386	937
15	Paspor	800x561	4 bit	488	1316
16	KTP	984x641	4 bit	677	1844
17	Cek Bank BCA	1050x408	4 bit	462	1286
18	Sertifikat PAP	1200x654	4 bit	845	2384
19	Sertifikat TPA	1293x526	4 bit	775	2090
20	Cek Bank	1400x544	4 bit	825	2456
21	Boardwalk	2000x1335	4 bit	3288	9928

Dari Tabel 5.1 tampak bahwa waktu yang dibutuhkan untuk proses *generate watermark* dengan citra yang memiliki variasi ukuran 500x480 piksel sampai dengan 2000x1335 piksel memiliki rentang nilai waktu proses antara 253 ms sampai 3288 ms. Hal ini membuktikan bahwa semakin besar ukuran citra maka waktu yang dibutuhkan melakukan proses *generate* akan semakin lama, kondisi ini dikarenakan semakin besar ukuran citra maka jumlah piksel yang diproses juga semakin banyak. Jika skema DWT-CRT yang dirancang tidak berbasis blok dan langsung diimplementasikan pada piksel citra,

maka pengujian ini juga dilakukan dengan menggunakan skema berbasis blok 8x8 dengan metode DWT-CRT yang memiliki urutan proses yang sama. Hasil dari skema DWT-CRT yang berbasis blok 8x8 memiliki rentang waktu proses antara 694 ms sampai dengan 9928 ms. Untuk lebih memperjelas tabel diatas, Gambar 5.2 berikut adalah grafik pengaruh ukuran citra terhadap waktu proses *generate*.



Gambar 5.2 Grafik pengaruh ukuran citra terhadap waktu proses *generate*

Jika memperhatikan grafik perbandingan waktu proses *generate* pada Gambar 5.2, terlihat bahwa skema DWT-CRT yang berbasis blok 8x8 memiliki waktu proses yang lebih lama dibandingkan dengan skema yang tidak menggunakan basis blok. Hal ini dikarenakan pada skema DWT-CRT blok, setiap proses yang dirancang pada Bab 4 diaplikasikan pada seluruh blok 8x8 citra, sedangkan pada skema DWT-CRT tanpa basis blok, proses langsung diaplikasikan pada citra, atau jika dalam analoginya skema DWT-CRT hanya memiliki 1 blok yakni citra secara keseluruhan, sedangkan pada DWT-CRT blok jika ukuran citra 512x512 piksel maka blok citra berjumlah 4096 blok. Jumlah perulangan yang lebih banyak inilah yang membuat waktu proses lebih lama.

## 5.2 Embedding Watermark

Setelah citra melalui proses *generate watermark* dan telah didapatkan bit *watermark* yang akan disisipkan pada citra induk, maka proses selanjutnya adalah proses *embedding watermark*. Urutan proses yang dilakukan sesuai dengan skema yang dirancang pada sub bab 4.3, dimulai dengan membagi *channel* warna kemudian tiap *channel* warna akan melalui proses PRNG untuk mengacak lokasi piksel yang akan disisipi *watermark*. Proses pengacakan melibatkan citra induk dan *key/seed*, penggunaan *key* berfungsi untuk memastikan nilai random akan tetap sama saat digunakan pada proses ekstraksi.

Jika proses pengacakan sudah dilakukan, selanjutnya nilai biner koefisien hasil *generate* akan dipisah per bit dan akan disisipkan pada piksel citra induk sesuai aturan CRT. Hasil pengujian proses *embeding* dengan menggunakan bit kuantisasi bernilai 4 dapat dilihat di Tabel 5.2.

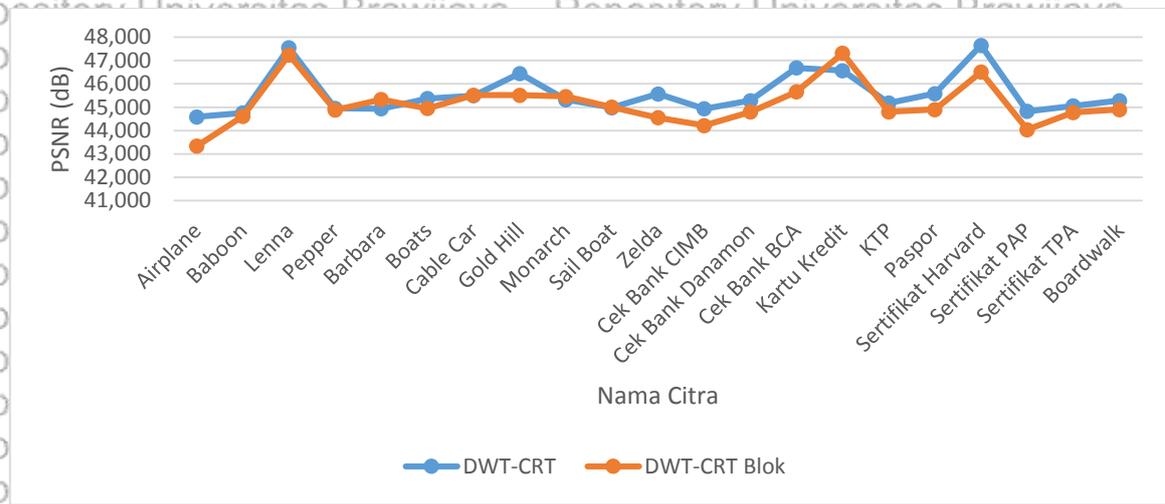
Tabel 5.2 Hasil proses *embedding watermark*

No.	Nama Citra	Ukuran (px)	PSNR (dB)		Waktu Proses (ms)	
			DWT-CRT	DWT-CRT Blok	DWT-CRT	DWT-CRT Blok
1	Baboon	500x480	44,762	44,613	397394	249
2	Cable Car	512x480	45,497	45,515	401846	217
3	Airplane	512x512	44,586	43,339	434295	236
4	Lenna	512x512	47,545	47,239	414513	155
5	Pepper	512x512	44,957	44,884	460543	253
6	Sail Boat	512x512	45,186	45,008	452849	216
7	Kartu Kredit	634x403	44,936	47,318	412857	258
8	Barbara	720x576	44,934	45,330	683294	437
9	Goldhill	720x576	45,573	45,516	661283	430
10	Sertifikat Harvard	756x580	45,588	45,506	701864	499
11	Monarch	768x512	46,683	45,461	630752	328
12	Boats	787x576	45,375	44,951	738237	488
13	Zelda	787x576	45,063	44,552	769343	497
14	Cek Bank CIMB	800x406	46,448	43,223	582373	276
15	Paspor	800x561	45,567	44,603	759237	524
16	KTP	984x641	45,284	44,308	1002383	748
17	Cek Bank BCA	1050x408	44,983	43,658	692384	387
18	Sertifikat PAP	1200x654	47,651	43,045	1458470	1094
19	Sertifikat TPA	1293x526	44,826	44,779	1128473	1005
20	Cek Bank Danamon	1400x544	45,328	43,802	1374628	1083
21	Boardwalk	2000x1335	45,285	44,404	16892738	12118

Hasil proses *embedding* pada Tabel 5.2 menunjukkan bahwa skema DWT-CRT memiliki rentang nilai PSNR antara 44,586 dB sampai 47,651 dB, dengan waktu proses berkisar antara 397394 ms (6,62 menit) sampai 16892738 ms (4,6 jam). Sedangkan untuk

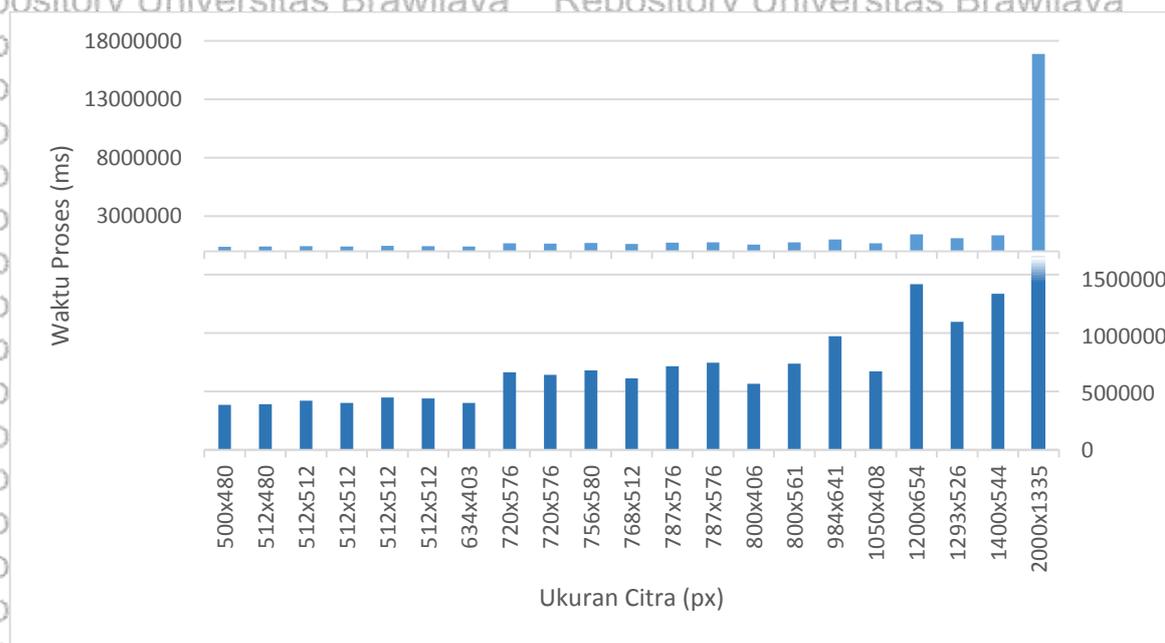


skema DWT-CRT berbasis blok 8x8 menghasilkan nilai PSNR antara 43,045 dB sampai 47,318 dB dan waktu proses antara 155 ms sampai 12118 ms (12 detik). Dari hasil tersebut dapat diketahui bahwa ukuran citra tidak mempengaruhi nilai PSNR, akan tetapi tetap berpengaruh terhadap waktu proses. Untuk melihat perbedaan nilai PSNR proses embedding dapat dilihat grafik pada Gambar 5.3.



Gambar 5.3 Grafik perbandingan PSNR *embedding watermark* metode DWT-CRT

Hasil PSNR pada proses *embedding* skema DWT-CRT yang berbasis blok maupun yang tidak memiliki nilai yang relatif sama, hal ini dikarenakan bit kuantisasi yang digunakan sama yakni 4, sehingga jumlah bit *watermark* yang disisipkan juga berjumlah sama. Sedangkan untuk waktu proses skema DWT-CRT yang tidak berbasis blok dapat dilihat pada grafik di Gambar 5.4.



Gambar 5.4 Grafik waktu proses *embedding* skema DWT-CRT

Berikutnya hasil *embedding* skema ini akan dibandingkan dengan penelitian sebelumnya. Sebagai pembanding adalah penelitian dengan metode DCT-CRT dan TLDCT-CRT, secara garis besar pada penelitian tersebut memiliki skema *embedding* yang sama dengan skema DWT-CRT blok, perbedaannya adalah metode yang digunakan untuk proses *generate* dan bit *watermark* yang disisipkan pada citra induk. Pada metode DCT-CRT dimulai dengan memastikan bahwa ukuran citra harus habis dibagi 8, jika tidak terpenuhi maka akan dilakukan penambalan (*padding*), selanjutnya pada tiap blok akan diproses DCT dan diambil 6 koefisien sisi kiri atas sesuai dengan aturan zig-zag kuantisasi DCT. Total bit *watermark* yang akan disisipkan berjumlah 22 bit.

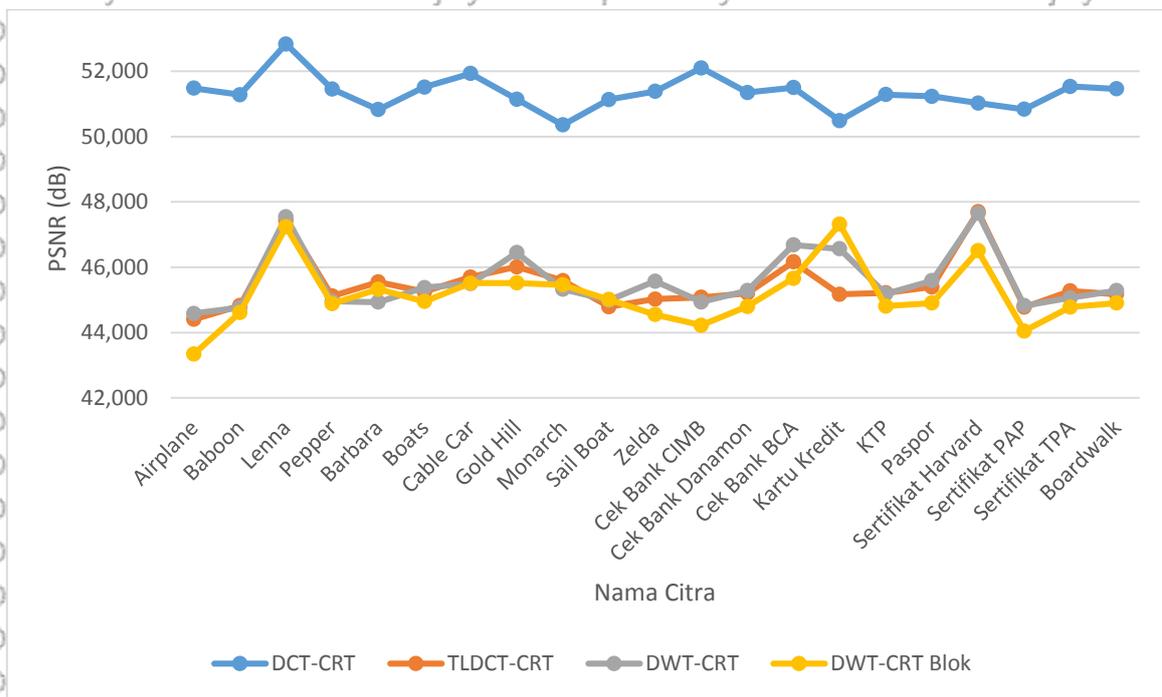
Sama halnya dengan penelitian DCT-CRT, pada skema TLDCT-CRT, ukuran citra awal juga harus habis dibagi 8. Pada skema ini proses *generate* melibatkan transformasi DCT sebanyak dua kali, blok 8x8 terlebih dahulu akan di proses DCT kemudian diambil koefisien 4x4 sisi kiri atas dan akan diteruskan dengan proses DCT terhadap blok 4x4 tersebut. Pemilihan jumlah bit *watermark* yang disisipkan bernilai 4 bit pada masing-masing koefisien, sehingga jika ditotal maka bit *watermark* yang disisipkan adalah 16 koefisien dikali 4 bit atau sama dengan 64 bit. Jadi tiap blok citra induk akan disisipi 1 bit *watermark*. Hasil perbandingan PSNR skema yang dirancang dengan penelitian terdahulu dapat dilihat di Tabel 5.3.

Tabel 5.3 Perbandingan nilai PSNR *embedding* dengan penelitian terdahulu (dalam dB)

No.	Nama Citra	DCT-CRT	TLDCT-CRT	DWT-CRT	DWT-CRT Blok
1	Airplane	51,486	44,404	44,586	43,339
2	Baboon	51,285	44,822	44,762	44,613
3	Lenna	52,837	47,434	47,545	47,239
4	Pepper	51,461	45,118	44,957	44,884
5	Barbara	50,829	45,547	44,934	45,330
6	Boats	51,517	45,251	45,375	44,951
7	Cable Car	51,935	45,698	45,497	45,515
8	Gold Hill	51,146	46,013	46,448	45,516
9	Monarch	50,361	45,591	45,328	45,461
10	Sail Boat	51,137	44,790	44,983	45,008
11	Zelda	51,386	45,024	45,573	44,552
12	Cek Bank CIMB	52,106	45,078	44,936	44,223
13	Cek Bank Danamon	51,351	45,207	45,284	44,802
14	Cek Bank BCA	51,503	46,165	46,683	45,658
15	Kartu Kredit	50,493	45,171	45,567	47,318
16	KTP	51,288	45,216	45,186	44,808
17	Paspor	51,237	45,395	45,588	44,903
18	Sertifikat Harvard	51,028	47,703	47,651	46,506
19	Sertifikat PAP	50,839	44,775	44,826	44,045

No.	Nama Citra	DCT-CRT	TLDCT-CRT	DWT-CRT	DWT-CRT Blok
20	Sertifikat TPA	51,538	45,272	45,063	44,779
21	Boardwalk	51,462	45,175	45,285	44,904

Berdasarkan Tabel 5.3 nilai PSNR hasil proses embedding pada skema DCT-CRT lebih unggul daripada skema lainnya dengan nilai rata-rata 51,344 dB, sedangkan nilai PSNR pada skema DWT-CRT, DWT-CRT blok dan TLDCT-CRT memiliki nilai yang relatif sama. Skema DCT-CRT lebih unggul karena total jumlah bit watermark yang disisipkan lebih sedikit daripada skema lainnya, seperti yang sudah dijelaskan sebelumnya. Untuk lebih memperjelas perbandingan nilai PSNR dapat dilihat grafik pada Gambar 5.5.



Gambar 5.5 Grafik perbandingan nilai PSNR dengan penelitian terdahulu

Evaluasi proses *embedding* selanjutnya adalah membandingkan waktu proses pada ketiga metode diatas, hasilnya dapat dilihat pada Tabel 5.4.

Tabel 5.4 Perbandingan waktu proses *embedding* (dalam ms)

No.	Nama Citra	Ukuran (px)	DCT-CRT	TLDCT-CRT	DWT-CRT	DWT-CRT Blok
1	Baboon	500x480	220	233	397394	249
2	Cable Car	512x480	235	212	401846	217
3	Airplane	512x512	253	235	434295	236
4	Lenna	512x512	239	240	414513	155
5	Pepper	512x512	247	262	460543	253
6	Sail Boat	512x512	250	259	452849	216
7	Kartu Kredit	634x403	230	221	412857	258
8	Barbara	720x576	422	353	683294	437

No.	Nama Citra	Ukuran (px)	DCT-CRT	TLDCT-CRT	DWT-CRT	DWT-CRT Blok
9	Goldhill	720x576	437	449	661283	430
10	Sertifikat Harvard	756x580	538	545	701864	499
11	Monarch	768x512	386	360	630752	328
12	Boats	787x576	542	559	738237	488
13	Zelda	787x576	536	502	769343	497
14	Cek Bank CIMB	800x406	318	293	582373	276
15	Paspor	800x561	374	398	759237	524
16	KTP	984x641	753	767	1002383	748
17	Cek Bank BCA	1050x408	430	413	692384	387
18	Sertifikat PAP	1200x654	1183	1130	1458470	1094
19	Sertifikat TPA	1293x526	1175	1161	1128473	1005
20	Cek Bank Danamon	1400x544	1206	1174	1374628	1083
21	Boardwalk	2000x1335	12578	11903	16892738	12118

Dari Tabel 5.4 dapat dilihat perbedaan waktu skema DWT-CRT yang berbasis blok 8x8 tidak berbeda jauh. Akan tetapi untuk skema DWT-CRT tanpa berbasis blok memiliki selisih yang sangat besar, hal ini dikarenakan proses random piksel pada java membutuhkan waktu yang cukup lama. Skema pada 2 metode penelitian sebelumnya menggunakan dua kali proses random yakni random pada blok dan random pada piksel didalam blok, hal ini juga digunakan pada skema DWT-CRT yang berbasis blok 8x8. Setiap proses random akan dicek satu persatu apakah nilai random sudah digunakan, jika iya maka proses random akan diulang lagi sampai ditemukan nilai yang belum pernah digunakan. Konsep tersebut hampir sama dengan skema yang dirancang dan sudah dijelaskan pada sub bab 4.3 Gambar 4.5.

Sebagai contoh pada skema DCT-CRT dan TLDCT-CRT, jika citra berukuran 512x512 maka jumlah blok pada citra tersebut bernilai 64x64 atau sama dengan 4096 blok, sedangkan masing-masing blok memiliki piksel 8x8 piksel atau totalnya 64 piksel. Seperti dijelaskan diatas, blok akan diacak dengan nilai random antara 0-4095, setelah itu piksel pada tiap blok akan diacak juga dengan nilai 0-63. Sedangkan pada metode DWT-CRT yang tidak berbasis blok, tidak melibatkan proses pengacakan blok, dan langsung dilakukan proses pengacakan piksel, sehingga nilai random piksel antara 0-262143 atau berjumlah total piksel citra 512x512 tersebut. Hal inilah yang menyebabkan proses *embedding* menjadi sangat lama, karena waktu yang dibutuhkan untuk proses random dan pengecekan memiliki kombinasi yang sangat besar dibandingkan dengan 2 metode penelitian sebelumnya.

Untuk menguatkan penjelasan diatas maka dapat dilakukan dengan mencari nilai kompleksitas algoritma pada skema yang dirancang. Jika melihat alur proses *embedding*

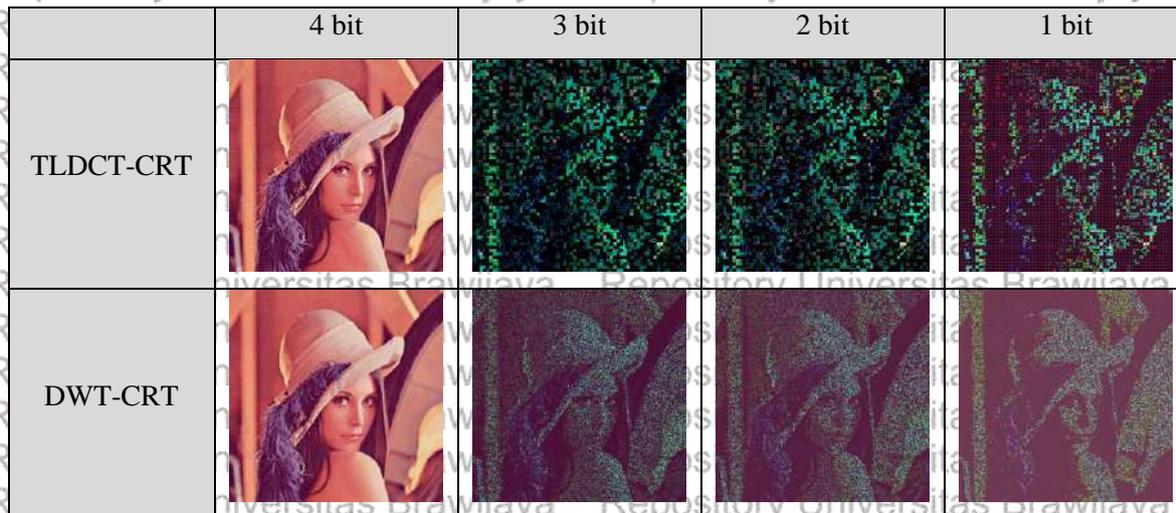
pada Gambar 4.5 halaman 33. Apabila piksel citra sebagai input dinotasikan dengan  $N$  dan ukuran bit dengan notasi  $B$ , serta proses diulang sebanyak 3 kali berdasarkan *channel* warna, maka berdasarkan alur tersebut, pada pengulangan pertama menggunakan *While* sebagai proses pengecekan agar tidak terdapat nilai random yang sama. Pada proses tersebut untuk *best case* memiliki kompleksitas  $N^2$ , sedangkan untuk *worst case* kompleksitasnya adalah  $N^3$ . Kemudian untuk pengulangan ke dua melibatkan panjang *watermark* dikali *bit size*, sehingga kompleksitasnya adalah  $\frac{1}{4}xNx B$ . Karena proses dilakukan sebanyak 3 kali, maka total kompleksitas proses *embedding* untuk *best case* adalah  $3(N^2 \times \frac{1}{4} NB) \in O(N^2)$ . Sedangkan untuk *worst case*  $3(N^3 \times \frac{1}{4} NB) \in O(N^3)$ .

Jika kembali pada pembahasan bit *watermark* yang disisipkan, seluruh pengujian skema DWT-CRT diatas melibatkan 4 bit kuantisasi. Pemilihan nilai tersebut berdasarkan salah satu tujuan dari penelitian ini yakni memperbaiki hasil proses ekstraksi. Seperti yang sudah diketahui bahwa semakin besar bit kuantisasi maka semakin banyak bit *watermark* yang disisipkan sehingga semakin baik hasil ekstraksi, namun mengakibatkan semakin menurun hasil proses *embedding*. Untuk melihat efek dari perubahan bit kuantisasi maka dilakukan pengujian dengan mengganti bit kuantisasi pada proses *generate*, kemudian dilanjutkan dengan melakukan proses *embedding* serta proses ekstraksi. Hasil pengujian tersebut dapat dilihat pada Tabel 5.5.

Tabel 5.5 Perbandingan efek bit kuantisasi terhadap PSNR embedding dan ekstraksi

Nama Citra	Bit Kuantisasi	PSNR Embedding (dB)		PSNR Ekstraksi (dB)	
		TLDCT-CRT	DWT-CRT	TLDCT-CRT	DWT-CRT
Lenna	4	47,434	47,545	26,905	28,838
Lenna	3	58,092	58,376	5,803	9,893
Lenna	2	59,832	60,316	5,961	10,618
Lenna	1	61,574	63,270	7,938	12,055

Dari Tabel 5.5 dapat diketahui bahwa pemilihan nilai 4 bit kuantisasi terlihat sangat ideal dibandingkan nilai bit kuantisasi yang lain karena PSNR proses *embedding* dan ekstraksi memiliki kombinasi nilai yang terbaik. Dalam tabel tersebut juga terlihat bahwa hasil dari skema DWT-CRT masih lebih baik dibandingkan penelitian sebelumnya yang menggunakan skema TLDCT-CRT. Untuk lebih memperjelas hasil pengujian diatas, berikut adalah citra ekstraksi hasil pengujian diatas yang ditunjukkan dalam Tabel 5.6.



Gambar 5.6 Citra ekstraksi hasil pengujian perbandingan bit kuantisasi

### 5.3 Ekstraksi Watermark

Sesuai dengan penjabaran pada sub bab 4.4, proses ekstraksi dilakukan dengan terlebih dahulu memanggil file kunci yang berisi data nilai maksimum subband, nilai minimum subband, nilai median subband, *key* PRNG, dan bit *watermark* hasil proses *generate*.

Dimulai dengan mengecek apakah citra penggunaan data *key* PRNG, berfungsi untuk memperoleh kembali posisi bit *watermark* yang disebar secara acak pada citra hasil *embedding*. Apabila posisi *watermark* sudah didapatkan maka akan diproses ekstraksi sesuai aturan CRT.

Hasil dari proses ekstraksi adalah bit *watermark* yang sebelumnya disisipkan. Bit watermark tersebut akan rekonstruksi menjadi citra kembali dengan melakukan proses inverse kuantisasi serta inverse metode DWT. Hasil pengujian dari proses ekstraksi ditunjukkan dalam Tabel 5.6.

Tabel 5.6 Hasil pengujian proses ekstraksi *watermark*

No.	Nama Citra	Ukuran Bit	PSNR (dB)		NC	
			DWT-CRT	DWT-CRT Blok	DWT-CRT	DWT-CRT Blok
1	Airplane	4 bit	28,014	27,422	0,998	0,998
2	Baboon	4 bit	21,519	21,355	0,972	0,947
3	Lenna	4 bit	29,234	28,497	0,996	0,996
4	Pepper	4 bit	28,334	27,464	0,994	0,994
5	Barbara	4 bit	25,584	25,171	0,988	0,989
6	Boats	4 bit	28,041	27,236	0,982	0,988
7	Cable Car	4 bit	27,858	27,083	0,995	0,990
8	Gold Hill	4 bit	30,148	29,053	0,995	0,995
9	Monarch	4 bit	28,578	27,851	0,992	0,993
10	Sail Boat	4 bit	26,580	26,075	0,993	0,993

No.	Nama Citra	Ukuran Bit	PSNR (dB)		NC	
			DWT-CRT	DWT-CRT Blok	DWT-CRT	DWT-CRT Blok
11	Zelda	4 bit	31,671	29,991	0,998	0,994
12	Cek Bank CIMB	4 bit	24,981	22,253	0,996	0,997
13	Cek Bank Danamon	4 bit	24,112	23,832	0,995	0,994
14	Cek Bank BCA	4 bit	21,991	21,266	0,991	0,967
15	Kartu Kredit	4 bit	30,148	21,629	0,942	0,925
16	KTP	4 bit	21,982	23,372	0,992	0,973
17	Paspor	4 bit	23,742	23,621	0,993	0,975
18	Sertifikat Harvard	4 bit	22,310	22,083	0,993	0,989
19	Sertifikat PAP	4 bit	23,731	22,006	0,994	0,986
20	Sertifikat TPA	4 bit	21,981	21,429	0,985	0,974
21	Boardwalk	4 bit	24,166	24,166	0,989	0,989

Pada Tabel 5.6 dapat dilihat bahwa hasil proses ekstraksi memiliki nilai PSNR dan NC yang cukup baik, artinya citra ekstraksi memiliki kemiripan yang cukup baik jika dibandingkan dengan citra awal. Skema DWT-CRT memiliki rata-rata nilai PSNR 25,617 dB dan rata-rata nilai NC sebesar 0,989, sedangkan pada skema DWT-CRT yang menggunakan basis blok 8x8 memiliki rata-rata PSNR 24,897 dB serta rata-rata NC sebesar 0,983. Dari hasil proses ekstraksi dapat disimpulkan bahwa skema DWT-CRT masih lebih baik dibandingkan dengan skema DWT-CRT yang berbasis blok 8x8.

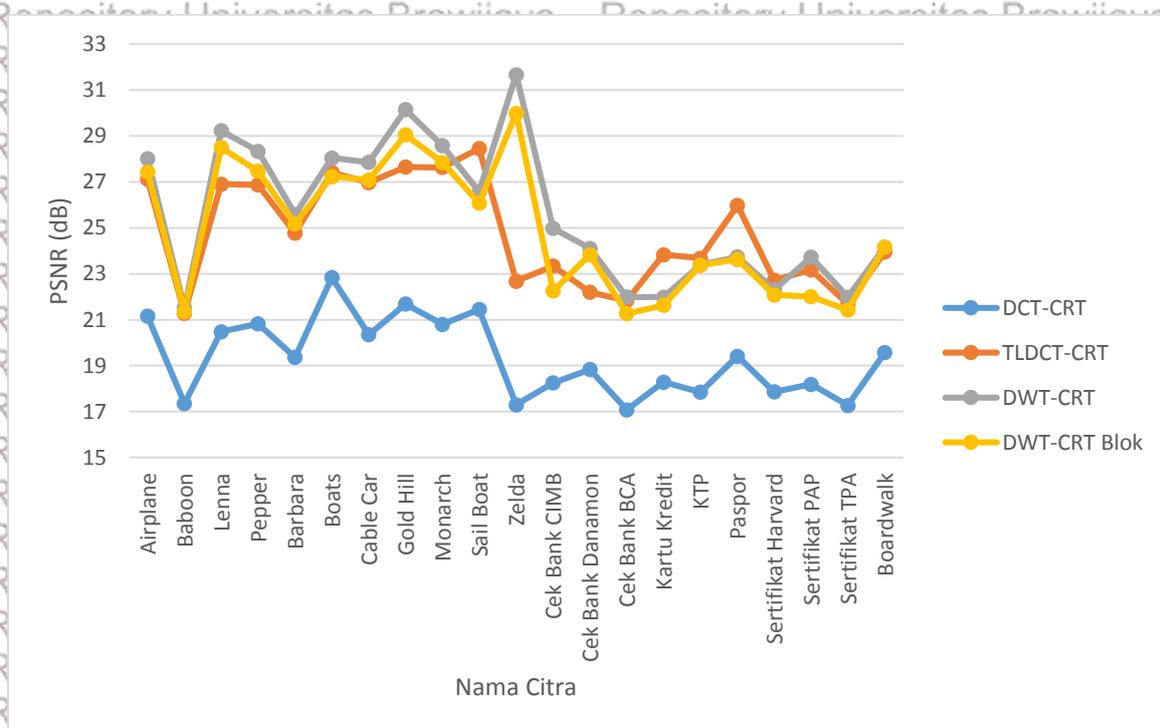
Untuk melihat performa ekstraksi pada skema yang dirancang maka hasil ekstraksi diatas juga akan dibandingkan dengan hasil ekstraksi pada metode penelitian sebelumnya. Perbandingan tersebut dapat dilihat pada Tabel 5.7.

Tabel 5.7 Perbandingan nilai PSNR ekstraksi dengan skema penelitian terdahulu

No.	Nama Citra	DCT-CRT (dB)	TLDCT-CRT (dB)	DWT-CRT (dB)	DWT-CRT Blok (dB)
1	Airplane	21,157	27,116	28,014	27,422
2	Baboon	17,358	21,284	21,519	21,355
3	Lenna	20,482	26,905	29,234	28,497
4	Pepper	20,823	26,871	28,334	27,464
5	Barbara	19,361	24,773	25,584	25,171
6	Boats	22,839	27,414	28,041	27,236
7	Cable Car	20,352	26,975	27,858	27,083
8	Gold Hill	21,683	27,645	30,148	29,053
9	Monarch	20,801	27,636	28,578	27,851
10	Sail Boat	21,447	28,451	26,580	26,075
11	Zelda	17,303	22,675	31,671	29,991
12	Cek Bank CIMB	18,251	23,338	24,981	22,253
13	Cek Bank Danamon	18,834	22,201	24,112	23,832
14	Cek Bank BCA	17,076	21,838	21,991	21,266
15	Kartu Kredit	18,289	23,830	30,148	21,629

No.	Nama Citra	DCT-CRT (dB)	TLDCT-CRT (dB)	DWT-CRT (dB)	DWT-CRT Blok (dB)
16	KTP	17,846	23,679	21,982	23,372
17	Paspor	19,411	25,977	23,742	23,621
18	Sertifikat Harvard	17,861	22,717	22,310	22,083
19	Sertifikat PAP	18,190	23,174	23,731	22,006
20	Sertifikat TPA	17,273	21,679	21,981	21,429
21	Boardwalk	19,574	23,953	24,166	24,166

Hasil perbandingan nilai PSNR pada Tabel 5.7, tampak bahwa nilai PSNR pada skema yang dirancang sedikit lebih baik dibandingkan dengan hasil ekstraksi metode penelitian sebelumnya. Untuk lebih memperjelas perbedaan antara ketiga skema diatas, dapat dilihat pada Gambar 5.7.



Gambar 5.7 Grafik perbandingan nilai PSNR ekstraksi dengan penelitian sebelumnya

Jika melihat grafik pada Gambar 5.7, skema DCT-CRT memiliki nilai PSNR yang paling rendah, hal ini dikarenakan jumlah bit *watermark* yang disisipkan paling sedikit dibandingkan dengan tiga skema lainnya. Sedangkan untuk skema DWT-CRT, DWT-CRT blok dan TLDCT-CRT, meskipun jumlah bit *watermark* yang disisipkan sama akan tetapi hasil PSNR ekstraksi masih lebih baik jika menggunakan metode DWT-CRT, bahkan pada skema DWT-CRT blok juga masih lebih baik daripada skema TLDCT-CRT. Sehingga bisa disimpulkan bahwa penggunaan algoritma DWT menghasilkan nilai ekstraksi yang lebih baik daripada algoritma TLDCT.

Untuk waktu proses ekstraksi, hasilnya meningkat dari waktu yang dibutuhkan pada proses *embedding*. Hasil pengujian dan perbandingan waktu proses ekstraksi 4 skema diatas dapat dilihat dalam Tabel 5.8.

Tabel 5.8 Perbandingan waktu proses ekstraksi (dalam ms)

No.	Nama Citra	Ukuran (px)	DCT-CRT	TLDCT-CRT	DWT-CRT	DWT-CRT Blok
1	Baboon	500x480	703	684	398392	673
2	Cable Car	512x480	738	723	402847	759
3	Airplane	512x512	741	731	434574	739
4	Lenna	512x512	891	852	414837	847
5	Pepper	512x512	826	835	448438	860
6	Sail Boat	512x512	849	820	453483	842
7	Kartu Kredit	634x403	893	874	411834	881
8	Barbara	720x576	963	952	683483	983
9	Goldhill	720x576	915	937	664354	945
10	Sertifikat Harvard	756x580	964	982	713884	953
11	Monarch	768x512	913	892	649237	883
12	Boats	787x576	973	949	732347	984
13	Zelda	787x576	1079	1037	764583	1048
14	Cek Bank CIMB	800x406	905	862	583487	893
15	Paspor	800x561	964	910	762347	934
16	KTP	984x641	1485	1328	1012383	1436
17	Cek Bank BCA	1050x408	1058	1037	702328	1083
18	Sertifikat PAP	1200x654	2031	1974	1528297	2018
19	Sertifikat TPA	1293x526	1948	1825	1138479	1983
20	Cek Bank Danamon	1400x544	2683	2482	1401849	2593
21	Boardwalk	2000x1335	26481	25271	17924273	26384

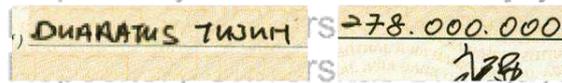
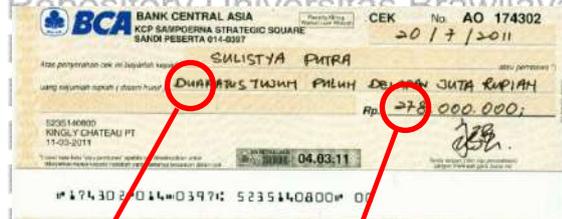
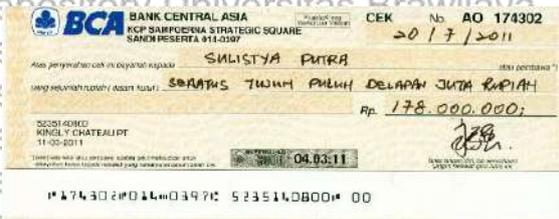
Dari Tabel 5.8 dapat dilihat bahwa terdapat peningkatan waktu proses pada semua skema, hal ini dikarenakan pada proses ekstraksi memiliki jumlah *step* yang lebih banyak daripada proses *embedding*, seperti pengambilan data dari file kunci, maupun proses penyusunan bit *watermark* menjadi matriks 2 dimensi, dan terakhir adalah mengubah matriks tersebut menjadi sebuah citra ekstraksi.

#### 5.4 Deteksi dan Pemulihan Citra

Pengujian terakhir adalah pengujian daya tahan skema yang dirancang terhadap serangan. Dalam pengujian ini citra hasil proses *embedding* akan diberikan serangan berupa pengubahan pada sebagian area citra (serangan *tampering*). Serangan *tampering* pada citra ditunjukkan dalam Gambar 5.8.

Citra ter-watermark

Proses tampering



Gambar 5.8 Proses tampering pada citra ter-watermark

Proses deteksi dilakukan dengan membandingkan citra pembanding, yang merupakan citra hasil rekonstruksi dari bit watermark proses generate yang disimpan dalam file kunci, kemudian dibandingkan dengan citra ekstraksi, jika nilai *error* melebihi batas yang ditetapkan maka citra tersebut dianggap sudah terindikasi terdapat proses penyuntingan, dan proses pemulihan akan dilakukan.

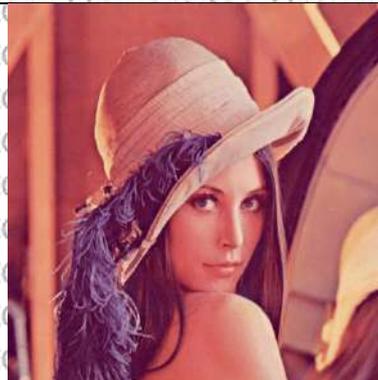
Proses pemulihan melibatkan citra ter-watermark dengan citra pembanding, intensitas warna pada lokasi piksel yang terindikasi terdapat penyuntingan yang sudah didapat dari proses perhitungan nilai *error* sebelumnya, akan diganti dengan nilai piksel citra pembanding, sebaliknya piksel yang tidak terdeteksi adanya penyuntingan nilai intensitas warnanya menggunakan nilai piksel citra ter-watermark. Proses ini sudah dijelaskan dalam sub bab 4.7. Hasil dari proses pemulihan citra pada kasus diatas dapat dilihat pada Gambar 5.9.

Citra Ekstraksi



PSNR = 26.9 dB; NC = 0.999

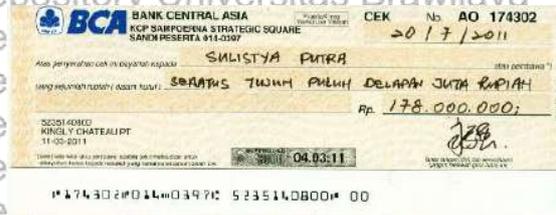
Citra Pemulihan



PSNR = 43.323 dB



PSNR = 21.336 dB; NC = 0.985



PSNR = 30.078 dB

Gambar 5.9 Citra hasil proses deteksi dan pemulihan

Sama dengan pengujian sebelumnya tentang pengujian daya tahan, kali ini skema yang dirancang akan diberikan serangan kompresi JPEG sebesar 10% rasio kompresi. Hasil dari serangan kompresi tersebut dapat dilihat dalam Gambar 5.10.

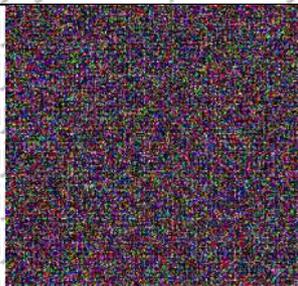
Citra ter-watermark

Citra ekstraksi

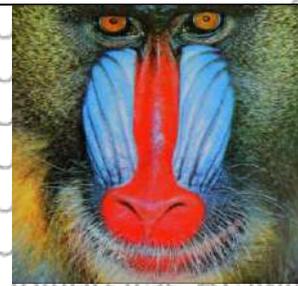
Citra Pemulihan



Kompresi JPEG 10%



PSNR = 8,726 dB



PSNR = 23,538 dB



Kompresi JPEG 10%



PSNR = 7,496 dB



PSNR = 30,137 dB

Gambar 5.10 Hasil proses pemulihan terhadap serangan kompresi JPEG 10%





Karena metode CRT adalah jenis metode watermarking fragile maka hasil pada Gambar 5.10 menunjukkan citra ekstraksi tidak dapat dikembalikan seperti citra watermark. Proses penyisipan pada metode CRT langsung diaplikasikan pada nilai intensitas piksel, sehingga apabila dilakukan kompresi JPEG maka seluruh nilai intensitas piksel akan berubah dan menyebabkan hilangnya informasi bit watermark yang ditanam.

Ketika terjadi perubahan secara menyeluruh pada intensitas warna citra dan skema mendeteksi bahwa terjadi serangan pada semua piksel, maka secara logika citra pemulihan merupakan citra yang didapat dari bit watermark yang disimpan dalam file kunci.