

# Uji Ketahanan Metode Block-base Watermarking pada Domain Wavelet Terhadap Serangan Gaussian Blur dan Random Noise

Ekasari Nugraheni  
P2 Informatika – LIPI  
Jl. Cisit, Sangkuriang, Bandung 40135  
iin@informatika.lipi.go.id

Nova Hadi Lestriandoko  
P2 Informatika - LIPI  
Jl. Cisit, Sangkuriang, Bandung 40135  
ryan@informatika.lipi.go.id

## Abstract

*This paper examines the robustness of the block-based watermarking using wavelet transform coefficient against Gaussian blur attack and random noise attack for copyright protection. It uses block of wavelet transform coefficient with size  $3 \times 3$  as embedding media. The load will be inserted into block at random location. The steps of inverse watermarking are similar with the watermark embedding. The difference is only in the step after wavelet calculation, it extracts the binary code from coefficient blocks. This watermarking scheme does not require the original image for watermark extraction (blind watermarking). The PSNR (Peak Signal Noise Ratio) and HVS (Human Visual System) is used to measure the robustness. Experimental results will be displayed to indicate the algorithm robustness from attacks.*

*Keywords: wavelet coefficient, block-based watermarking, robustness, Gaussian Blur, random noise*

## Abstrak

*Makalah ini mengkaji tentang ketahanan metode block-based watermarking pada domain wavelet terhadap gangguan Gaussian blur dan derau acak (random noise) untuk perlindungan hak cipta gambar digital. Metode ini menggunakan blok koefisien transformasi wavelet berukuran  $3 \times 3$  sebagai media penanaman watermark. Muatan akan dimasukkan ke dalam blok koefisien wavelet pada lokasi yang acak. Pembacaan kembali informasi yang ditanamkan pada proses watermarking serupa dengan langkah-langkah penanaman. Perbedaannya hanya pada langkah setelah perhitungan koefisien wavelet, yaitu ekstraksi kode biner. Selain itu, skema watermarking ini tidak memerlukan citra asli untuk proses ekstraksi watermark (blind watermarking). Sebagai tolak ukur ketahanannya, digunakan nilai PSNR (Peak Signal Noise Ratio) dan HVS (Human Visual System). Hasil-hasil eksperimen akan didiskusikan untuk mengetahui kekokohan algoritma terhadap serangan.*

*Kata kunci : koefisien wavelet, block-based watermarking, ketahanan, Gaussian blur, derau acak*

## 1. Pendahuluan

Citra digital seperti data digital lainnya memiliki karakteristik mudah digandakan dan didistribusikan. Penggunaan internet sebagai platform untuk distribusi data digital telah berkembang dan menyebabkan kekhawatiran tentang penyalahgunaan manipulasi konten digital khususnya citra digital yang perubahannya pun tidak mudah diketahui oleh beberapa sistem visual manusia[1].

Melalui internet, pihak lain dapat dengan mudah mendapatkan salinan dan melakukan manipulasi pada citra digital. Masalah akan muncul jika citra digital tersebut merupakan hasil karya yang harus dilindungi, seperti gambar fotografi, penginderaan jauh, rekam medis, dan gambar-gambar lainnya yang bersifat privat dan sensitif. Pelanggaran copyright, masalah pemalsuan dokumen gambar, dan kepemilikan (ownership)[2] merupakan masalah yang sering terjadi sampai saat ini. Masalah perlindungan hak cipta gambar digital berkaitan erat dengan menjaga originalitas dan semangat kreatifitas pembuat akan hasil karyanya.

Digital image watermarking merupakan suatu cara untuk menyisipkan atau menyembunyikan data/informasi tertentu ke dalam media citra digital tanpa diketahui kehadirannya oleh indera manusia dan mampu menghadapi proses-proses pengolahan sinyal digital sampai pada tahap tertentu [3]. Data informasi ini kemudian dapat diekstrak kembali sebagai pernyataan untuk membuktikan otentikasi atau bukti kepemilikan citra digital. Informasi yang dimasukkan dapat berupa teks, logo, data audio, maupun rangkaian bit yang tidak bermakna.

Metode penyisipan watermark dapat dilakukan dalam dua domain, yaitu domain spasial dan domain transform/frekuensi [4]. Metode spasial akan menyisipkan watermark secara langsung ke dalam piksel citra. Cara ini lebih cepat tetapi biasanya watermark menjadi rapuh saat gambar dimanipulasi. Least Significant Bit (LSB) dan Most Significant Bit (MSB) adalah contoh untuk metode watermarking dalam domain spasial. Metode yang lebih kokoh untuk watermarking dapat ditingkatkan dalam domain transform. Penyisipan dengan metode transform berarti watermark disisipkan ke dalam koefisien transformasi. Metode watermarking

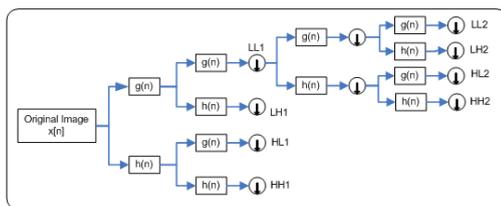
dalam transformasi domain yang umum digunakan adalah *Discrete Fourier Transform* (DFT), *Discrete Cosine Transform* (DCT), dan *Discrete Wavelet Transform* (DWT). Cara ini lebih kompleks jika dibandingkan dengan metode spasial, tetapi *watermark* akan lebih kokoh saat terjadi serangan pada citra. Serangan ini dapat berupa operasi geometri seperti proses rotasi, translasi; atau operasi non-geometri seperti pemberian derau (*noise*), pengaburan citra (*blurring*) atau kompresi JPEG.

Dalam tulisan ini, kami mengusulkan metode blok-base watermarking dalam domain wavelet transform. Sebuah bit *watermark* akan disisipkan ke dalam koefisien wavelet yang dikelompokkan ke dalam blok. *Watermark* diperoleh dari blok LL koefisien wavelet. Ekstraksi *watermark* menggunakan metode *blind watermark*, dimana pada proses ekstraksi hanya membutuhkan kunci atau parameter-parameter dan tidak memerlukan citra asli. Tulisan ini juga akan membahas seberapa jauh ketahanan metode *watermarking* yang telah dikembangkan terhadap serangan *gaussian blur* dan *noise* berdasarkan keutuhan *watermark* setelah diserang.

## 2. Discrete Wavelet Transform

Transformasi wavelet diskrit (DWT) umumnya digunakan untuk tujuan *watermarking* dan kompresi. Sinyal asli dilambangkan dengan  $x[n]$  terbagi menjadi dua bagian, *lowpass filter* dinotasikan dengan  $g[n]$  dan *highpass filter* dilambangkan dengan  $h[n]$ . Pemilahan ini disebut sebagai dekomposisi. *Highpass filter* digunakan untuk menganalisis frekuensi tinggi dan *lowpass filter* digunakan untuk menganalisis frekuensi rendah. Komponen mutakhir sinyal sebagian besar terbatas pada bagian frekuensi tinggi. Bagian frekuensi rendah dibagi lagi menjadi dua bagian frekuensi tinggi dan rendah[5].

Gambar 1 menunjukkan dekomposisi yang diwakili oleh pohon biner dengan simpul mewakili ruang sub band dengan lokalisasi frekuensi waktu yang berbeda.



Gambar 1. Dekomposisi DWT

Pada setiap tingkatan dalam diagram di atas, sinyal diuraikan menjadi frekuensi rendah dan tinggi. Karena proses dekomposisi sinyal masukan harus kelipatan  $2^n$ , dimana  $n$  adalah jumlah tingkat[6]. Secara matematis dekomposisi tingkat satu dapat dinyatakan sebagai berikut :

$$y_{high}[k] = \sum_n x[k]h[2k-n] \dots\dots(1)$$

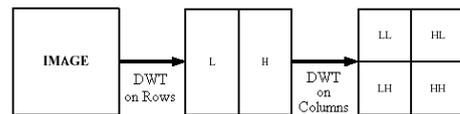
$$y_{low}[k] = \sum_n x[k]g[2k-n] \dots\dots(2)$$

dimana  $y_{high}[k]$  dan  $y_{low}[k]$  adalah hasil dari *highpass* dan *lowpass filters* setelah dipilah menjadi dua. Proses ini dapat diulang untuk dekomposisi lebih lanjut[7].

Hasil dari *highpass* dan *lowpass filters* disebut dengan koefisien DWT, dan dengan menggunakan koefisien DWT gambar asli dapat direkonstruksi ulang. Proses rekonstruksi ini disebut dengan *inverse DWT* (IDWT). Formula rekonstruksi (untuk setiap tingkat) bisa dinyatakan sebagai berikut :

$$x[n] = \sum_n (y_{high}[k]h[2k-n] + y_{low}[k]g[2k-n])$$

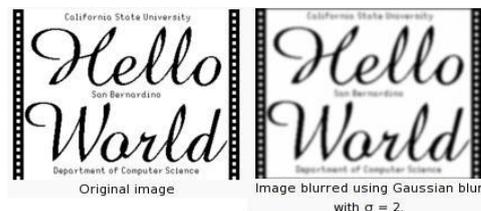
Sebuah citra juga dapat didekomposisi dalam struktur piramida seperti diperlihatkan pada gambar 2. Setiap tingkat dari dekomposisi DWT akan menghasilkan empat subband yaitu LL (*low-low frequency*) menyerupai dengan citra asli, serta tiga subband lain yang menyimpan sebagian besar informasi dari citra asli, yaitu LH (*low-high frequency*), HL (*high-low frequency*) dan HH (*high-high frequency*)[7].



Gambar 2. Dekomposisi piramida DWT pada sebuah citra

## 3. Gaussian Blur dan Random Noise

*Gaussian blur* (biasa dikenal sebagai *Gaussian smoothing*) adalah efek yang banyak digunakan dalam perangkat lunak untuk mengurangi derau (*noise*), penghalusan (*smoothing*), pengaburan (*blur*) dan mengurangi detail pada gambar digital. Efek visual dari teknik ini adalah *smooth blur* yang menyerupai melihat gambar melalui layar transparan. [8] Secara kasat mata, gangguan ini mudah dilihat oleh mata, karena tampak berbeda dengan piksel aslinya.



Gambar 3. (kiri) Gambar asli. (kanan) Gambar dengan efek *Gaussian Blur*

Pengaburan citra (*blur*) didapatkan dengan mengkonvolusikan citra dengan sebuah penapis (filter). Penapis ini disebut juga penapis lolos-rendah (*low-pass filter*), karena menekan komponen yang berfrekuensi tinggi dan meloloskan yang berfrekuensi rendah. Intensitas blur dapat diatur dengan mengubah nilai radius terhadap sumbu vertikal atau horisontal. Nilai radius yang semakin tinggi akan menghasilkan citra yang lebih kabur.

*Random noise* ditandai dengan peningkatan intensitas gambar yang terjadi melalui perubahan intensitas warna. Perubahan terjadi secara acak dan tidak dapat diprediksi lokasinya. Persentase piksel yang akan dikenakan noise ditentukan dengan nilai persentase randomisasi. Beberapa tipe *random noise* diantaranya adalah *random hurl*, *random slur* dan *random pick*.

*Random hurl* mengubah warna secara acak pada piksel yang terkena sehingga menghasilkan gangguan acak (*random noise*) yang nyata terlihat. Semua jalur (*channel*) warna, termasuk jalur alpha (jika ada) akan diacak. Gangguan dikenakan pada semua atau beberapa piksel pada lapisan (*layer*) yang aktif[9].

*Random Slur* menghasilkan efek seperti gambar yang mencair ke bawah dengan peluang 80% piksel tersebut diganti oleh nilai piksel langsung di atasnya atau salah satu dari piksel di kanan atau di kirinya[10].

*Random Pick* menggantikan setiap piksel yang terkena dengan nilai piksel yang dipilih secara acak dari delapan tetangganya dan dirinya sendiri (blok 3x3 piksel)[11].



Citra Asli                      Random Pick Noise

**Gambar 4. (kiri) Gambar asli. (kanan) Gambar dengan *Random Pick Noise***

HVS noise menghasilkan gangguan pada lapisan (*layer*) aktif dengan menggunakan model warna *Hue*, *Saturation* dan nilai cahaya (*luminosity*)

#### 4. Peak Signal To Noise Ratio (PSNR)

PSNR digunakan untuk menghitung rasio puncak sinyal terhadap gangguan antara dua citra dalam satuan unit *decibels*(dB). PSNR juga digunakan untuk mengukur kualitas citra original dan citra terkompresi [12]. Semakin tinggi PSNR, semakin bagus kualitas citra.

$$PSNR = 10 * \log_{10} \left( \frac{Max_I^2}{MSE} \right) \quad (4)$$

$$MSE = \frac{1}{m * n} \sum_{i=1}^m \sum_{j=1}^n [I(i, j) - I'(i, j)]^2 \quad (5)$$

dimana ,

$m, n$  : lebar dan tinggi citra

$I(i, j)$  : nilai piksel dari citra asli pada koordinat (i,j),

$I'(i, j)$  : nilai piksel dari citra terkompresi pada koordinat (i,j)

$Max_I$  : 255 (nilai maksimum dari piksel)

Dari persamaan diketahui, PSNR berbanding terbalik dengan MSE. Nilai MSE yang rendah akan menghasilkan PSNR yang tinggi. Hal ini menunjukkan bahwa citra hasil kompresi tidak jauh berbeda dengan citra semula. Semakin tinggi nilai PSNR-nya, maka kualitas citra akan semakin bagus, begitu juga sebaliknya

#### 5. Metode Block-base watermarking

Konsep dari *block-based watermarking* dalam domain wavelet adalah menempatkan sebuah *watermark* pada blok dari koefisien *wavelet*. Untuk setiap 8 bit *payload* akan disisipkan ke blok berukuran 3x3 piksel. Untuk setiap koefisien dalam wilayah LL mewakili 4 piksel yang berukuran 2x2, sebuah piksel akan dipilih secara acak untuk ditanamkan *watermark*[13].

##### 5.1. Inisialisasi citra

Pada proses inisialisasi, *watermark* yang akan disisipkan ke dalam citra asli harus memenuhi syarat tertentu. Diasumsikan bahwa citra asli (citra A) berukuran ( $h1 \times w1$ ) piksel dan *watermark* yang akan disisipkan (citra B) berukuran ( $h2 \times w2$ ) piksel. Maka proses *wavelet* akan berlaku jika syarat:

$$h2 \leq \frac{h1}{(2 \times 3)} \quad \text{dan} \quad w2 \leq \frac{w1}{(2 \times 3)} \quad (6)$$

terpenuhi.

##### 5.2. Penanaman Watermark

Citra A sebagai *host* yang akan disisipkan *watermark* akan dipilah menggunakan DWT dan menghasilkan empat subband LL (rata-rata elemen), LH (perbedaan antar kolom), HL (perbedaan antar baris) dan HH (perbedaan antar diagonal). Untuk pasangan nilai piksel (x,y) dari *watermark* citra B dengan RGB (24 bits) berlaku :

$$0 \leq x_{RGB} < \frac{w1}{2} \quad \text{dan} \quad 0 \leq y_{RGB} < \frac{h1}{2}$$

*Signature* data digital didapat dari *watermark* citra B. Setiap nilai piksel dari *watermark* citra B harus dikonversi ke *biner* terlebih dahulu untuk menghasilkan 8 bit per piksel. Kemudian ditambahkan 1 bit 0 disetiap awal piksel, sehingga nantinya setiap piksel akan memiliki 9 bit. Data *biner* 9 bit per piksel ini disebut dengan *payload*.

Koefisien pada subband LL akan dipilih sebagai frekuensi untuk penanaman *watermark*. Berdasarkan koefisien DWT, proses rekonstruksi dari piksel asli ( $x1, x2, y1, y2$ ) akan menghasilkan nilai sisa dari LL/4 (0, 0.25, 0.50, 0.75), dimana LL adalah nilai rata-rata dari elemen piksel. Jika nilai sisa LL adalah xx.00 atau xx.50 maka akan dibaca "0" (nol). Jika nilai sisa adalah xx.25 atau xx.75 maka akan dibaca "1" (satu).

### 5.3. Pembacaan Watermark

Pembacaan kembali informasi dari *watermarked image* dapat dilakukan sama dengan metode penanaman *watermark*. Perbedaannya hanya setelah perhitungan koefisien *wavelet*, maka dilakukan pembacaan terhadap kode biner yang disisipkan. Dari kode yang didapatkan, akan direkonstruksi pesan/gambar *watermark*. Apabila dari hasil ekstraksi tersebut *watermark* dapat dikenali, maka dapat dijadikan sebagai pembuktian kepemilikan citra digital.

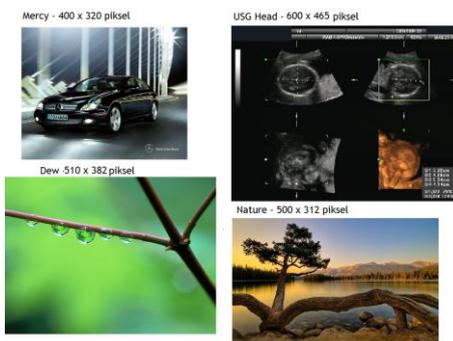
## 6. Hasil Pengujian

Tiga layer RGB untuk setiap bit piksel akan digunakan, sehingga ukuran data payload menjadi  $8 \times 3 \times$  resolusi citra.



Gambar 5. *watermark*- Logo LIPI

*Watermark* yang akan digunakan adalah logo LIPI berukuran  $45 \times 50$  piksel (Gambar 5) dengan data payload sebesar  $8 \times 3 \times 2.250 = 54.000$  bits.



Gambar 6. Citra Asli

Berdasarkan persamaan 6, ukuran minimal citra asli yang dapat disisipkan *watermark* adalah  $270 \times 300$  piksel.

Citra asli yang digunakan untuk pengujian adalah file “png” dalam format RGB dengan ukuran bervariasi seperti diperlihatkan pada gambar 6. Hasil dari metode *watermark* ini diperlihatkan pada Tabel 1.

Tabel 1. Nilai PSNR pada *watermarked image*

Filename	Size (pixel)	Sum of Diff	PSNR (dB)
Mercy	400x320	30.355	59.1518
USG-Head	600 x 456	33.035	62.1683
Nature	500x312	30.123	60.0442
Dew	510x382	30.598	60.9414

Gambar 7 memperlihatkan citra yang telah ditanamkan *watermark* dan *watermark* hasil ekstraksi dengan nilai PSNR 59.1518, *watermark* hasil ekstraksi terlihat dengan jelas.



Gambar 7. *watermarked Image* dan *watermark* hasil ekstraksi

### 6.1. Serangan Random Noise dan Gaussian Blur

*Watermarked Image* yang digunakan pada pengujian serangan ini adalah citra 1 (Mercy.png) dengan ukuran  $400 \times 320$  piksel dan citra 2 (USG-Head.png) dengan ukuran  $600 \times 456$  piksel.

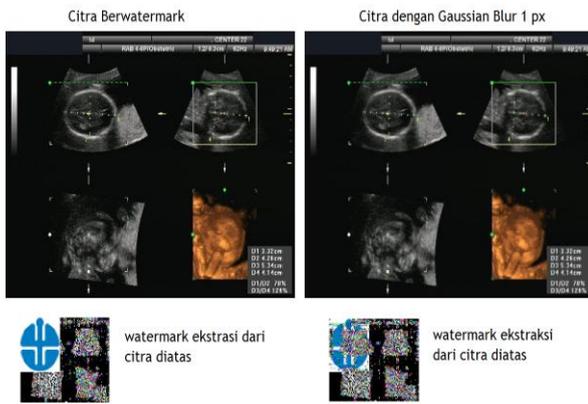
Serangan pada *watermarked image* menggunakan perangkat pengolah citra Gimp 2.6.11. *Noise* dikenakan pada *watermarked image* dengan beberapa tipe variasi, diantaranya *random slur*, *random hurl*, *random pick* dan *HVS noise* dengan intensitas yang beragam. *Gaussian blur* dikenakan pada *watermarked image* dengan radius 1 px. Kemudian *watermarked image* yang telah diserang tersebut diekstrak kembali sesuai dengan domain penanamannya sehingga diperoleh *watermark* terekstrak hasil serangan.

Gambar 8 berikut menunjukkan hasil simulasi dari *watermarked image* (MercyWatermarked.png) yang telah dikenakan gangguan (*noise*) *Random Slur* dengan intensitas 20%. *Watermark* hasil ekstraksi menunjukkan bahwa *watermark* masih bisa dikenali walaupun terjadi kerusakan.



Gambar 8. *Watermarked image* yang diberi gangguan (*noise*) dan *watermark* hasil ekstraksi

Simulasi dari *watermarked image* (USG-HeadWatermarked.png) yang dikenakan efek *Gaussian Blur* pada radius 1 px pada sumbu vertikal dan horisontal ditunjukkan pada gambar 9. Pada bagian bawah diperlihatkan *watermark* hasil ekstraksi dari *watermarked image* yang belum dan yang sudah diberi efek *Gaussian blur*.



**Gambar 9. Watermarked image dan watermark hasil ekstraksi dengan Gaussian Blur**

*Watermark* hasil ekstraksi dari citra yang telah terserang masih bisa dikenali walaupun mendapatkan banyak gangguan.

Terdapat tiga tingkatan untuk kualitas citra *watermark* yaitu : baik, sedang, buruk. Pada tingkat baik, citra terlihat jelas dan bersih. Pada tingkat sedang, citra masih bisa terlihat jelas walaupun telah terjadi kerusakan. Pada tingkat buruk, walaupun mendapatkan banyak gangguan, citra masih bisa dikenali. Secara lengkap beberapa pengujian yang telah dilakukan diperlihatkan pada Tabel 2.

**Tabel 2. Serangan pada *watermarked image***

No	Tipe Serangan	Citra 1	Citra 2
Gaussian Blur			
1	Radius 1 pix	n/a	 Buruk
2	Radius 2 pix	n/a	n/a
Noising			
1	HVS Noise	n/a	n/a
2	Random Slur 5%	 Baik	 Baik
3	Random Hurl 5%	 Baik	 Baik
4	Random Slur 20%	 Sedang	 Sedang
5	Random Hurl 20%	 Sedang	 Sedang
6	Random Pick 20%	 Sedang	 Sedang
7	Random Hurl 25%	 Buruk	 Buruk

Berdasarkan Tabel 2, metode ini bertahan pada hampir semua jenis gangguan (*noise*) hingga intensitas 25 %. Sedangkan untuk *Gaussian blur* hanya citra 2 yang bertahan pada radius 1 px, sedangkan pada citra 1 *watermark* tidak berhasil diekstraksi kembali.

## 7. Kesimpulan

Berdasarkan Tabel 2, seluruh *watermarked image* memiliki nilai PNSR diatas 59 dB, ini menunjukkan bahwa kualitas dari *watermarked image* hampir sama dengan citra aslinya. Hasil pengujian terhadap citra 1 dan 2 pada Tabel 3 menunjukkan bahwa metode dapat bertahan terhadap beberapa variasi pengolahan citra *random noise*. Untuk efek *Gaussian Blur* dengan radius 1 px, metode ini hanya mampu bertahan pada citra 2 dan mengalami kegagalan pada citra 1. Agar lebih kokoh, metode ini masih perlu dikembangkan lebih lanjut baik dari penggunaan koefisien wavelet yang lain (LH, HL, dan HH), ukuran blok yang digunakan maupun dari teknik penanaman *watermark*.

## 8. Daftar Pustaka

- [1] R. Munir, "Sekilas Image Watermarking untuk Memproteksi Citra Digital dan Aplikasinya pada Citra Medis," International Conference on Telecommunication, Bandung, 2006.
- [2] J. Pan, H. Huang, and L. Jain, "Intelligent watermarking techniques", Singapore: World Scientific Pub Co Inc.
- [3] Suhono H. Supangkat, Kuspriyanto, Juanda, "Watermarking sebagai Teknik Penyembunian Label Hak Cipta pada Data Digital", Jurnal Teknik Elektro ITB, Vol.6, No.3.
- [4] M.F.Al-Hunaity, S.A.Najim, and I.M.El-Emary, "Colored Digital Image Watermarking using Wavelet Technique," American Journal of Applied Sciences 4 (9):658-662, 2007.
- [5] T. Venkat Narayana Rao, A.Govardhan, "Comparative Study of Visible Reversible Watermarking Algorithms-Image Security Paradigm", International Journal of Engineering Studies, ISSN 0975-6469 Vol.1, Number 3, pp. 193-206, 2009.
- [6] I.F. Kallel, M.S. Bouhleb, and J. Lapayre, "Improved Tian 's Method for Medical Image Reversible Watermarking," vol. 7, 2007.
- [7] Xiang-Gen Xia, Charles G. Boncelet and Gonzalo R. Arce, "Wavelet Transform Based Watermark For Digital Images", OPTICS EXPRESS 498, Vol.3 No.12, 1998.
- [8] Shapiro, L.G. Stockman, G.C., "Computer Vision", page 137, 150. , Prentice Hall, 2001.
- [9] The GIMP Documentation Team, Hurl Overview, <http://docs.gimp.org/en/plugin-randomize-hurl.html>, (Diakses tanggal 12 Maret 2012).

- [10] The GIMP Documentation Team , Slur Overview, <http://docs.gimp.org/en/plugin-randomize-slur.html>, (Diakses tanggal 12 Maret 2012).
- [11] The GIMP Documentation Team, Pick Overview, <http://docs.gimp.org/en/plugin-randomize-pick.html>, (Diakses tanggal 12 Maret 2012).
- [12] Peak Signal to Noise Ratio, [http://en.wikipedia.org/wiki/Peak\\_signal-to-noise\\_ratio](http://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio), (Diakses tanggal 16 April 2012).
- [13] Ekasari Nugraheni, Nova Hadi Lestriandoko, Taufiq Wirahman, "A Block-based Wwatermarking Using Wavelet Coefficients", International Conference on Telecommunication Systems, Services, and Applications (TSSA), Denpasar Bali, Indonesia, 20-21 Oktober 2011.