ANALISIS DAN IMPLEMENTASI IMAGE WATERMARKING MENGGUNAKAN HISTOGRAM-BASED REVERSIBLE DATA HIDING DENGAN BORDER POINT DAN LOCALIZATION

Analysis And Implementation of Image Watermarking Using Histogram-Based Reversible Data Hiding With Border Point And Localization

Alfian Ghifari

Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro,
Universitas Telkom

alfianghifari@gmail.com

Abstrak

Penggunaan *internet* dewasa ini sangat mempermudah segala hal, salah satunya adalah publikasi karya secara *digital*. Mudahnya mengakses media yang dipublikasi secara *digital* menimbulkan masalah seperti pelanggaran hak cipta. Oleh karena itu dibutuhkan solusi, *digital watermarking* adalah salah satunya. Pada penelitian ini dilakukan analisis dan implementasi *watermarking* pada berkas citra menggunakan *histogram-based reversible data hiding* dengan *border point* dan *localization*. Dari hasil pengujian yang dilakukan, didapat rata-rata jumlah *Byte* yang dapat disisipkan pada *host image* dengan partisi blok terkecil 2x2 piksel adalah 9492 *Byte* (75942 *bit*), sementara pada partisi blok terbesar 64x64 piksel adalah 13 *Byte* (106 *bit*). Dan didapat PSNR sebesar 56,553 dB pada penyisipan dengan jumlah rata-rata 9492 *Byte* (75942 *bit*), sementara pada penyisipan dengan jumlah rata-rata 13 *Byte* (106 *bit*) didapat nilai PSNR sebesar 85,729 dB.

Kata kunci : *image watermarking, histogram, reversible data hiding, border point, PSNR, BER.*

Abstract

Nowadays internet usage really simplifies everything, such as digital work publication. The easiness of accessing published digital media cause problems like copyright infringement. Therefore, it need solutions, digital watermarking is one of the solutions. In this paper, writer will analyze and implement watermarking on image file using histogram-based reversible data hiding with border point and localization. From the result of tests performed, the amount of Bytes in average that can be embedded to host image with the smallest block partition 2x2 pixels is 9492 Bytes (75942 bits), while in the largest block partition 64x64 pixels is 13 Bytes (106 bits). And the PSNR for embedding 9492 Bytes (75942 bits) in average is 56,553 dB, while the PSNR for embedding 13 Bytes (106 bits) in average is 85,729 dB.

Keywords: image watermarking, histogram, reversible data hiding, border point, PSNR, BER.

1. Pendahuluan

1.1 Latar Belakang

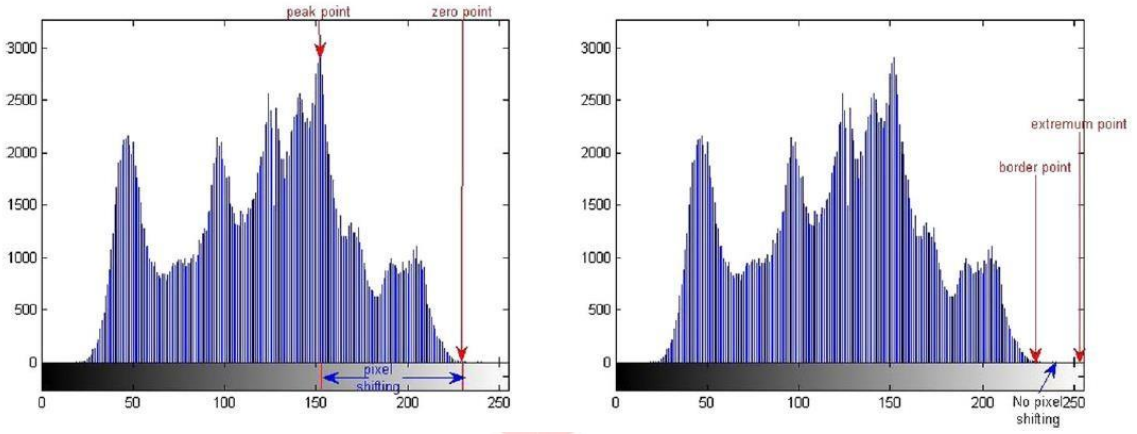
Akses informasi yang semakin mudah ternyata memiliki dampak negatif, contohnya adalah mudahnya melakukan pencurian, manipulasi, dan distribusi informasi tanpa memperhatikan aspek hak cipta. Gambar atau foto adalah salah satu target pelanggaran tersebut. Manipulasi, pengklaiman, atau penggunaan gambar tanpa izin dapat merugikan seseorang atau suatu lembaga pemilik hak cipta gambar tersebut.

Digital image watermarking dengan teknik *reversible data hiding* adalah salah satu solusi untuk mencegah pelanggaran hak cipta pada berkas citra. Teknik ini dapat menyembunyikan *watermark* pada berkas citra dan dapat mengekstrak kembali *watermark* untuk membuktikan hak cipta berkas citra tersebut. Pada referensi [1] dijelaskan bahwa metode *reversible data hiding* yang biasa digunakan adalah pergeseran *histogram*. Metode ini menggunakan *peak point* untuk menyisipkan data *watermark*. Sebelum penyisipan, semua piksel antara *peak point* dan *zero point* harus bergeser ke *zero point* guna memberikan ruang ekstra untuk penyisipan. Hal ini sering kali menyebabkan distorsi.

Berdasarkan latar belakang tersebut, pada tugas akhir ini akan dilakukan analisis dan perancangan *image watermarking* menggunakan *histogram-based reversible data hiding* dengan *border point* dan *localization*. Pada metode ini, penyisipan *watermark* akan dilakukan menggunakan *border point*, sedangkan *localization* digunakan untuk menghasilkan lebih banyak *border point* guna meningkatkan kapasitas penyisipan.

2. *Histogram-based Reversible Data Hiding* dengan *Border Point* dan *Localization*

Metode *histogram-based reversible data hiding* sudah sering kali dimodifikasi guna mendapatkan kualitas *embedded image* yang baik, dan juga meningkatkan kapasitas penyisipan. Salah satu modifikasi yang dilakukan adalah menggunakan *border point*. Penyisipan dilakukan antara *border point* dan *extremum point*. Karena jumlah piksel di antara kedua point tersebut sedikit, maka pergeseran yang dapat menyebabkan distorsi dapat dikurangi [1].



Gambar 2.1 Contoh perbandingan *histogram* [1].

2.1 Proses penyisipan

Host image akan dibagi-bagi menjadi blok-blok *non-overlapping* dengan ukuran S x S. *Border point* kanan digunakan untuk menyisipkan data, sedangkan yang kiri digunakan sebagai *reference point*. Lalu akan dihitung nilai residual *d* antara *border point* kanan dan kiri.

d = *Bright – Bleft* (2.1)

Ketika nilai *d* genap dan *secret bit* bernilai 0, maka *border point* kanan tidak berubah. Sedangkan jika *secret bit* bernilai 1, maka *border point* kanan bertambah 1.

{ (2.2)

Ketika nilai *d* ganjil dan *secret bit* bernilai 1, maka *border point* kanan tidak berubah. Sedangkan jika *secret bit* bernilai 0, maka *border point* kanan bertambah 1.

{ (2.3)

Karena suatu piksel bernilai maksimal 255, maka apabila host image mengandung nilai 255 akan terlebih dahulu di normalisasi menjadi 254. Dan untuk blok dengan nilai residual d 0, dimana semua nilai piksel pada blok tersebut sama, maka tidak akan lakukan penyisipan pada blok tersebut. Dibutuhkan pula sebuah bit map untuk mencatat fitur ganjil-genap (odd-even) dari nilai residual d. Bit map ini nantinya akan digunakan untuk me-recovery host image.

2.2 Proses ekstraksi dan *recovery*

Pada sisi penerima, *embedded host image* dibagi menjadi blok-blok *non-overlapping* dengan ukuran yang sama pada saat proses penyisipan. *Border point* kiri dipilih sebagai *reference point* karena tidak mengalami perubahan selama proses penyisipan. Nilai residual antara *border point* kiri dan kanan dikalkulasi dengan persamaan :

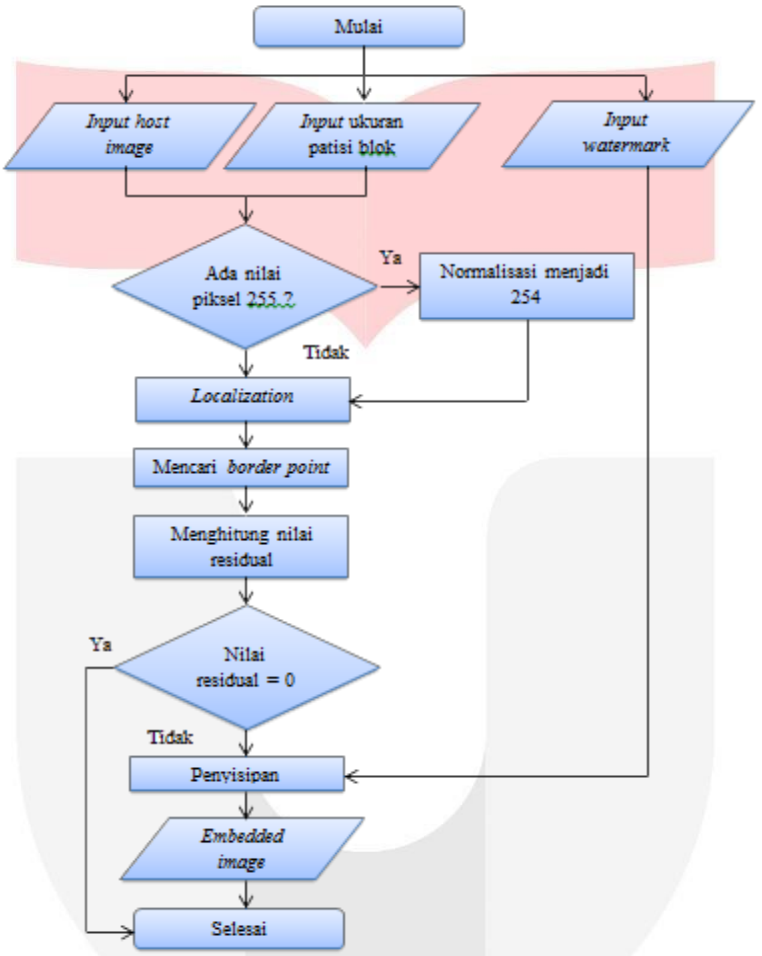
= B_{left} (2.4) Jika nilai residual genap, *secret bit* bernilai 0. Jika nilai residual ganjil, *secret bit* bernilai 1. Semua *secret* data dapat diekstrak dengan cara seperti ini untuk semua blok.

{ (2.5)

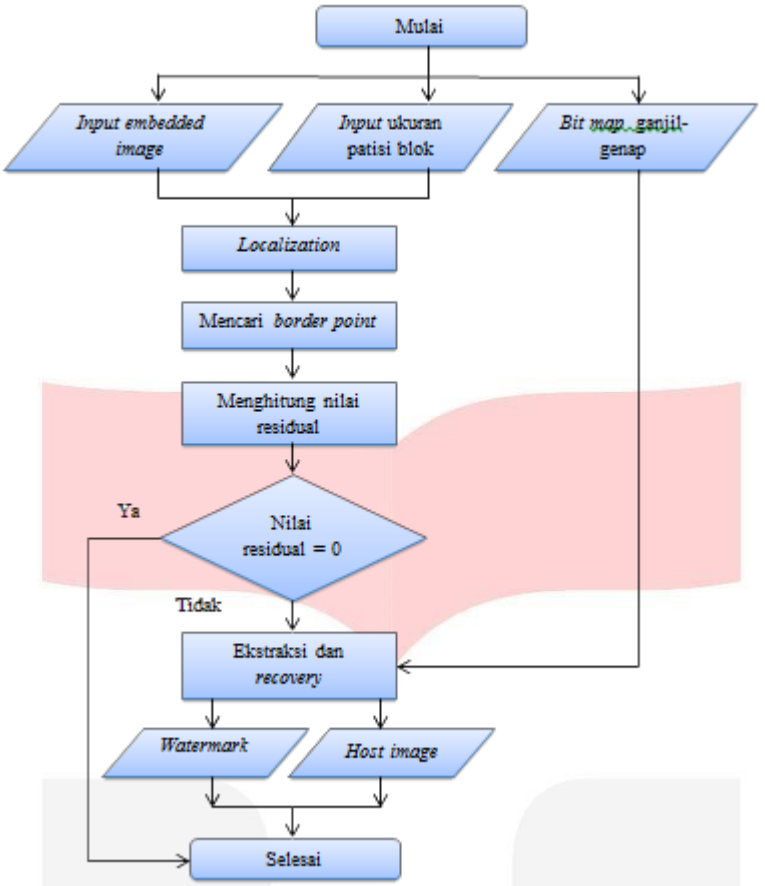
Sementara *secret data* diekstrak semuanya, *bit map odd-even* akan didekompres dan digunakan untuk memulihkan *border point* kanan yang asli. Jika fitur *odd-even* dari nilai residual untuk setiap *embedded block* sama dengan blok aslinya, *border point* kanan tidak berubah. Sedangkan jika berbeda, maka *border point* kanan dikurangi 1.

{ (2.6)

Dalam penulisan tugas akhir ini sistem penyisipan akan diilustrasikan dengan diagram alir (*flowchart*) pada gambar 2.1, sedangkan ekstraksi akan diilustrasikan dengan diagram alir (*flowchart*) pada gambar 2.2.



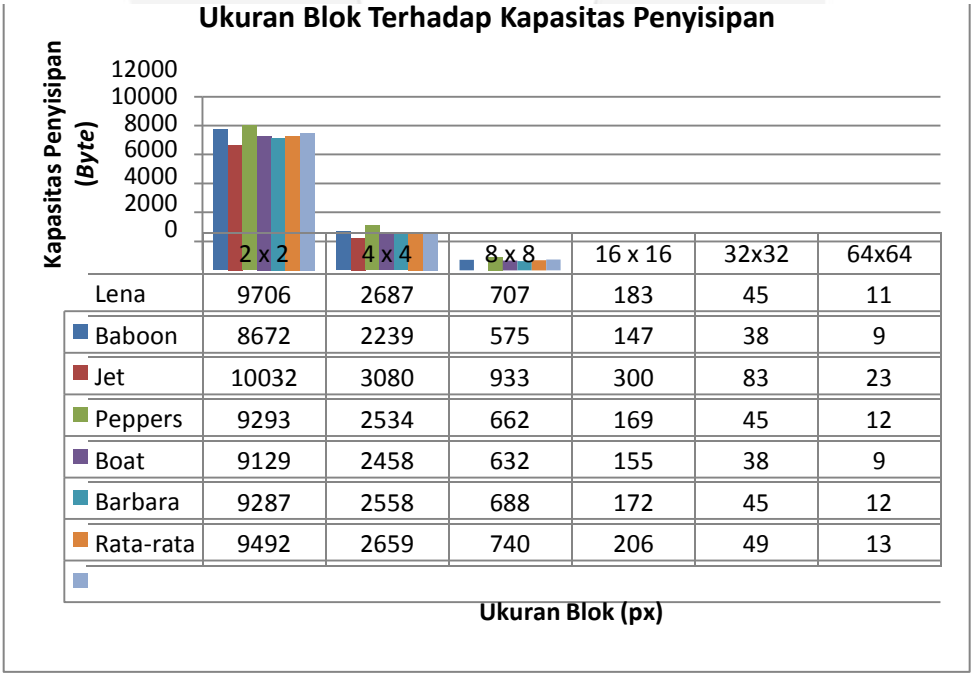
Gambar 2.1 Diagram alir sistem pada proses penyisipan.



Gambar 2.2 Diagram alir sistem pada proses ekstraksi.

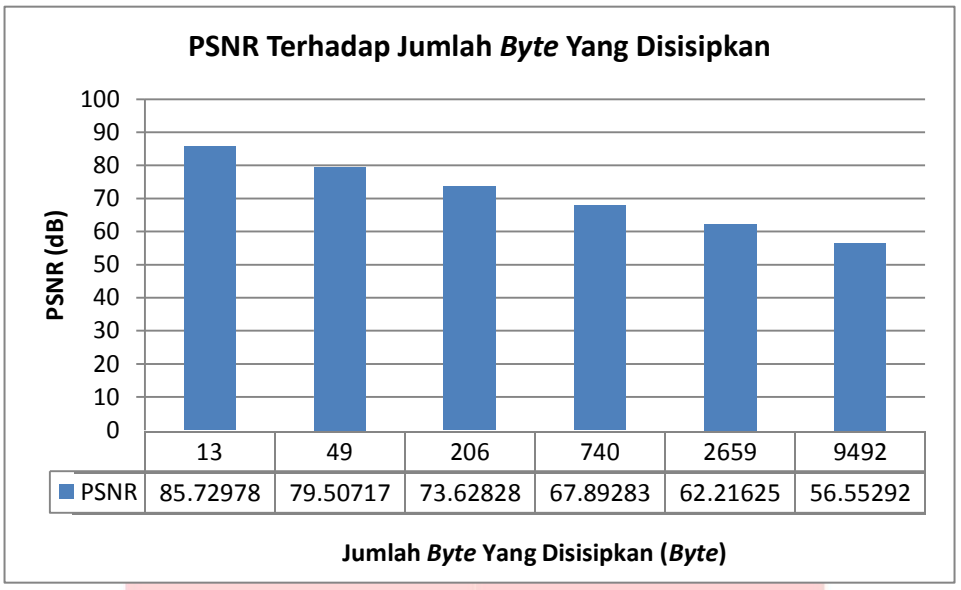
3. Pembahasan

Hasil dari pengujian kapasitas penyisipan dapat dilihat pada gambar 3.1 dengan hasil sebagai berikut :



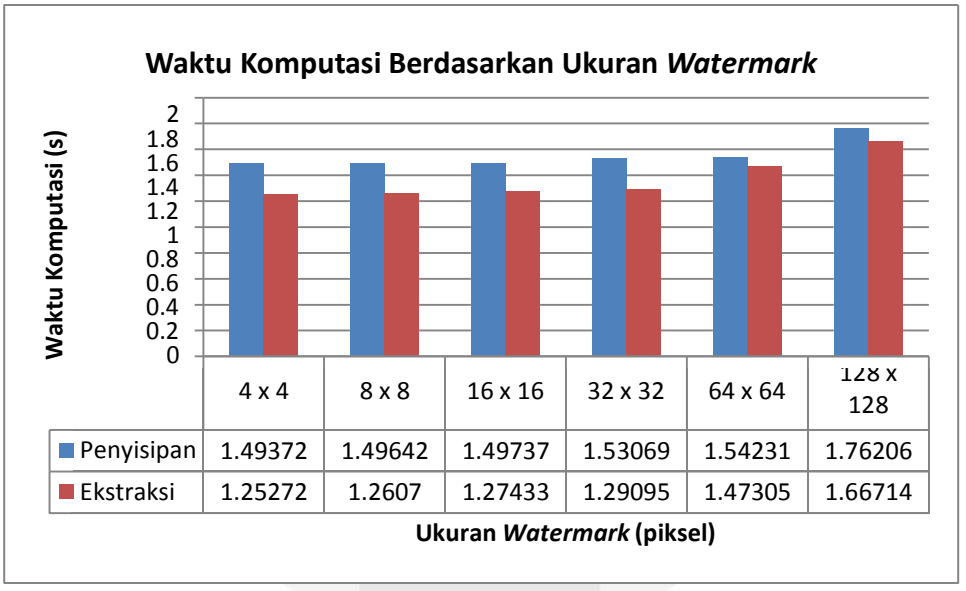
	2 x 2	4 x 4	8 x 8	16 x 16	32x32	64x64
Lena	9706	2687	707	183	45	11
Baboon	8672	2239	575	147	38	9
Jet	10032	3080	933	300	83	23
Peppers	9293	2534	662	169	45	12
Boat	9129	2458	632	155	38	9
Barbara	9287	2558	688	172	45	12
Rata-rata	9492	2659	740	206	49	13

Gambar 3.1 Grafik pengaruh ukuran partisi blok terhadap kapasitas penyisipan.

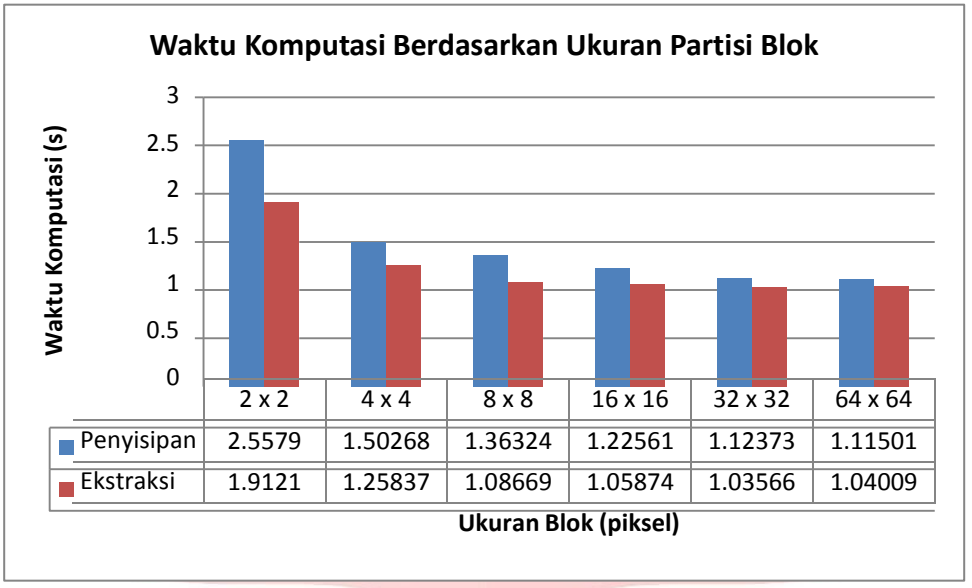


Gambar 3.2 Nilai rata-rata PSNR untuk jumlah *Byte* sisip rata-rata yang berbeda.

Pada pengujian ketahanan dengan penambahan *noise Gaussian*, *watermark* masih dapat terbaca hanya sampai variansi $3x10^{-6}$ dengan BER 0,381. Pada penambahan *noise Salt* & *Pepper*, nilai BER masih terbilang baik yaitu ≤ 0,2. Pada skenario kompresi di bawah kualitas 99% dan *rescaling*, didapat BER ≥0,49, dimana *watermark* sudah tidak dapat terbaca. Pada pengujian waktu komputasi didapat nilai seperti grafik di bawah :



Gambar 3.3 Waktu komputasi berdasarkan ukuran *watermark* yang disisipkan.



Gambar 3.4 Waktu komputasi berdasarkan partisi blok.

Dari grafik di atas dapat diketahui bahwa waktu komputasi dipengaruhi oleh ukuran *watermark* yang disisipkan dan ukuran partisi blok. Dimana, semakin besar ukuran *watermark* yang disisipkan maka semakin lama waktu komputasinya. Sedangkan semakin besar ukuran partisi blok maka waktu komputasi akan semakin cepat.

4. Kesimpulan

Adapun kesimpulan yang dapat ditarik dari hasil pengujian dan analisis diantaranya adalah :

1. Semakin kecil ukuran partisi blok, semakin besar jumlah *bit* yang dapat disisipkan. Dari hasil pengujian didapat rata-rata jumlah *bit* yang dapat disisipkan pada *host image* dengan partisi blok terkecil 2x2 piksel adalah 9492 *Byte* (75942 *bit*), sementara pada partisi blok terbesar 64x64 piksel adalah 13 *Byte* (106 *bit*).
2. Dari hasil pengujian, didapat PSNR sebesar 56,553 dB pada penyisipan dengan jumlah rata-rata 9492 *Byte* (75942 *bit*), sementara pada penyisipan dengan jumlah rata-rata 13 *Byte* (106 *bit*) didapat nilai PSNR sebesar 85,729 dB. Hal ini menunjukkan bahwa semakin besar jumlah *bit* yang disisipkan, maka akan semakin kecil nilai PSNR.
3. Pada skenario pengujian ketahanan dapat dilihat bahwa teknik watermarking dengan metode ini tidak memiliki ketahanan yang baik terhadap *noise Gaussian*, kompresi di bawah 99%, dan juga *rescaling*. Tetapi memiliki ketahanan yang cukup baik terhadap *noise Salt* & *Pepper*.
4. Waktu komputasi dipengaruhi oleh ukuran *watermark* yang disisipkan dan ukuran partisi blok. Dimana, semakin besar ukuran *watermark* yang disisipkan maka semakin lama waktu komputasi. Sedangkan semakin besar ukuran partisi blok maka waktu komputasi akan semakin cepat.

Daftar Pustaka:

[1] Pan, Zhibin; Hu, Sen; Ma, Xiaoxiao; Wang, Lingfei. 2014. *"A novel reversible data hiding using border point and localization"*. New York: Springer Science+Business Media.

[2] Johnson, Neil F., & Jajodia, Sushil. 1998. "*Exploring Steganography: Seeing the Unseen*". IEEE Journal, George Mason University.

[3] Cox, I. J; Miller, M.L; Bloom, J. A; Fridrich, Jessica; Kalker, Ton. 2008. *"Digital Watermarking and Steganography Second Edition"*. San Francisco: Morgan Kaufmann.

[4] Pamboukian, D; Vicente, Sergio; Kim, Hae Yong. 2006. *"Reversible data hiding and reversible authentication watermarking for binary images"*. Sao Paulo: SBSeg.

[5] Nosrati, Masoud; Karimi, Ronak; Hariri, Mehdi. 2012. *" Reversible Data Hiding: Principles, Techniques, and Recent Studies"*. World Applied Programming.

[6] Bansal, Raj Kumar; Goe, Ashok Kumar; Sharma, Manoj Kumar. 2009. *"MATLAB and Its Applications in Engineering"*. New Delhi : Pearson Education.

[7] Lim, Jit. 2010. "*Is BER the bit error ratio or the bit error rate?"*. California: EDN.
[8] Cheddad, A., Condell, J., Curran, K., Kevitt, P.Mc., 2010. "*Digital Image Steganography : Survey and Analysis of Current Methods. Signal Processing*". Northern Ireland : Elsevier.
[9] Male, Ghazali Moenandar; Wirawan; Setijadi, Eko. 2012. *"Analisa Kualitas Citra Pada Steganografi Untuk Aplikasi e-Government"*. Surabaya : Prosiding Seminar Nasional Manajemen Teknologi XV.
[10] S. Katzenbeisser and F. A. P. Petitcolas. 2000. "*Information Hiding Techniques for Steganography and Digital Watermarking*". Boston : Artech House.
[11] Jannah, Asmaniatul. 2008. "*Analisis Perbandingan Metode Filter Gaussian, Mean dan Median Terhadap Reduksi Noise Salt and Peppers*". Malang : UIN Malang.
[12] Pennebaker, William B. Mitchell, Joan L. 1993. "*JPEG still image data compression standard (3rd ed.)*". Springer. p. 291. ISBN 978-0-442-01272-4.
[13] Rinaldi Munir. 2004. "*Pengolahan Citra digital dengan Pendekatan Algoritmik*". Bandung : Informatika.
[14] Knox Keith T. 1999. "*Reversible Digital Images*". California : IS&T/SPIE Vol 3657.