Jurnal Informatika, Vol. 13, No. 1, Mei 2015, 9-14 DOI: 10.9744/informatika.13.1.9-14
ISSN 1411-0105

# KEAMANAN CITRA DENGAN *WATERMARKING* MENGGUNAKAN PENGEMBANGAN ALGORITMA *LEAST SIGNIFICANT BIT*

**Kurniawan[1], Indah Agustien Siradjuddin[2], Arif Muntasa[3]**
[1,2,3]Program Studi Teknik Informatika, Universitas Truonojoyo Madura
Jl. Raya Telang, PO BOX 2, Kamal, Bangkalan–69162, Indonesia
Email: ndoro.awank@gmail.com[1], indah.agustien@if.trunojoyo.ac.id[2], arifmuntasa@gmail.com[3]

**Abstrak:** *Image security* adalah proses mengamankan data digital yang berbentuk citra. Salah satu metode pengamanan data citra digital adalah *watermarking* menggunakan algoritma *Least Significant Bit* atau *LSB*. Konsep kerja *image security* menggunakan algoritma *LSB* adalah dengan mengganti nilai *bit* citra pada lokasi tertentu sehingga membentuk pola. Pola hasil dari pergantian nilai *bit* pada citra inilah yang disebut dengan tanda air atau *watermark*. Pemberian *watermark* pada citra menggunakan algoritma *LSB* memiliki konsep sederhana sehingga informasi yang disisipkan akan mudah hilang saat mengalami serangan seperti serangan *noise* atau kompresi. Sehingga perlu dilakukan modifikasi yaitu pengembangan algoritma *LSB*. Hal ini dilakukan untuk mengurangi distorsi informasi *watermark* terhadap serangan yang terjadi. Dalam penelitian ini dibagi menjadi 6 proses, yaitu ekstraksi kanal warna citra *cover*, pencarian area *busy*, penyisipan *watermark*, perhitungan akurasi penyisipan, ekstraksi *watermark* dan perhitungan akurasi ekstraksi. Ekstraksi kanal warna citra *cover* adalah proses mendapatkan kanal warna biru pada citra *cover*. Informasi *watermark* akan disisipkan pada area *busy* atau sibuk dengan mencari area yang memiliki unsur terbanyak pada citra cover. Selanjutnya citra *watermark* disisipkan ke dalam citra cover sehingga menghasilkan citra terwatermark menggunakan algoritma beberapa pengembangan LSB dan mencari akurasinya dengan menghitung nilai *Peak Signal to Noise Ratio*. Sebelum citra terwatermark diekstraksi, dilakukan pengujian dengan memberi noise dan melakukan kompresi format *jpg*. Akurasi hasil ekstraksi dicari dengan menghitung nilai *Bit Error Rate*.

**Kata kunci:** *Image Security; Watermarking; Least Significant Bit;* Pengembangan Algoritma *LSB*.

***Abstract:*** *Image security is a process to save digital. One method of securing image digital is watermarking using Least Significant Bit algorithm. Main concept of image security using LSB algorithm is to replace bit value of image at specific location so that created pattern. The pattern result of replacing the bit value of image is called by watermark. Giving watermark at image digital using LSB algorithm has simple concept so that the information which is embedded will lost easily when attacked such as noise attack or compression. So need modification like development of LSB algorithm. This is done to decrease distortion of watermark information against those attacks. In this research is divided by 6 process which are color extraction of cover image, busy area search, watermark embed, count the accuracy of watermark embed, watermark extraction, and count the accuracy of watermark extraction. Color extraction of cover image is process to get blue color component from cover image. Watermark information will embed at busy area by search the area which has the greatest number of unsure from cover image. Then watermark image is embedded into cover image so that produce watermarked image using some development of LSB algorithm and search the accuracy by count the Peak Signal to Noise Ratio value. Before the watermarked image is extracted, need to test by giving noise and doing compression into jpg format. The accuracy of extraction result is searched by count the Bit Error Rate value.*

***Keywords****: Image Security, Watermarking, Least Significant Bit,* Development of LSB Algorithm.

## PENDAHULUAN

*Internet* saat ini bukan lagi menjadi alat pendukung yang dapat membantu pekerjaan manusia, bahkan sebagian orang menjadikan *internet* sebagai kebutuhan utama dalam hidupnya. Fungsinya yang sangat banyak menjadikan alasan para pengguna tidak bisa lepas dari *internet*. Selain sebagai media pencarian informasi, media *social* dan media pembelajaran, kebanyakan orang memanfaatkan *internet* sebagai media utama pengiriman data *digital* dari tempat satu ke tempat yang lain bahkan lintas negara. Perpindahan data dibuat sangat mudah dan cepat dengan internet.

Apabila dilihat dari fungsinya, data dapat dibagi menjadi tiga yaitu *public*, *private* dan *protected*. *Public* menunjukkan bahwa data tersebut bersifat untuk umum, *private* menunjukkan bahwa data

tersebut bersifat pribadi dan *protected* ialah data yang bersifat dilindungi. Data yang bersifat *private* biasanya si pemilik tidak ingin data tersebut dilihat oleh orang lain sehingga disimpan secara pribadi. Sedangkan data *protected* biasanya dikhususkan untuk data yang bersifat penting sehingga keamanannya sangat diperhatikan. Berdasarkan sifat-safat tersebut, muncul penilitian terkait keamanan data. Penelitian ini bertujuan untuk mencari bagaimana cara melindungi data dari segala kemungkinan manipulasi atau kerusakan yang terjadi.

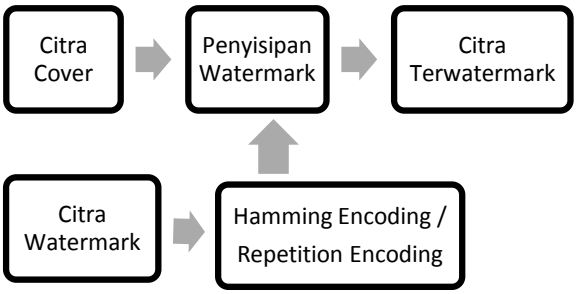
*Watermarking* adalah teknik yang digunakan untuk menyisipkan informasi pada media multimedia dengan tujuan memberi tanda kepemilikan atau menjaga keaslian data tetapi tidak diketahui keberadaannya oleh indera manusia. Watermarking dapat dimanfaatkan pada media digital seperti data *text*, citra, audio dan video. Dalam penelitian Tugas Akhir ini akan mengulas salah satu metode *watermarking* pada citra yaitu *Least Significant Bit (LSB)*. Penulis juga melakukan penggabungan metode enkripsi *hamming code* dan *repetition code* yang bertujuan untuk memperbaiki kekurangan dari metode *classic* dari *LSB*. Hasil dari penggabungan metode-metode tersebut dinamakan pengembangan *Least Significant Bit*.

## *WATERMARKING*

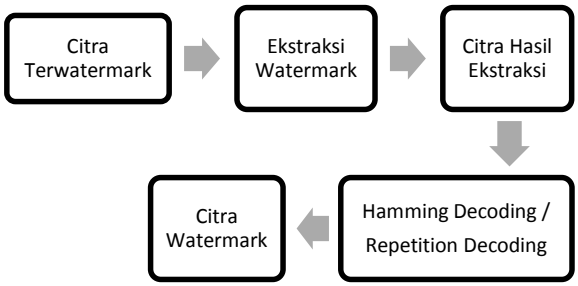
*Watermarking* adalah kumpulan *bit* pola atau informasi yang disisipkan pada citra digital, audio atau video yang menunjukkan kepemilikan, kebenaran dan lain sebagainya [1]. Disamping itu *watermarking* digunakan untuk menjaga kebsahan atau melindungi data. Walau bagaimanapun, *watermarking* harus tak terlihat kehadirannya pada citra yang disisipi *watermark* [2]. Dan juga *watermarking* harus cukup tahan (*robust)* terhadap dari segala perubahan yang bertujuan menghilangkan informasi *watermark* seperti penambahan *noise* dan lain sebagainya [3].

Pada citra digital, informasi dapat disisipkan secara langsung ke dalam setiap *bit* citra atau pada area tertentu sehingga dapat menyembunyikan informasi pada citra [4]. *Least Significant Bit (LSB)* adalah salah satu metode penyisipan *watermark* pada citra dengan cara mengganti nilai pada *bit* tertentu citra. Citra *watermark* berbentuk citra *biner* disisipkan pada citra berwarna atau citra *RGB*. Agar *watermark* tidak tampak pada citra yang akan disisipkan *watermark*, komponen yang akan diubah nilai pikselnya adalah salah satu dari komponen *RGB* tersebut.

Setiap komponen citra *RGB*, nilai piksel pada citra memiliki rentang nilai 0-255 atau 8 *bit*. Prinsip penyisipan pada Algoritma *LSB* yaitu dengan mengganti nilai *biner* ke-*n* menjadi nilai *watermark*. Jika nilai piksel pada citra adalah 138 maka jika dilakukan konversi ke *biner* menjadi 10000110 dan semisal nilai *watermark* yang disisipkan adalah 1. Maka nilai piksel tersebut akan menjadi 10000111 atau 139 dalam bentuk decimal. Metode *hybrid* atau gabungan yang digunakan pada penelitian ini adalah *LSB with Hamming Code* dan *LSB with Repetition Code*. Gambar 1a dan 1b menunjukkan alur kerja sistem secara umum.




**Gambar 1a.** Proses Penyisipan *Watermark*




**Gambar 1b.** Proses Ekstraksi *Watermark*

### Penentuan Area Sibuk

Area sibuk atau *busy* pada penelitian ini ditentukan dengan cara menghitung jumlah ragam objek. Ragam objek diperoleh dengan proses *labeling*. Proses *labeling* adalah proses membaca nilai 1 pada citra biner dan memberi dengan nilai 1 sampai *n* dengan aturan ketetanggaan. Namun sebelum proses *labeling*, citra dilakukan operasi *splitting* atau pembagian menjadi blok-blok kecil seukuran citra *watermark*. Blok citra yang memiliki ragam atau label terbanyak, maka blok tersebut dinyatakan sebagai lokasi sibuk atau *busy* dan tempat citra *watermark* disisipkan. Dalam pelabelan menggunakan aturan 4 atau 8 obyek yang terhubung. Sebagai contoh *labeling 4 connected* dan *labeling 8 connected* dapat dilihat pada Gambar 2a dan 2b.

### *Hamming Code*

*Hamming code* adalah salah satu enkripsi yang sering digunakan dalam keamanan text. Dalam *(n,k) Hamming code* dimana *k* adalah jumlah *bit* pesan, *n* adalah jumlah *bit* yang berkorespodensi hasil enkripsi

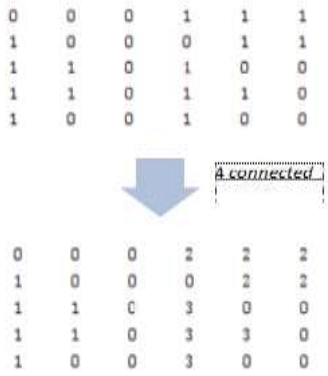
dengan $n>k$ dan $(n-k)$ adalah jumlah cek *bit* [8]. *Hamming code* dapat dirumuskan seperti berikut:

$n = 2^m - 1$

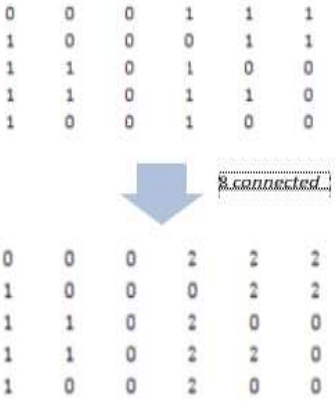
$k = 2^m - 1 - m$

Dimana m = 2, 3, 4, . . .

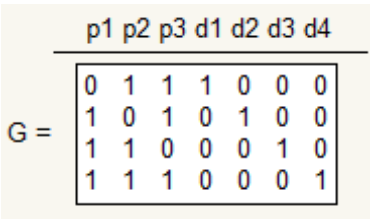
Jika m = 3 maka berkoresponden dengan kode (7,4) *hamming code*. *Hamming code* secara umum tersusun dari matrik parity dan matrik identitas. Dalam kasus (7,4) *hamming code* dapat dilihat pada Gambar 3.




**Gambar 2a.** Contoh *Labeling 4 Connected*




**Gambar 2b.** Contoh *Labeling 8 Connected*




**Gambar 3.** Matrik (7,4) *Hamming code*

Dengan aturan:

$p1 = d2 \oplus d3 \oplus d4$

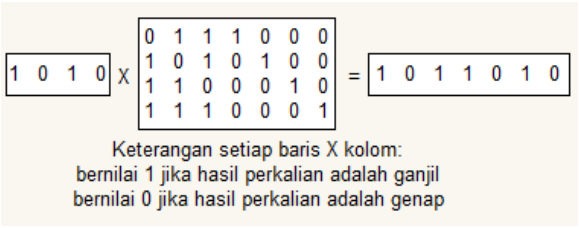
$p2 = d1 \oplus d3 \oplus d4$

$p3 = d1 \oplus d2 \oplus d4$

$\oplus$ adalah operasi logika XOR

d1, d2, d3 dan d4 merupakan matrik identitas

Sehingga hasil enkripsi nilai 1010 menggunakan (7,4) *hamming code* dapat dilihat pada Gambar 4 di bawah ini.




**Gambar 4.** Enkripsi Menggunakan (7,4) *Hamming code*

### *Repetition Code*

*Repetition code* merupakan model sederhana untuk enkripsi block linear dimana setiap nilai *bit* dienkripsi sebanyak n dengan nilai sama. Contoh kasus (3,1) *repetition code* nilai 1 *bit* akan dienkripsi menjadi 3 *bit* dengan mengulang nilai *bit* yang sama. Sama halnya dengan (5,1) *repetition code* nilai 1 *bit* akan dienkripsi menjadi 5 *bit* dengan mengulang nilai *bit* yang sama [8]. Tabel 1 menunjukkan enkripsi (3,1) *repetition code* dan (5,1) *repetition code*.

**Tabel 1.** Enkripsi Data Menggunakan *Repetition Code*

| *Repetition code* | Pesan *Bit* | Enkripsi Data |
|---|---|---|
| (3,1) | 00 | 000000 |
| | 01 | 000111 |
| (5,1) | 10 | 1111100000 |
| | 11 | 1111111111 |

### Perhitungan Akurasi

*Mean Square Error* (MSE) dan *Peak Signal to Noise Ratio (PSNR)* adalah dua pengukuran yang biasa digunakan untuk membandingkan kualitas kompresi citra. Rasio ini sering digunakan sebagai mengukur kualitas antara citra asli dan citra terwatermark.

Tujuan dari pengukuran *PSNR* adalah untuk membandingkan kualitas antara citra cover *(CI)* dan citra terwatermark *(TWI)*. Untuk menghitung nilai *PSNR*, terlebih dahulu harus menghitung nilai *MSE*. *MSE* sendiri adalah nilai error kuadrat rata-rata antara *CI* dan *TWI* [5]. *MSE* dapat dihitung dengan rumus:

$$MSE = \frac{1}{m * n} \sum_{i=1}^{m} \sum_{j=1}^{n} [I(i,j) - K(i,j)]^2 \quad (3)$$

Keterangan:

*I* : citra asli

*K* : citra terwatermark

*m,n* : dimensi citra

Semakin tinggi nilai PSNR menunjukkan semakin erat kemiripannya antara *CI* dan *TWI*. *PSNR* dapat dihitung dengan rumus:

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

Keterangan:
$MAX_I$ : nilai maksimum dari jenis citra. Untuk citra *grayscale* $MAX_I$ bernilai 255
*MSE* : nilai *MSE* yang diperoleh dari persamaan (3)

Tingkat ketahanan antara *original watermark* dan hasil ekstraksi *watermark* dapat diukur dengan menghitung nilai *Bit Error Rate (BER)*. *BER* dapat dihitung dengan rumus :

$$BER = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} W(i,j) \oplus W'(i,j)}{m,n} \quad (5)$$

Keterangan:
*W(i,j)* : *original watermark*
*W'(i,j)* : ekstraksi *watermark*
*m,n* : dimensi citra *watermark*
$\oplus$ : operasi Boolean XOR

Pada keadaan ideal nilai *BER* seharusnya bernilai 0 dan maksimal error bernilai 1 [6].

## HASIL DAN PEMBAHASAN

Sistem keamanan citra dengan *watermarking* menggunakan algoritma *hybrid least significant bit* diuji dengan beberapa skenario uji coba penyisipan *watermark* dan ekstraksi *watermark*. Tabel 2 merupakan skenario uji coba penyisipan *watermark* dan Tabel 2 merupakan skenario uji coba ekstraksi *watermark* yang dapat dilihat di Tabel 2.

**Tabel 2.** Skenario Uji Coba Penyisipan *Watermark*

| Skenario | Metode *LSB* | Level |
|---|---|---|
| 1 | *LSB with Classic Method* | 1 - 8 |
| 2 | *LSB with Hamming Code* | 1 - 8 |
| 3 | *LSB with Repetition 3* | 1 - 8 |
| 4 | *LSB with Repetition 5* | 1 - 8 |

**Tabel 3.** Skenario Uji Coba Ekstraksi *Watermark*

| Skenario | Metode *LSB* | Noise | Kompres |
|---|---|---|---|
| 1 | *LSB with Classic Method* (level 1 - 8) | a) 0.3<br>b) 0.5 | a) Format jpg/jpeg |
| 2 | *LSB with Hamming Code* (level 1 - 8) | a) 0.3<br>b) 0.5 | a) Format jpg/jpeg |
| 3 | *LSB with Repetition 3* (level 1 - 8) | a) 0.3<br>b) 0.5 | a) Format jpg/jpeg |
| 4 | *LSB with Repetition 5* (level 1 - 8) | a) 0.3<br>b) 0.5 | a) Format jpg/jpeg |

### Hasil Uji Coba

Data uji coba yang digunakan pada penelitian ini adalah 10 citra RGB yang berukuran 512x512 piksel yang didownload dari *SIPI Database*. Dan citra *biner watermark* berukuran 64x64 piksel. Data uji coba dapat dilihat pada Gambar 5.


**Gambar 5.** Data Uji Coba

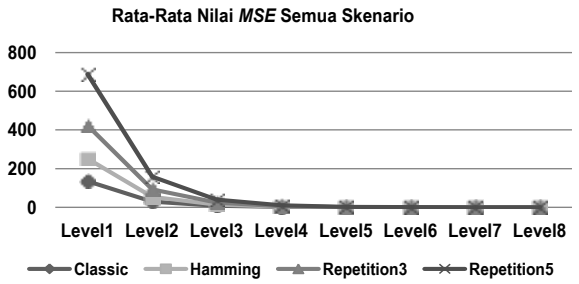
Pada skenario 1, 2, 3 dan 4 hasil penyisipan *watermark* dengan metode *LSB* di level 1 terlihat sangat mencolok citra watermarknya dan berangsur tak terlihat sampai level 8. Hasil perbandingannya dapat dilihat pada Tabel 4.

**Tabel 4.** Hasil Uji Coba Penyisipan

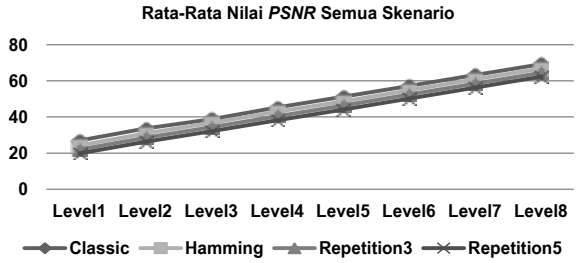
| Level | Skenario 1 | Skenario 2 | Skenario 3 | Skenario 4 |
|---|---|---|---|---|
| 1 |  *MSE* = 150.188 *PSNR* = 26.364 |  *MSE* = 256.813 *PSNR* = 24.035 |  *MSE* = 382.438 *PSNR* = 22.305 |  *MSE* = 666.25 *PSNR* = 19.894 |
| 3 |  *MSE* = 7.34 *PSNR* = 39.474 |  *MSE* = 13.563 *PSNR* = 36.807 |  *MSE* = 26.188 *PSNR* = 33.95 |  *MSE* = 41.68 *PSNR* = 31.932 |
| 8 |  *MSE* = 0.008 *PSNR* = 69.109 |  *MSE* = 0.014 *PSNR* = 66.732 |  *MSE* = 0.024 *PSNR* = 64.406 |  *MSE* = 0.039 *PSNR* = 62.182 |

Nilai MSE dan PSNR dapat dihitung dengan membandingkan citra cover dan citra terwatermark. Nilai MSE pada level 1 lebih besar dibandingkan pada level 2 sampai 8. Semakin tinggi nilai MSE menunjukkan semakin tinggi perbedaan yang telah terjadi. Lain halnya dengan PSNR, nilai PSNR menunjukkan tingkat kelayakan citra watermark yang tertanam dari segi penampakan. Pada skenario 2, 3 dan 4 sebelum watermark disisipkan, citra watermark dilakukan enkripsi menggunakan metode hamming dan repetition sehingga ukuran citra berubah menjadi lebih besar. Keadaan ini menyebabkan nilai MSE menjadi tinggi. Rata-rata MSE dan PSNR untuk semua skneario dapat dilihat grafik pada Gambar 6 dan Gambar 7. Hasil ekstraksi watermark dapat dilihat pada Tabel 5.

Penulis sengaja memberikan perlakuan noise dan kompresi pada citra terwatermark sebelum dilakukan ekstraksi. Ini bertujuan untuk mengetahui tingkat ketahan citra watermark. BER dihitung untuk mengetahui tingkat error hasil ekstraksi watermark setelah diberi beberapa perlakuan. Dari Tabel 5 di atas terlihat bahwa hasil ekstraksi terbaik dengan perlakuan noise 0,3, noise 0,5 dan kompresi adalah skenario 4 yaitu *LSB with Repetition 5*. Rata-rata BER untuk noise 0.3, 0.5 dan kompresi jpg/jpeg semua skneario dapat dilihat grafik pada Gambar 8, 9, dan 10).



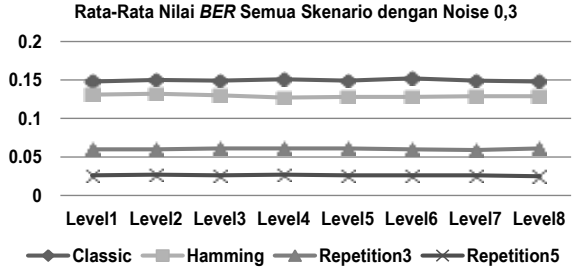

**Gambar 6.** Diagram Rata-Rata *MSE* Semua Skenario



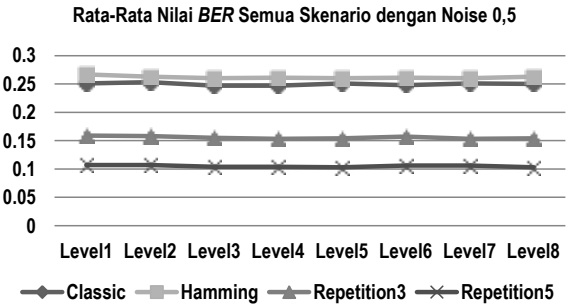

**Gambar 7.** Diagram Rata-Rata *PSNR* Semua Skenario

**Tabel 5.** Hasil Uji Coba Ekstraksi

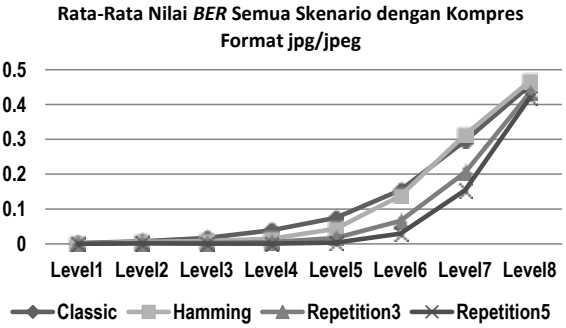
| Perlakuan | Skenario 1 | Skenario 2 | Skenario 3 | Skenario 4 |
|---|---|---|---|---|
| Noise 0,3 |  |  |  |  |
| | *BER* = 0.151 | *BER* = 0.134 | *BER* = 0.064 | *BER* = 0.025 |
| Noise 0,5 |  |  |  |  |
| | *BER* = 0.265 | *BER* = 0.261 | *BER* = 0.161 | *BER* = 0.107 |
| Kompres jpg/jpeg |  |  |  |  |
| | *BER* = 0.0085 | *BER* = 0.0005 | *BER* = 0.00024 | *BER* = 0 |




**Gambar 8.** Diagram Rata-Rata *BER* Semua Skenario dengan *Noise* 0.3




**Gambar 9.** Diagram Rata-Rata *BER* Semua Skenario dengan *Noise* 0,5




**Gambar 10.** Diagram Rata-Rata *BER* Semua Skenario dengan Kompresi Format jpg/jpeg

## KESIMPULAN

Keamanan citra dengan *watermarking* menggunakan algoritma *hybrid least significant bit* mampu menyimpan informasi *watermark* dengan baik. Dengan ditambahnya konsep *hybrid* yaitu dengan penambahan enkripsi *hamming code* dan *repetition code* mampu memperbaiki hasil ekstraksi *watermark* dibandingkan dengan metode *classic* atau LSB murni.

Penyisipan watermark pada level yang berbeda mempengaruhi penampakan pada citra hasil. Dengan metode LSB, penyisipan pada level 1 dan 2 masih terlihat secara mencolok watermarknya, dan berangsur tak terlihat dari level 3 sampai level 8. Keadaan ini memnuhi hakekat *watermarking* yaitu harus tak terlihat keberadaannya.

Algoritma *hybrid least significant bit* juga telah terbukti dapat menjaga informasi walaupun telah dilakukan simulasi serangan seperti pemberian noise dan kompresi kualitas jpg/jpeg. Dari percobaan 4 skenario yaitu *LS with classic method*, *LSB with hamming code*, *LSB with repetition 3* dan *LSB with repetition 5*.

## DAFTAR PUSTAKA

[1] Cramer C. 2005. *About Digital Watermarking*. http://www.willamette.edu/wits/idc/mmcamp/watermarking.htm. diakses 9 September 2014 jam 23.00.

[2] Serrao, C. dan Guimaraes, J. 1999. *Protecting Intellectual Proprietary Right through Secure Interactive Contract Negotiation*. Springer-Verlag Berlin Heidelberg 1999.
[3] Gulati, K. 2003. *Information Hiding Using Fractal Encoding*. Thesis for master degree, Mumbai, India.
[4] Titty, T. Steganography: *Reversible Data Hiding Methods for Digital Media*. Bachelor Project.
[5] Chopra, Deepshikha, Gupta, Preeti., Sanjay, G.B.C., Gupta, Anil. 2012. Lsb Based Digital Image Watermarking For Gray Scale Image. *IOSR Journal of Computer Engineering*, Vol. 6, Issue 1.
[6] Rohith, Mr S dan Dr. K. N. hari bhat. 2012. *A* Simple Robust Digital Image Watermarking against Salt and Pepper Noise using Repetition codes. *ACEEE Int. J. on Signal & Image Processing*, Vol. 03, No. 01.
[7] Bamatraf, Abdullah., Ibrahim, Rosziati,. Najib, Mohd, M.S. 2011. A New Digital Watermarking Algorithm Using Combination of Least Significant Bit (LSB) and Inverse Bit. *Journal of Computing*, Vol. 3 Issue 4.
[8] Verma, Rajni dan Archana Tiwari. 2014. Copyright Protection for Watermark Image Using LSB Algorithm in Colored Image. *Research India Publications*, Vol. 4 No. 5.
[9] Murni, Aniati. 1992. *Pengantar Pengolahan Citra*. Jakarta. Penerbit: PT Elexmedia Komputindo.
[10] Sitorus, S., Suyanto. 2006. *Pengolahan Citra Digital*. Medan. Penerbit: USU Press.
[11] Sutoyo. T. 2009. *Teori Pengolahan Citra Digital*. Yogyakarta. Penerbit: ANDI.
[12] Pengolahan Citra. http:// id.wikipedia.org/wiki/ Pengolahan_citra. Diakses pada 13 Oktober 2014 jam 21.15.