Kinerja Metode *Minimum Error Least Significant Bit Replacement Advanced Encryption Standard* Pada Citra Digital

Gunawan, K. [1)], Rachman, A.S. [2)]
STMIK Mataram [1)]
Universitas Mataram [2)]
karya.gunawan@stmikmataram.ac.id [1)], asrachman@unram.ac.id [2)]

Abstrak – Dalam steganografi, ada beberapa media digital yang dapat digunakan sebagai cover untuk menyembunyikan keberadaan sebuah pesan, seperti: citra, audio, teks, video. Dalam makalah ini, media cover yang digunakan adalah citra digital dengan format piksel 24-bit. Metode steganografi yang digunakan adalah model steganografi yang didasarkan pada penyisipan secara *Minimum Error Least Significant Bit Replacement-Advanced Encryption Standard* (MELSBR-AES), yaitu LSB (*Least Significant Bits*) yang bertujuan untuk meningkatkan kapasitas penyisipan dengan hasil yang masih dapat ditoleransi.

Hasil analisis menunjukkan bahwa kinerja metode yang digunakan ini dapat menjalankan kedua tugas utama dalam sistem steganografi (penyisipan dan pengektrakan) dengan baik. Dari hasil percobaan terhadap beberapa citra uji, dapat diketahui bahwa besar ukuran *message file* yang disisipkan harus lebih kecil , maksimal 10% dari ukuran *cover image*. Citra yang baik untuk digunakan sebagai *cover image* adalah citra yang memiliki kekontrasan yang tinggi (*real cover image*).

Kata kunci: steganografi, cover image, piksel, least significant bits, minimum-error replacement

1. Latar Belakang

Internet telah mengalami perkembangan yang pesat, teknologi ini mampu mengintegrasikan /mengkomunikasikan hampir semua komputer yang ada di dunia sehingga bisa saling berkomunikasi dan bertukar data. Bentuk data yang dapat ditukar diantaranya yaitu citra, text, audio, video digital dan lain-lain.

Sebagai suatu jaringan publik, internet rawan terhadap pencurian data, dan pencegahan pengkopian secara langsung. Untuk menjaga keamanan data yang ingin dikirimkan melalui jaringan internet. Dapat digunakan dengan metode steganografi atau metode kriptografi.

Steganografi merupakan seni untuk menyembunyikan pesan di dalam pesan lainnya sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu di dalam pesan tersebut. Dengan kata lain steganografi menyembunyikan pesan sehingga tidak terlihat, sedang kriptografi mengacak pesan sedemikian rupa sehingga tidak dimengerti. Pesan dalam kriptografi mungkin akan menimbulkan kecurigaan sedangkan pesan yang dibuat dengan steganografi tidak. [SUB-02]

Dalam steganografi atau penyembunyian pesan menurut [JNS-06] dalam papernya yang berjudul “Steganalysis of Images Created Using Current Steganography Software” memiliki beberapa komponen yang kemudian ditulis sebagai berikut:

cover_medium + hidden_data + stego_key = stego_medium

Berdasarkan uraian di atas, makalah ini akan dibahas mengenai bagaimana menyembunyikan suatu pesan kedalam pesan lainnya yaitu file citra, tentunya dengan mempertahankan kualitas citra hasil penyisipan (*stego image*) supaya tidak jauh berbeda dengan citra asli (*cover image*) dengan menggunakan metode MELSBR (*Minimum Error Least Significant Bit Replacement*).

2. Tinjauan Pustaka

a. Penyisipan LSB (Least Significant Bit)

Metode penyisipan LSB ini adalah menyisipi pesan dengan cara mengganti bit ke 8, 16 dan 24 pada representasi biner file citra dengan representasi biner pesan rahasia yang akan disembunyikan. Dengan demikian pada setiap pixel file citra BMP 24 bit dapat disisipkan 3 bit pesan, misalnya terdapat data raster original file citra adalah sebagai berikut:

00100111	11101001	11001000
00100111	11001000	11101001
11001000	00100111	11101001

Sedangkan representasi biner huruf A adalah 01000001, dengan menyisipkannya ke dalam *pixel* di atas maka akan dihasilkan:

00100110	11101001	11001000
00100110	11001000	11101000
11001000	00100111	11101001

Terlihat pada bit ke-8, 16 dan 24 diganti dengan representasi biner huruf A, dan hanya tiga bit rendah yang berubah (cetak tebal). [SED-96]

b. Tahapan Utama Proses Penyisipan

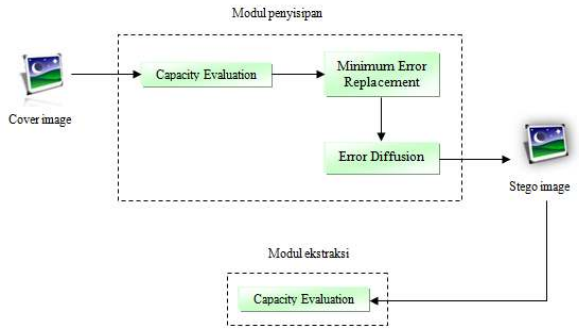
Metode MELSBR yang diterapkan pada citra berwarna (bitmap 24-bit) memiliki beberapa langkah atau tahapan utama untuk melakukan proses penyisipan, antara lain Capacity Evaluation, Minimum Error Replacement dan Error Diffusion. Untuk proses ekstraksi, tahapan yang dilakukan yaitu Capacity Evaluation [GMD-03]. Adapun gambaran umum dari metode yang diajukan adalah seperti pada Gambar 1.

Sebelum dilakukan proses penyisipan, maka langkah pertama yang harus dilakukan adalah mengevaluasi kapasitas penyisipan (*capacity evaluation*) dan mencari nilai color variation. Kemudian setelah mendapatkan nilai color variation, nilai tersebut diproses kembali untuk mendapatkan kapasitas penyisipan sejumlah K-bit, selanjutnya untuk beradaptasi dengan karakteristik lokal piksel, maka sejumlah K-bit tersebut ditangani dengan proses evaluasi kapasitas.

Proses selanjutnya adalah mencari MER, dimana proses ini akan menentukan apakah bit ke K+1 akan dilakukan perubahan atau tidak, dan yang akan menentukan itu adalah berdasarkan pada nilai *embedding error* (Er).

c. *Capacity Evaluation*

Capacity Evaluation merupakan tahap pertama dan yang paling krusial dari metode penyisipan MELSBR. Tahap ini mengacu pada karakterisitik human visual system (HVS) yang tidak sensitif terhadap noise dan perubahan warna yang terdapat di dalam citra [GMD-03].



Gambar 1 Gambaran umum metode MELSBR. Sumber: [GMD-03]

Langkah pertama yang akan dilakukan pada evaluasi kapasitas adalah mencari nilai color variation (V) atau variasi warna yang melibatkan piksel A, B, C dan D. Adapun rumus dari V adalah sebagai berikut [GMD-03].

$$V = round\ \{(|C-A|+|A-B|+|B-C|+|C-D|)/4\}$$

dimana :

V = variasi warna (*color variation*)
round = fungsi matematika untuk pembulatan

Rumus di atas akan menghasilkan ketentuan toleransi modifikasi yang akurat di setiap piksel P. Langkah ke-dua adalah mencari kapasitas penyisipan (K) pada piksel P dan dapat diterapkan rumus sebagai berikut [GMD-03].

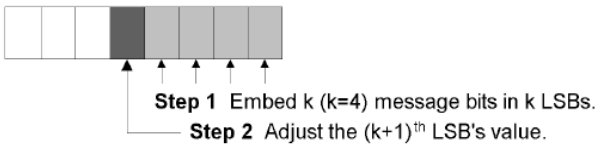
$$K = round\ (|log2\ V|)$$

dimana :

K = kapasitas penyisipan pada piksel P dalam bit.
V = variasi warna
round = fungsi matematika untuk pembulatan

d. MER (*Minimum Error Replacement*)

Tahap ini berfungsi untuk meminimalkan terjadinya perubahan piksel pada citra penampung akibat dari proses penyisipan. Proses MER dilakukan dengan mengubah nilai bit ke K+1 pada piksel P. Perubahan ini akan terjadi pada salah satu dari ke-tiga komponen warna (R, G atau B) yang terpilih [CHL-00].



Gambar 2 Langkah pada MER [CHL-00].

Bila pada langkah sebelumnya (penyisaipan pesan) didapat K = 4, maka bit yang ke-lima akan diubah nilainya, misal nilai awal adalah 1, maka akan diubah menjadi 0, begitu juga sebaliknya. Namun demikian pengubahan bit ke *K+1* belum tentu dilakukan, karena pada tahap MER juga dilakukan proses pengecekan nilai *embedding error*. *Embedding error* (Er) adalah selisih nilai (dalam desimal) pada komponen warna yang terpilih di piksel P, sebelum (*original*) dan sesudah dilakukan proses penyisipan, atau dengan rumus seperti di bawah ini :

$$Er = Abs\ [P(x,y) - P'(x,y)]$$

dimana :

Abs = Nilai absolut
Er = Nilai embedding error
P(x,y) = Piksel P asli

P'(x,y) = Piksel P yang telah dimodifikasi

Pengubahan pada bit ke K+1 akan dilakukan apabila nilai *embedding error* memenuhi syarat pada saat pengecekan, uraiannya bisa dijelaskan sebagai berikut. Asumsi *P(x,y)* adalah piksel P original, *P'(x,y)* adalah piksel P yang telah disisipkan sejumlah K-bit tanpa mengubah bit ke *K+1* dan *P"(x,y)* adalah piksel P yang telah disisipkan sejumlah K-bit sekaligus mengubah bit ke *K+1. Minimum error* yang dapat terjadi di piksel P haruslah *P'(x,y)* atau *P"(x,y)* [CHL-00]. Kemudian pengecekan nilai *embedding error* dilakukan lewat rumus sebagai berikut:

$$Er1 = Abs\,[P(x,y) - P'(x,y)]$$
$$Er2 = Abs\,[P(x,y) - P"(x,y)]$$

Apabila *Er1* < *Er2*, maka *P'(x,y)* yang akan menggantikan *P(x,y)*. Jika sebaliknya maka *P"(x,y)* yang akan menggantikan *P(x,y)* [CHL-00].

e. Mekanisme Penyisipan Pesan (*Embedded*)

Dalam proses penyisipan bit-bit pesan ke bit-bit piksel citra, menggunakan operator logika and dan or. Contoh: 1 and 1 = 1, 1 and 0 = 0, 0 and 0 = 0, 1 or 1 = 1, 1 or 0 = 1, 0 or 0 = 0. Dengan mengeset penyisipan 4 bit per kanal warna, maka dalam proses penyisipan diperlukan suatu bilangan yang berfungsi mempertahankan 4 bit MSB (*Most Significant Bits*) nilai intensitas kanal warna pada suatu piksel citra karena hanya akan merubah 4 bit LSB-nya, bilangan yang dimaksud adalah F0 (hexadecimal) = 240 (decimal) = 11110000 (binary). Sedangkan untuk mempertahankan bit-bit pesan, digunakan bilangan 0F = 15 = 00001111. Contoh ilustrasi : suatu citra dengan ukuran 2 x 2 piksel (Gambar 3), akan disisipkan karakter 'A' = 01000001 (ASCII). Sehingga menghasilkan perubahan intensitas piksel seperti pada Gambar 4.

f. Mekanisme Ekstraksi Pesan (*Extract*)

Pengektrakan (*extract*) adalah memperoleh kembali pesan yang disisipkan di citra stego (*stego image*). Pengekstrakan dapat dilakukan dengan cara mengambil kembali 4 bit LSB-nya sesuai kapasitas tiap kanal warnanya untuk memperoleh bit-bit pesan yang disisipkan. Setelah bit-bit pesan diperoleh lalu ubah bit-bit tersebut menjadi karakter kembali. Contoh : Mengekstrak atau mengambil kembali karakter yang disisipkan ke Gambar 4. Ilustrasi sebagai berikut:

Hasil1 pada kanal warna *Blue* pada Piksel 1 = 00110001
Hasil2 pada kanal warna *Green* pada Piksel 1 = 01010100
Langkah 1 :
(00110001 and 00001111) = 00000001
hasil1 F eks1
Langkah 2 :
(01010100 and 00001111) = 00000100
hasil2 F eks2
geser 4 bit LSB eks2 kekiri sehingga menjadi 01000000
(00000001 or 01000000) = 01000001
eks1 eks2 setelah digeser 'A'

Sebelum melakukan pengekstrakan pesan (message file), terlebih dahulu dilakukan validasi atau pembandingan password yang diberikan user pada saat akan melakukan pengekstrakan dengan password yang ada saat penyisipan, dimana password tersebut berfungsi untuk membangkitkan PRNG untuk proses pengekstrakan. Sehingga hanya user saja yang dapat melakukan pengekstrakan.

Piksel 1 Blue Green Red 00110011 01011101 01000110	Piksel 2 Blue Green Red 00110111 01011101 01000100
Piksel 3 Blue Green Red 00110010 01011100 01001110	Piksel 4 Blue Green Red 00110011 01011111 01100110

Gambar 3 Ilustrasi piksel citra-*cover*

Piksel 1 Blue Green Red 00110001 01010100 01000110	Piksel 2 Blue Green Red 00110111 01011101 01000100
Piksel 3 Blue Green Red 00110010 01011100 01001110	Piksel 4 Blue Green Red 00110011 01011111 01100110

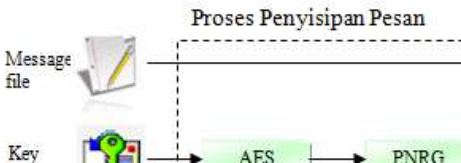
Gambar 4 Ilustrasi piksel citra-*stego*

g. Rumusan Parameter MSE (*Mean Squared Error*) dan PSNR (*Peak Signal-To-Noise Ratio*)

Distrorsi/gangguan pada sinyal dapat diukur dengan menggunakan rumus MSE (*Mean Square Error*) dan PSNR (*Peak Signal-to-Noise Ratio*) [HAK-10]

Rumus untuk MSE:

$$MSE = \frac{1}{m * n} \sum_{i=0}^{m} \sum_{j=0}^{n} (A_{ij} - B_{ij})^2 \quad (1)$$



Gambar 5 Rancangan Aplikasi *Steganography*, Sumber: Peracangan

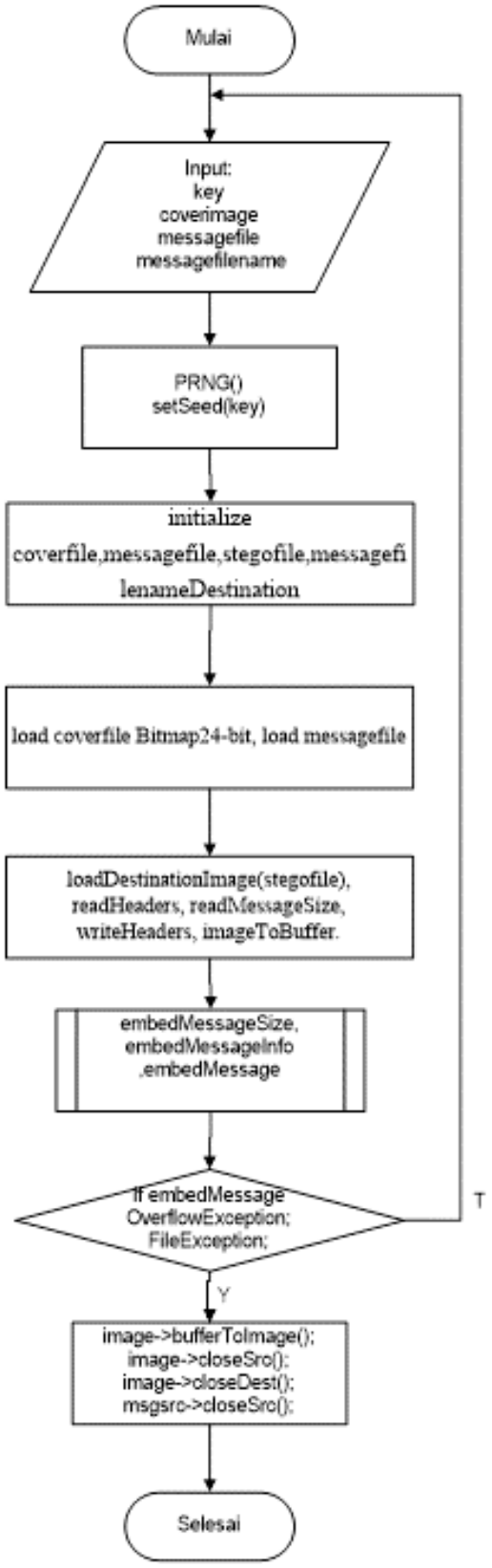
sedangkan rumus untuk PSNR:

$$PSNR = 10 * \log_{10} \frac{(Max)^2}{\frac{1}{m * n} \sum_{i=0}^{m} \sum_{j=0}^{n} (A_{ij} - B_{ij})^2} \quad (2)$$

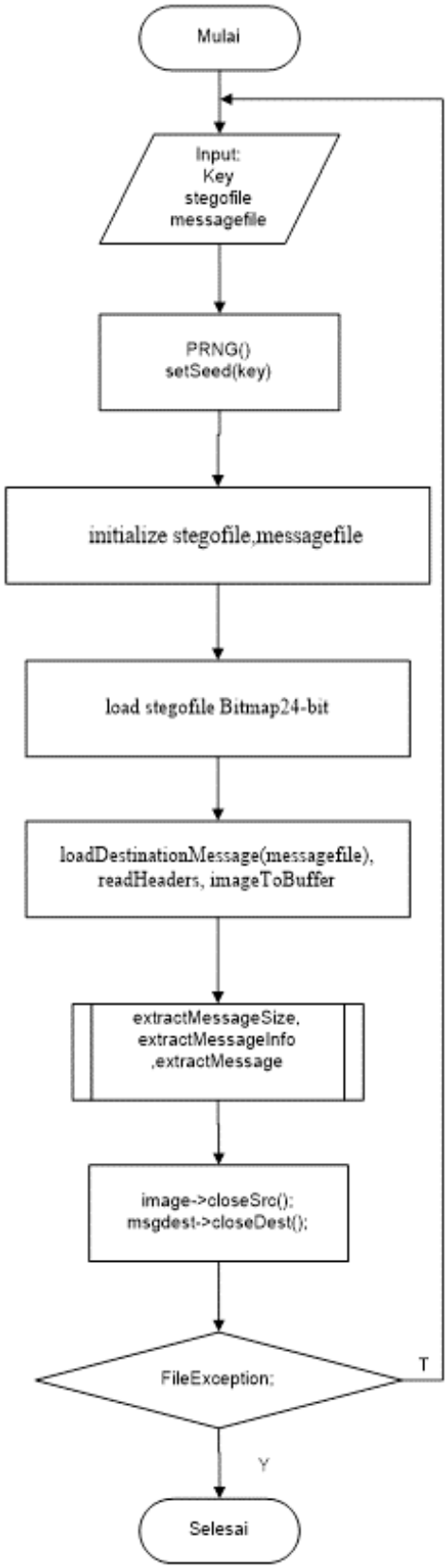
dimana:
MSE = *Mean Square Error*
PSNR = *Peak Signal-to-Noise Ratio* (dB)
Ai,j = *cover image*
Bi,j = *stego image*

3. Metodologi

Pada umumnya prinsip kerja sistem aplikasi steganography pada citra ini sama seperti halnya aplikasi steganography lain yang telah ada, hanya saja pada sistem aplikasi yang dibangun ini dengan menggunakan metode *Minimmum Error Least Significant Bit Replacement* (MELSBR) yang digunakan untuk penyembunyian pesan rahasia pada aplikasi ini adalah metode penyisipan pesan ke dalam bit rendah dari pixel yang menyusun file citra BMP 24 bit dengan AES.



Gambar 6 Diagram Alir Algoritma Penyisipan Pesan MELSBR



Gambar 7 Diagram Alir Algoritma Ekstraksi Pesan MELSBR

Dari gambar di atas dapat dilihat bahwa pada rancangan aplikasi *steganography* pada Gambar 5, terdapat proses untuk penyisipan pesan kedalam suatu citra asli (*cover image*) yang menghasilkan suatu citra yang telah disisipi oleh pesan (*stego image*), kemudian terdapat pula proses untuk mengekstraksi pesan dari citra yang telah disisipi pesan tersebut agar pesan dapat dibaca kembali.

Dalam proses penyisipan pesan diatas, nantinya user akan diberi kebebasan untuk memilih *key* AES dalam bentuk passpharase (karakter *password*), memilih pesan atau memilih citranya terlebih dahulu. Karena satu byte warna hanya dapat menyimpan empat bit data, maka jika user memilih citra (*cover image*) terlebih dahulu, ukuran file pesan akan dibatasi sesuai ukuran citra yang dipilih. Sedangkan jika user memilih pesan terlebih dahulu, maka akan ditentukan ukuran citra minimum yang dapat menampung data.

Gambaran umum langkah untuk penyisipan dan ekstraksi steganografi digital dengan metode MELSBR adalah sebagaimana pada Gambar 6 dan 7.

4. Pengujian dan Analisis

a. Metode pengujian MSE dan PSNR

Dalam pengujian ini digunakan aplikasi MATLAB, dengan pengujian MSE dan PSNR.

Syntax persamaan (1) pada MATLAB adalah sebagai berikut:

```
mse=mean((ci(:)-si(:)).^2);
```

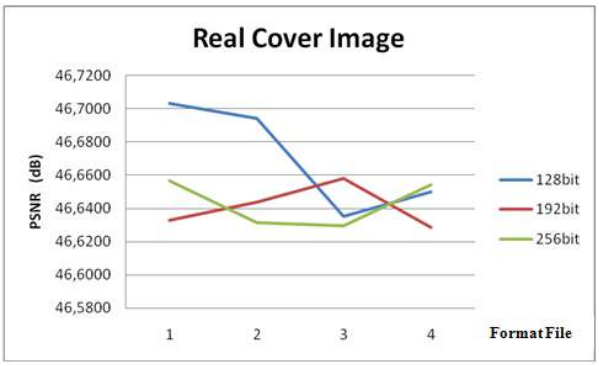
dimana `ci(:)` adalah *cover image* dan `si(:)` adalah *stego image*.

Syntax persamaan (2) pada MATLAB adalah sebagai berikut:

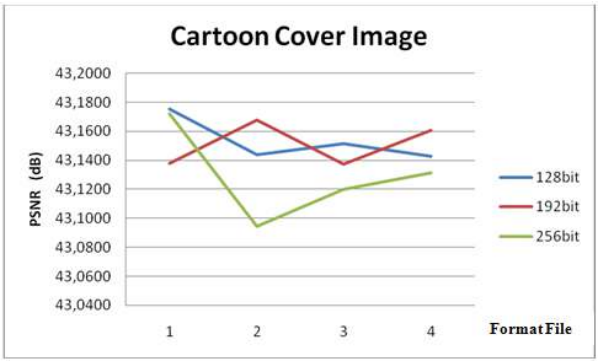
```
psnr=10*log10((255)^2/mse);
```

b. PSNR untuk *real cover image* terhadap panjang *key*

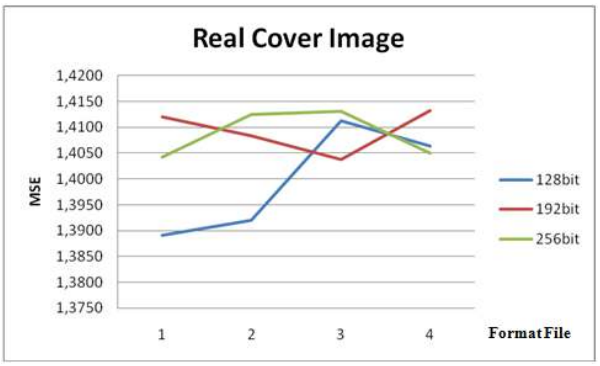
Pada Gambar 8 dapat dilihat bahwa perubahan yang terjadi pada nilai PSNR dalam decibel (dB) tidak berubah secara significant dengan nilai PSNR tertinggi 46,7034 dB (key_length 128 bit) dan PSNR terendah 46,6294 dB (key_length 256 bit). Untuk panjang key (key_length) 128 bit hanya berubah pada kisaran 0,0333 ; panjang key (key_length) 192 bit hanya berubah pada kisaran 0,0131 ;dan panjang key (key_length) 256 bit hanya berubah pada kisaran 0,0145.



Gambar 8 Perbandingan nilai PSNR terhadap panjang key pada *real cover image*



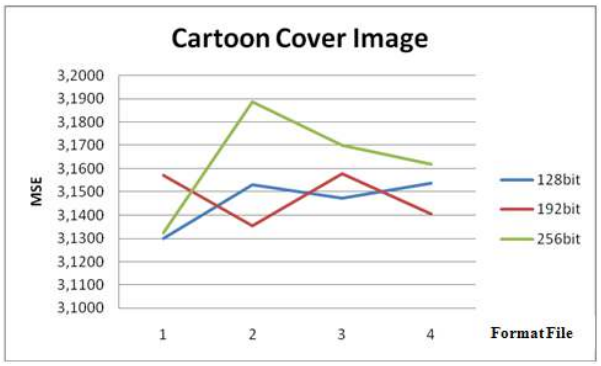
Gambar 9 Perbandingan nilai PSNR terhadap panjang key pada *cartoon cover image*



Gambar 10 Perbandingan nilai MSE terhadap panjang key pada *real cover image*

c. PSNR untuk *cartoon cover image* terhadap panjang *key*

Pada Gambar 9 dapat dilihat bahwa perubahan yang terjadi pada nilai PSNR dalam decibel (dB) tidak berubah secara significant dengan nilai PSNR tertinggi 43,1754 dB (key_length 128 bit) dan PSNR terendah 43,0946 dB (key_length 256 bit).



Gambar 11 Perbandingan nilai MSE terhadap panjang key pada *cartoon cover image*

Untuk panjang key (key_length) 128 bit hanya berubah pada kisaran 0,0153 ; panjang key (key_length) 192 bit hanya berubah pada kisaran 0,0157 ;dan panjang key (key_length) 256 bit hanya berubah pada kisaran 0,0324.

d. **MSE untuk *real cover image* terhadap panjang *key***

Pada Gambar 10 dapat dilihat bahwa perubahan yang terjadi pada nilai MSE tidak berubah secara significant dengan nilai MSE tertinggi 1,4132 (key_length 192 bit) dan MSE terendah 1,3891 (key_length 128 bit). Untuk panjang key (key_length) 128 bit hanya berubah pada kisaran 0,0108 ; panjang key (key_length) 192 bit hanya berubah pada kisaran 0,0043 ;dan panjang key (key_length) 256 bit hanya berubah pada kisaran 0,0047.

e. **MSE untuk *cartoon cover image* terhadap panjang *key***

Pada Gambar 11 dapat dilihat bahwa perubahan yang terjadi pada nilai MSE tidak berubah secara significant dengan nilai MSE tertinggi 3,1888 (key_length 256 bit) dan MSE terendah 3,1299 (key_length 128 bit). Untuk panjang key (key_length) 128 bit hanya berubah pada kisaran 0,0111 ; panjang key (key_length) 192 bit hanya berubah pada kisaran 0,0114 ;dan panjang key (key_length) 256 bit hanya berubah pada kisaran 0,0236.

5. Kesimpulan

1. Aplikasi penyembunyian pesan ini dapat digunakan untuk melakukan penyembunyian pesan pada citra digital dengan metode penyisipan Minimum Error Least Significant Bit Replacement-Advanced Encryption Standard (MELSBR-AES).
2. Pesan hasil ekstraksi yang diperoleh dari stego image tidak menimbulkan perubahan terhadap pesan asli sebelum disisipkan.
3. Nilai MSE dikatakan baik atau identik dengan citra asli (cover image) jika nilai MSE nya sama atau mendekati 0. Nilai rata-rata MSE pada pengujian real cover image 1,4059. Sedangkan nilai rata-rata MSE pada pengujian cartoon cover image 3,1523. Hal ini menunjukan bahwa penggunaan real cover image lebih baik dibandingkan dengan cartoon cover image dilihat dari nilai rata-rata MSE.
4. Semakin panjang key (key_length) yang digunakan tidak berpengaruh secara significant terhadap perubahan PSNR, tetapi panjang key (key_length) berpengaruh terhadap probabilitas pemecahan key 128, 192 dan 256 bit.
5. Besar ukuran file message yang disisipkan harus lebih kecil dari ukuran cover image (maksimal 10%).
6. Besar ukuran file yang disisipkan dengan key AES 128, 192 dan 256 tidak mempengaruhi tampilan visual citra (stego image) jika dilihat oleh mata.

6. Daftar Pustaka

[CHL-00] Chen, Ling-Hwei and Lee, Yeuan-Kuen.2000. An Adaptive Image Steganographic Model Based on Minimum-Error LSB Replacement. http://front.cc.nctu.edu.tw/6322.html.

[GMD-03] Gan, M. D.2003.Chameleon Image Steganography. http://chameleonstego.tripod.com/downloads/Chameleon Technical_Paper.pdf.

[JNS-06] Johnson, N. F& Sushil Jajodia. 2006. Steganalysis of Images Created Using Current Steganography Software. http://www.jjtc.com/ihws98/jjgmu.html.

[SED-96] Sellars, Duncan.1996. An Introduction to Steganography. http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html

[SUB-02] Sukmawan, Budi. 2002. Steganografi. http://bdg.centrin.net.id/~budskman/steganografi.htm.

[HAK-10] Hmood, Ali K. 2010. On the accuracy of hiding information metrics: Counterfeit protection for education and important certificates. http://www.budi.insan.co.id/courses/ec5010/projects/wihartantyo-report.doc.