

Metode Steganografi Penyisipan Karakter dengan Teknik LSB dan Penempatan Bit mengikuti Langkah Kuda Catur (*L-Shape*)

Charits Muntachib^{1,*}, Ratri Dwi Atmaja¹, Bambang Hidayat¹

¹ S1 Teknik Telekomunikasi, Universitas Telkom

Jl. Telekomunikasi Dayeuh Kolot Bandung

* E-mail : charitsmuntachib@yahoo.com

Abstrak. Semakin pesatnya perkembangan teknologi sebagai sarana pertukaran informasi, harus diimbangi pula dengan keamanan yang dapat menjamin terjaganya informasi yang ada di dalamnya. Pada khususnya *image steganography* dibutuhkan metode yang baik agar citra yang telah disisipi tidak memiliki perbedaan yang signifikan dengan citra *cover* asalnya dan memiliki nilai parameter kualitas citra yang baik dari segi penglihatan manusia yang dilihat dari nilai MOS (*Most Opinion Score*), SNR (*Signal to Noise Ratio*), RMSE (*Root Mean Square Error*), SSIM (*Structural Similarities*), BER (*Bit Error Rate*) dan CER (*Character Error Rate*) untuk pesan berupa teks. Untuk memenuhi kebutuhan tersebut dibuatlah sebuah metode penyisipan pesan dalam hal ini bit-bit karakter pada citra pada domain spasial dengan letak piksel penyisipan bit-bit menyesuaikan langkah kuda (*knight movement / L-Shape*) pada permainan catur dan posisi penyisipan bit mengadopsi metode yang sudah ada sebelumnya yaitu metode *LSB (Least Significant Bits)*. Penggunaan kunci (*key*) pada algoritma penyisipan dapat meningkatkan jumlah kemungkinan piksel penempatan untuk satu buah bit informasi. Pengujian dilakukan dengan penyisipan pesan teks ke dalam 3 citra *Carrier* berukuran 800 x 600 piksel dengan karakteristik yang berbeda. Dari hasil simulasi penyisipan teks dalam citra dengan 3 *key* yang telah dibuat, didapatkan bahwa nilai MOS sempurna didapatkan untuk semua *key*. SNR, RMSE dan SSIM terbaik dihasilkan oleh *key* = 1 dan dari segi keamanan, nilai *key* yang terbaik adalah untuk *key* = 3 dengan nilai probabilitas kemungkinan bit-bit dapat terbaca kecil dan BER terbaik. Dari nilai parameter yang dihasilkan, penggunaan *key* pada pemetaan *L-Shape* efektif untuk meningkatkan keamanan pesan yang disisipkan, dengan *trade-off* pada sisi kapasitas informasi *Carrier* ketika digunakan *key* yang lebih besar.

Kata Kunci: *Chess Knight Movement, LSB, L-Shape, BER, Steganografi*

1. Pendahuluan

Semakin pesatnya perkembangan teknologi sebagai sarana pertukaran informasi, harus diimbangi pula dengan keamanan yang dapat menjamin terjaganya informasi yang ada di dalamnya. Pada khususnya *image steganography* dibutuhkan metode yang baik agar citra yang telah disisipi tidak memiliki perbedaan yang signifikan dengan citra *cover* asalnya dan memiliki nilai parameter yang baik dari segi visual maupun keamanan informasi yang dilihat dari nilai MOS, SNR, RMSE, SSIM, BER dan CER untuk pesan berupa teks. Untuk memenuhi kebutuhan tersebut dibuatlah sebuah metode penyisipan pesan dalam hal ini bit-bit karakter pada citra pada domain *spatial* dengan letak piksel penyisipan bit-bit menyesuaikan langkah kuda (*knight movement / L-Shape*) pada permainan catur dan posisi penyisipan bit mengadopsi metode metode *LSB (Least Significant Bits)*. Penggunaan kunci (*key*) pada algoritma penyisipan dapat meningkatkan jumlah kemungkinan piksel penempatan untuk satu buah bit informasi. Pengujian dilakukan dengan penyisipan pesan teks ke dalam 3 citra *Carrier* berbeda berukuran 800 x 600 piksel dengan karakteristik yang berbeda dalam hal jumlah *peak* pada histogram citra. Nilai parameter hasil simulasi nantinya dianalisis untuk mendapatkan *key* yang optimal untuk serta kesimpulan lain yang dapat diambil dari sisi kualitas citra *Stego-image* maupun sisi keamanan informasi yang disisipkan.

2. Dasar Teori

2.1 Steganografi

Steganografi adalah sebuah seni dan ilmu tentang komunikasi yang tak terlihat. Hal ini dilakukan dengan menyembunyikan informasi yang dikirimkan dalam informasi yang lain atau *Carrier*-nya, dengan kata lain menyembunyikan keberadaan dari informasi yang dikomunikasikan tersebut. Kata steganografi berasal dari Bahasa Yunani, “*stegos*” yang berarti *cover* dan “*grafia*” yang berarti tulisan [1].

2.2 LSB (Least Significant Bit)

Teknik LSB merupakan teknik yang paling umum untuk steganografi pada citra digital sebagai *cover*-nya dan hanya butuh pendekatan yang sederhana untuk penyisipan informasi. Bit terakhir piksel (bit ke-8) akan diganti dengan bit pesan yang akan disembunyikan. Jika menggunakan citra RGB 24-bit, maka bit pesan dapat disisipkan pada bit terakhir piksel pada ketiga layer (*Red*, *Green* dan *Blue*) sehingga tiap piksel dapat disisipi 3 bit pesan [2].

2.3 RMSE (Root Mean Square Error)

Nilai RMSE adalah hasil akar dari nilai MSE yang dapat menunjukkan besar error di setiap piksel pada citra keluaran atau *Stego-image* yang dihasilkan

$$RMSE = \sqrt{\frac{\sum_{J=1}^M \sum_{K=1}^N (C_{(J,K)} - S_{(J,K)})^2}{M \cdot N}} \quad (1)$$

dimana : M, N adalah ukuran dari citra.

2.4 SSIM (Structural Similarities)

Pengukuran nilai SSIM adalah sebuah cara untuk mengukur tingkat kemiripan antar dua citra (*cover* dan *Stego-image*) dari parameter statistic [3]. Nilai SSIM yang dihasilkan mencerminkan kemiripan kedua citra dan merupakan hasil pengembangan teknik pengukuran kualitas citra sebelumnya. Berikut adalah persamaan untuk mengukur nilai kemiripan 2 citra (x dan y):

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (2)$$

2.5 SNR (Signal to Noise Ratio)

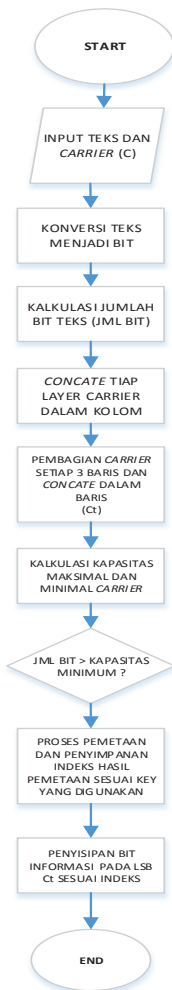
SNR pada umumnya merupakan perbandingan antara daya sinyal dengan daya noise yang dihasilkan dari sebuah sistem yang dilalui sinyanya tersebut. Semakin besar nilai SNR, menunjukkan bahwa kualitas sinyal keluaran semakin baik dan noise yang datang dapat diabaikan. Nilai SNR dapat didapatkan dengan membagi total daya yang masuk dalam sistem dengan daya noise atau hasil pengurangan total daya keluaran sistem dan daya total masukan dalam sistem [3].

2.6 BER (Bit Error Rate)

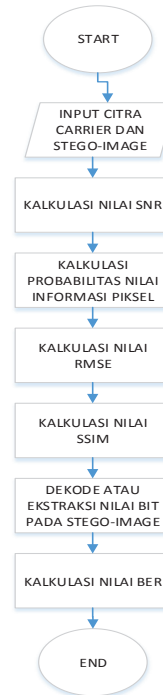
BER adalah nilai yang menunjukkan jumlah kesalahan bit informasi yang diterima dari jumlah total data yang dikirimkan atau masuk dalam sebuah sistem. Semakin kecil kesalahan yang diakibatkan oleh sistem, semakin baik pula kualitas sistem yang dilalui atau kualitas informasi yang dikirimkan.

3. Perancangan Sistem

3.1 Diagram Alir Sistem



Gambar 1. Diagram Alir Penyisipan



Gambar 2. Diagram Alir Pengukuran Nilai Performansi

3.2 Proses Penyisipan

Berikut merupakan penjelasan untuk proses penyisipan :

1. Input program berupa teks informasi dalam bentuk file *.txt* dan *Carrier* berupa citra digital berukuran 800 x 600 piksel.
2. Dilakukan konversi pesan teks input menjadi bit dari nilai ASCII setiap karakter pesan yang diinputkan.
3. Dilakukan kalkulasi jumlah bit hasil konversi pesan teks input.
4. Proses selanjutnya adalah *concat* atau penyambungan setiap *layer* citra input dalam baris, sehingga jumlah baris menjadi tiga kali jumlah baris semula.
5. Dilakukan pembagian hasil *concat* poin 4 di atas menjadi setiap 3/6/9 baris yang menentukan jumlah baris dalam 1 blok. Setiap bagian dilakukan proses *concat* dalam kolom sehingga jumlah kolom menjadi jumlah kolom awal dikalikan dengan jumlah blok. (Ct)
6. Dilakukan kalkulasi kapasitas maksimal dan kapasitas minimal dari *Carrier*.
7. Proses selanjutnya adalah membandingkan nilai jumlah bit dan kapasitas minimum dari hasil kalkulasi poin 6. Jika jumlah bit lebih kecil atau sama dengan kapasitas minimum maka dilakukan proses pemetaan dan pencarian indeks pemetaan pada domain spatial citra *Carrier*. Setiap *key* yang digunakan memiliki algoritma yang berbeda (algoritma terlampir).
8. Dilakukan proses penyisipan bit informasi dalam LSB indeks hasil pemetaan.

3.3 Proses Pengukuran Performansi

Berikut merupakan penjelasan untuk proses pengukuran performansi :

1. Input program berupa citra asli sebelum disisipi dan *Stego-image* yang berisi pesan informasi.
2. Dilakukan kalkulasi Nilai SNR, MSE, RMSE, SSIM maupun BER dari *Stego-image* dengan persamaan yang telah disebutkan pada tinjauan pustaka.
3. Selain pengukuran yang disebutkan di atas, dilakukan pula pengukuran MOS (*Most Opinion Score*) yang merepresentasikan kualitas kemiripan visual kasat mata manusia pada *Stego-image* melalui kuesioner yang disebarakan melalui media online.
4. Nilai nilai tersebut pada bagian selanjutnya akan dianalisis untuk mendapatkan nilai kunci yang optimal.

4. Hasil Pengujian dan Analisis

4.1 SNR

Nilai SNR dari setiap *key* untuk setiap *Carrier* hampir sama dan dapat dikatakan nilai *key* tidak berpengaruh pada nilai SNR kecuali pada nilai *key*=1 yang selalu memberikan nilai SNR terbesar. Hasil tersebut menunjukkan bahwa penyisipan bit yang tersebar lebih luas akan menghasilkan *noise* yang lebih besar sebagaimana distribusi penyisipan bit yang sebagian akan menjadi *noise* ketika nilai LSB piksel tempat penyisipan berubah dari nilai LSB semula.

Tabel 1. Hasil SNR

No	Carrier	Key	SNR (dB)
1	Flower.jpg	1	46.282000654627126
2	Flower.jpg	2	44.461650524945560
3	Flower.jpg	3	44.531394516399350
4	Sand.jpg	1	44.749679189762000
5	Sand.jpg	2	44.706296139510485
6	Sand.jpg	3	44.673555843864250
7	Candle.jpg	1	38.491785679510880
8	Candle.jpg	2	37.771061195935620
9	Candle.jpg	3	37.863792498921605

4.2 RMSE dan SSIM

Dari data Hasil RMSE dan SSIM dapat dilihat bahwa untuk nilai RMSE hampir sama pada setiap *key* yang digunakan. Tidak ada selisih yang cukup signifikan diantara ketiga *key* untuk setiap *Carrier* yang digunakan proses kalkulasi nilai RMSE, tidak memandang bagaimana distribusi *error* tersebar di dalam citra *Stego-image*. Berbeda dengan RMSE, tingkat kemiripan dua buah citra dengan SSIM tidak menggunakan *error* setiap piksel melainkan dari faktor nilai pengukuran statistik setiap citra, sehingga citra yang memiliki pola nilai piksel berbentuk (*pattern*) seperti Flower.jpg dan Candle.jpg akan dihasilkan nilai SSIM yang lebih besar.

Tabel 2. Hasil RMSE dan SSIM

No	Carrier	Key	RMSE	SSIM
1	Flower.jpg	1	0.075906558053673	0.986888574217531
2	Flower.jpg	2	0.093604516747620	0.986870702431353
3	Flower.jpg	3	0.092855921847894	0.986864291991179
4	Sand.jpg	1	0.093485738662821	0.980850256157849
5	Sand.jpg	2	0.092878355330448	0.980843657474934
6	Sand.jpg	3	0.093229108472980	0.980842286582338
7	Candle.jpg	1	0.094247752699409	0.985164630276559
8	Candle.jpg	2	0.102401714395371	0.985191298172717
9	Candle.jpg	3	0.101314280006983	0.985203252861815

4.3 MOS (Most Opinion Score)

Dari hasil yang didapatkan, secara visual setiap *Carrier* dengan karakteristiknya maupun setiap *key* yang digunakan tidak mempengaruhi penampakan citra *Stego-image* secara visual manusia.

Tabel 3. Hasil MOS

No	Carrier	Key	MOS
1	Flower.jpg	1	5
2	Flower.jpg	2	5
3	Flower.jpg	3	5
4	Sand.jpg	1	5
5	Sand.jpg	2	5
6	Sand.jpg	3	5
7	Candle.jpg	1	5
8	Candle.jpg	2	5
9	Candle.jpg	3	5

4.3 BER

Berikut ini adalah hasil kalkulasi BER dan CER dari *Stego-image* yang dihasilkan dari algoritma yang digunakan atau tanpa noise dan hasil setelah penambahan *noise* gaussian dengan variasi variasi :

Tabel 4. Hasil BER dan CER Tanpa *Noise*

No	Carrier	Key	BER	CER
1	Flower.jpg	1	0	0
2	Flower.jpg	2	0	0
3	Flower.jpg	3	0	0
4	Sand.jpg	1	0	0
5	Sand.jpg	2	0	0
6	Sand.jpg	3	0	0
7	Candle.jpg	1	0	0
8	Candle.jpg	2	0	0
9	Candle.jpg	3	0	0

Tabel 5. Hasil BER untuk Penambahan *Noise* Gaussian

Carrier	Key	Noise Gaussian				
		v=0.01	v=0.02	v=0.03	v=0.04	v=0.05
Flower.jpg	1	0.50928	0.508	0.51168	0.51384	0.51508
Flower.jpg	2	0.49648	0.50768	0.51112	0.51232	0.51228
Flower.jpg	3	0.49568	0.49836	0.49676	0.50232	0.50824
Sand.jpg	1	0.50128	0.50332	0.50584	0.50544	0.50616
Sand.jpg	2	0.496	0.50116	0.50364	0.50456	0.50052
Sand.jpg	3	0.4936	0.49936	0.50072	0.50444	0.5038
Candle.jpg	1	0.4808	0.48648	0.4844	0.48132	0.48444
Candle.jpg	2	0.48916	0.48708	0.49116	0.48192	0.48628
Candle.jpg	3	0.48728	0.48864	0.48712	0.49144	0.48608

Tabel 6. Hasil CER untuk Penambahan *Noise* Gaussian

Carrier	Key	Variansi noise gaussian				
		v=0.01	v=0.02	v=0.03	v=0.04	v=0.05
flower.jpg	1	0.9795	0.9872	0.98976	0.98208	0.98208
flower.jpg	2	0.98208	0.95648	0.9744	0.97696	0.97184
flower.jpg	3	0.96928	0.96672	0.96416	0.97952	0.97696
sand.jpg	1	0.9968	0.99584	0.9952	0.99776	0.99616
sand.jpg	2	0.99776	0.99584	0.99648	0.99456	0.99744
sand.jpg	3	0.9968	0.99744	0.99616	0.99712	0.9968
candle.jpg	1	0.99488	0.99648	0.99616	0.99232	0.99488
candle.jpg	2	0.9968	0.99552	0.9952	0.99456	0.9952
candle.jpg	3	0.9952	0.99488	0.9952	0.99424	0.99712

Dari data di atas didapatkan bahwa setiap bit yang disisipkan dapat diterima di sisi penerima tanpa ada satupun bit informasi yang salah atau hilang untuk *Stego-image* tanpa penambahan noise yang menunjukkan algoritma ekstraksi bit informasi bekerja tanpa kesalahan. Pada penambahan noise, dari nilai BER yang didapatkan dapat terlihat bahwa semakin besar nilai *key* yang digunakan, semakin kecil pula tingkat kesalahan penerimaan bit-bit informasi dari *Stego-image*. Hal tersebut menunjukkan bahwa semakin luas persebaran bit informasi dalam *Stego-image* semakin tahan pula informasi di dalamnya terhadap *noise* gaussian. Nilai CER yang terdistribusi merata menunjukkan bahwa error LSB tiap piksel akibat noise adalah merata pada semua piksel sesuai dengan distribusi noise gaussian.

5. Kesimpulan

Dari analisis yang telah dilakukan didapatkan kesimpulan bahwa penggunaan *key*=1 paling optimal dari segi kualitas citra *Stego-image* dari nilai SNR, RMSE maupun SSIM, serta paling optimal dari segi kapasitas maksimal informasi yang dapat dimasukkan dalam *Carrier*, namun memiliki tingkat keamanan paling minimum jika dibandingkan dengan semua *key* yang digunakan dari nilai probabilitas informasi dalam setiap pikselnya serta BER yang dihasilkan. Disimpulkan pula bahwa *key*=2 memiliki tingkat keamanan berada di antara *key*=1 dan *key*=3 serta optimal digunakan dalam semua *Carrier* dari sisi keamanan. Nilai *key*=3 memiliki tingkat keamanan terbaik dari probabilitas informasi dalam setiap pikselnya serta BER yang dihasilkan baik dengan *noise* maupun tanpa *noise*. *Key* = 3 paling optimal digunakan untuk mengamankan informasi penting dengan ukuran yang tidak terlalu besar atau ketika digunakan citra *carrier* dengan resolusi yang besar.

Selain dari sisi *key* yang digunakan, dapat disimpulkan bahwa citra terbaik digunakan dari segi kualitas citra *Stego-image* yang dihasilkan adalah citra dengan banyak peak histogram (Flower.jpg). Citra terbaik dari segi kemampuan mempertahankan informasi dalam hal ini terhadap *noise* adalah citra dengan karakteristik seperti *Carrier* 3 (Candle.jpg). Nilai CER tidak bergantung pada *key* yang digunakan melainkan distribusi noise dalam *Stego-image*. Persepsi visual manusia secara umum tidak dapat membedakan citra *Carrier* dan *Stego-image* hasil penyisipan untuk metode *L-Shape* dari nilai MOS yang didapatkan. Secara keseluruhan, metode penyisipan dalam domain *spatial* dengan pemetaan bit *L-Shape* memiliki performansi yang baik dari segi kualitas citra keluaran maupun keamanan informasi yang disisipkan.

6. Daftar Referensi

- [1] Moerland, T., “*Steganography and Steganalysis*”, Leiden Institute of Advanced Computing Science, Leiden University, Leiden, 2003.
- [2] Dumitrescu, S., W. Xiaolin & Z. Wang “*Detection of LSB steganography via sample pair analysis*,” In: LNCS, Vol. 2578, Springer-Verlag, New York, pp: 355 -372, 2003
- [3] Stefan Katzenbeiser & Fabien A.P. Petitcolas, “*Information Hiding Techniques for Steganography and Digital Watermarking*”, Artech House, Computer Security series, Boston, London, 1999.