

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/338124748>

Analisis dan Perbandingan Teknik Steganografi Citra Digital Algoritma LSB dan DCT dengan menggunakan Algoritma Kriptografi RC4

Article · December 2019

CITATIONS

0

READS

45

2 authors:



Dewi Wulan Sari

Siliwangi University

2 PUBLICATIONS 0 CITATIONS

[SEE PROFILE](#)



Ihsan Pratama

Siliwangi University

2 PUBLICATIONS 0 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Analisis dan Perbandingan Teknik Steganografi Citra Digital Algoritma LSB dan DCT dengan menggunakan Algoritma Kriptografi RC4 [View project](#)



Implementasi Server Load Balancing Menggunakan Haproxy dengan Algoritma Roundrobin Serta Penerapan Cache File [View project](#)

Analisis dan Perbandingan Teknik Steganografi Citra Digital Algoritma LSB dan DCT dengan menggunakan Algoritma Kriptografi RC4

Dewi Wulan Sari
Department of Informatics
Siliwangi University
Tasikmalaya, Indonesia
177006021@student.unsil.ac.id

Ihsan Pratama
Department of Informatics
Siliwangi University
Tasikmalaya, Indonesia
177006094@student.unsil.ac.id

Abstract— Tingkat kejahatan pada data-data teknologi informasi semakin tinggi seiring dengan semakin canggihnya dunia teknologi informasi, sehingga data-data yang dimiliki atau dipertukarkan harus diamankan salah satunya dengan metode steganografi dan/atau kriptografi. Teknik merahasiakan atau menyembunyikan pesan rahasia ke dalam pesan lain agar keberadaan pesan yang disembunyikan tidak dapat diakses orang lain yang tidak berwenang disebut dengan teknik steganografi. Teknik steganografi berbeda dengan kriptografi. Pada kriptografi pesan yang disisipi pesan rahasia akan terlihat sangat berbeda dengan pesan yang tidak disisipi pesan rahasia, sehingga akan menimbulkan kecurigaan bagi pihak ketiga atau orang lain yang tidak memiliki wewenang untuk mengakses pesan. Penelitian ini bertujuan untuk membandingkan teknik penyembunyian data teks yang sudah dienkripsi oleh kriptografi algoritma RC4 (Rivest-Cipher 4) kedalam sebuah cover image menggunakan algoritma steganography LSB dan DCT. Adapun proses perbandingan kinerja dan indeks kualitas algoritma steganografi LSB dan DCT diukur menggunakan uji fidelity (kualitas citra digital tidak berubah), dan recovery (dokumen yang disembunyikan dalam citra digital harus dapat dibaca kembali). Hasil pengujian recovery menunjukkan algoritma DCT memiliki kecepatan waktu proses dan kompresi file yang lebih baik. Adapun hasil pengujian fidelity memperlihatkan metode LSB lebih baik dibandingkan dengan DCT.

Keywords—Steganografi, LSB, DCT, Kriptografi, RC4

I. INTRODUCTION

Teknologi informasi terus berkembang, memberikan pengaruh besar terhadap individu maupun organisasi yang bertujuan untuk memenuhi kebutuhan bagi pengguna [1]. Dewasa ini setiap orang dapat saling berkomunikasi tanpa lagi terbatas jarak dan waktu seiring dengan semakin luasnya perkembangan teknologi komunikasi dan informasi, sehingga menyebabkan peningkatan trafik pengguna komunikasi. Tingginya tingkat komunikasi dan luasnya akses menyebabkan adanya peningkatan ancaman terhadap penyadapan dan pencurian informasi. Penyadapan dan pencurian informasi tidak hanya dilakukan dengan membajak langsung pihak yang sedang berkomunikasi, namun juga dapat memindai data dan informasi yang mengalir pada jalur komunikasi. Ancaman terhadap pembajakan dan pencurian informasi tersebut dapat dikendalikan dengan menggunakan teknik keamanan seperti penyandian data yaitu kriptografi atau menggunakan penyembunyian informasi kedalam media lain yaitu steganografi atau kombinasi dari keduanya [2].

Kriptografi dilakukan dengan mengubah pesan yang dapat dipahami menjadi pesan yang tidak dapat dipahami dengan menggunakan suatu kunci tertentu. Sehingga, hanya pihak yang memiliki kunci untuk membuka dokumen tersebut saja

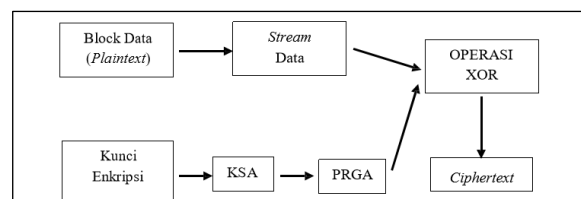
yang dapat memahami isi dari dokumen tersebut. Namun, kriptografi tidak dapat menyembunyikan keberadaan data, sehingga tidak dapat melindungi data dengan efektif. Penyadap dapat dengan mudah mendeteksi keberadaan data yang terenkripsi dan mencoba berbagai serangan untuk mendapatkan data aslinya. Dalam meningkatkan keamanan data, dibutuhkan pendekatan dengan dua lapis proteksi untuk memberikan keamanan yang lebih baik. Salah satu caranya adalah dengan menggunakan steganografi [3].

Steganografi adalah ilmu dan seni menyembunyikan komunikasi dan data penting. Steganografi digunakan untuk menyembunyikan informasi rahasia ke dalam suatu media sehingga keberadaan pesan tersebut tidak diketahui oleh orang lain. Steganografi bertujuan untuk menghilangkan kecurigaan dengan cara menyamarkan pesan tersebut.

Pada penelitian ini, penulis mengajukan teknik pengamanan pesan rahasia steganografi dengan keamanan berlapis, dengan menambahkan kriptografi RC4 terhadap pesan rahasia yang akan disisipkan kedalam citra digital kemudian pesan disisipkan kedalam citra digital melalui steganografi menggunakan metode LSB dan DCT. Pengamanan pesan dengan kombinasi kriptografi dan steganografi dapat mempersulit para kriptanalis dalam melakukan pencurian atau perusakan data teks. Selanjutnya, dalam penelitian ini akan dilakukan perbandingan hasil kualitas pengamanan dan penyembunyian pesan melalui steganografi LSB dengan DCT.

Rivest Cipher 4 (RC4)

RC4 merupakan salah satu algoritma kunci simetris yang berbentuk stream cipher, yaitu memproses unit atau input data pada satu saat. Disebut algoritma kriptografi simetrik karena menggunakan kunci yang sama untuk mengenkripsi ataupun mendekripsi suatu pesan, data, ataupun informasi. Unit atau data pada umumnya sebuah byte atau bit. RC4 pertama kali dibuat oleh Ron Rivest di Laboratorium RSA pada tahun 1987. RC4 merupakan enkripsi stream simetrik proprietary yang dibuat oleh RSA Data Security Inc (RSADSI) [4]. Berikut ini diagram enkripsi algoritma RC4:



Gambar 1 Diagram Enkripsi RC4

Keterangan:

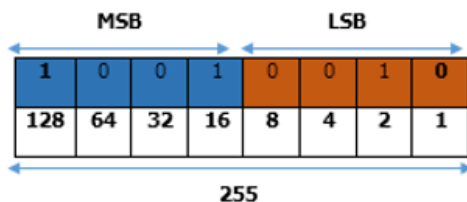
KSA = *Key Scheduling Algorithm*

PRGA = *Pseudo Random Generation Algorithm*

Penelitian ini menggunakan Kriptografi dengan algoritma RC4 untuk proses enkripsi pesan yang kemudian dikombinasikan dengan Steganografi dengan metode Least Significant Bit (LSB) serta DCT untuk menyisipkan pesan dalam bentuk biner pada bit terkecil pada file dengan format PNG.

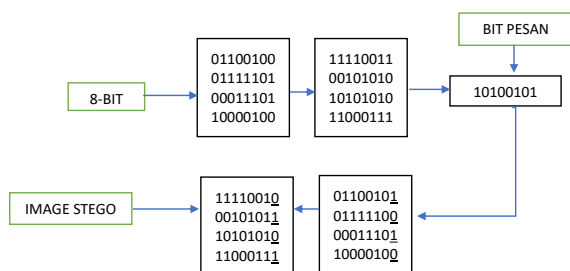
Least Significant Bit (LSB)

Least Significant Bit (LSB) merupakan metode steganografi yang paling sederhana dan mudah untuk diimplementasikan ke sebuah aplikasi. Metode ini menggunakan citra digital sebagai covertext. Pada susunan bit di dalam sebuah byte (1 byte = 8 bit), ada bit yang paling berarti (most significant bit atau MSB) dan bit yang paling kurang berarti (least significant bit atau LSB) [5], [6].



Gambar 2 Struktur LSB

Berikut cara kerja dari metode LSB (Least Significant Bit): byte 10010010 angka 1 (pertama dari kiri) adalah bit dari MSB (Most Significant Bit) dan angka 0 (pertama dari kanan) adalah LSB (Least Significant Bit). Bit yang cocok untuk digantikan adalah bit LSB, Karena perubahan dari LSB tersebut hanya mengubah nilai byte satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya, misalnya byte tersebut menyatakan warna orange, maka perubahan satu bit LSB (Least Significant Bit) tidak mengubah warna orange tersebut secara berarti. Mata manusia tidak dapat membedakan perubahan kecil [5].



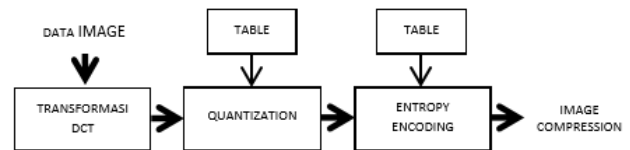
Gambar 3 Cover Image Selection

Penyisipan pesan ke dalam cover dinamakan encoding (embedded), sedangkan ekstraksi pesan dari stego dinamakan decoding (extraction) [7].

Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) yang mengubah data digital ke dalam bentuk domain frekuensi. Metode yang dilakukan dalam teknik transformasi DCT adalah memecah citra digital menjadi blok-blok kecil dengan ukuran yang tetap kemudian dikonversikan dari domain spatial menjadi domain frekuensi. Tahapan-tahapan pada metode DCT ini adalah data image diproses mulai dari tahap preparation

process, selanjutnya dilakukan transformasi DCT, quantization dan entropy encoding yang kemudian keluar sebagai image compression (hasil kompresi). Tahapan proses tersebut digambarkan sebagai berikut [8], [9] :



Gambar 4 Tahapan Proses DCT

1. Transformasi DCT : Transformasi akan semakin baik jika kemampuan mengompresi informasi dalam koefisien yang lebih sedikit pun semakin baik. Pada tahap ini dilakukan perubahan input data ke dalam format untuk mengurangi redundansi interpixel pada gambar masukan. Teknik perubahan pengkodean menggunakan reversibel, linier matematika transformasi untuk memetakan nilai piksel ke satu set koefisien, yang kemudian dikuantisasi dan dikodekan.
2. Quantization : Tahap kuantisasi dilakukan untuk membersihkan koefisien DCT yang tidak penting untuk proses pembentukan image baru.
3. Entropy Encoding yaitu proses penggunaan algoritma entropi, algoritma huffman digunakan untuk mengkodekan koefisien hasil proses DCT yang akan mengeliminasi nilai-nilai matriks yang bernilai nol dimana akan menghilangkan kelebihan dari keluaran kuantiser secara zigzag dan akhirnya akan diperoleh image yang telah direkonstruksi (image yang sudah dikompres).

II. RELATED WORK

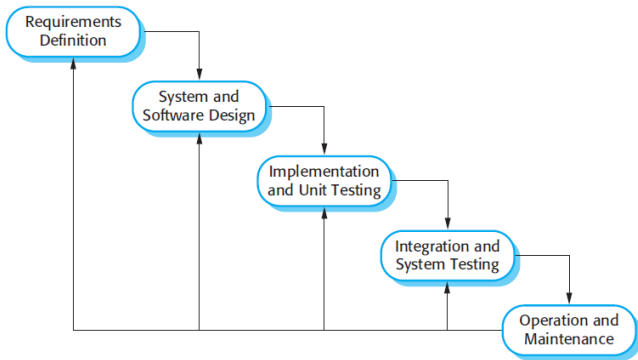
Pada penelitian [10], dilakukan analisis terhadap gambar steganografi yang menggunakan metode Discrete Cosine Transform. Hasil penelitian tersebut menunjukkan bahwa besar nilai rata-rata PSNR dari tiga gambar yang dilakukan penelitian ialah 37.44 dB. Selain itu, pesan yang terdapat dalam gambar steganografi DCT tahan kompresi gambar.

Penelitian lainnya [5], mengkombinasikan antara algoritma RC4 dengan metode LSB dengan menggunakan Visual Studio 2008. Hasil penelitian tersebut berupa waktu proses enkripsi dan dekripsi serta perubahan ukuran gambar setelah dienkripsi. Dari penelitian tersebut diketahui bahwa ukuran file mengalami perubahan setelah dilakukan penyisipan pesan, walaupun perubahan yang dihasilkan tidak signifikan. Terlihat perubahan file hanya 1 KB tapi sesungguhnya jika dilihat dalam ukuran bytes gambar, gambar yang sudah dienkripsi mengalami perubahan sekitar 200 bytes.

Analisis perbandingan steganografi LSB dan DCT dengan menggunakan algoritma kriptografi RC4 merupakan fokus utama penelitian ini. Tujuan utama penelitian ini ialah mengembangkan suatu aplikasi pengamanan data yang berlapis menggunakan kriptografi RC4 dan beberapa algoritma steganografi yaitu LSB dan DCT sebagai penyamaran keberadaan data, selanjutnya dilakukan perbandingan dan uji kualitas dari penerapan algoritma steganografi LSB dan DCT.

III. SYSTEM DESIGN

Metode pengembangan sistem yang digunakan adalah waterfall model yang melibatkan fase-fase sebagai berikut:



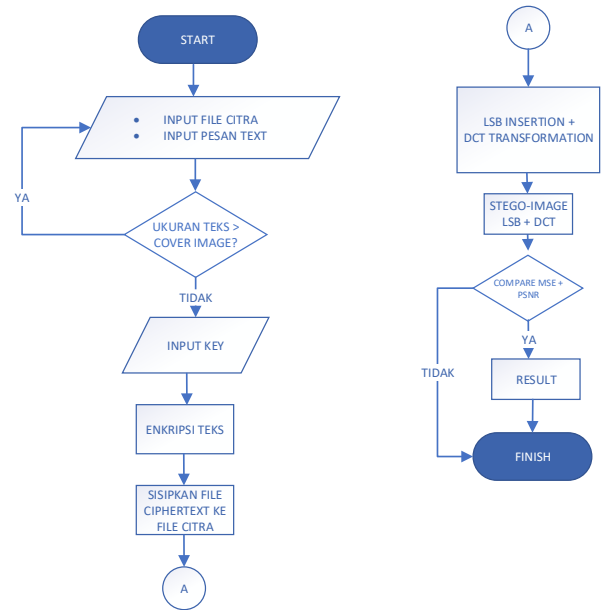
Gambar 5 Waterfall Model

Adapun spesifikasi software yang digunakan dalam pengembangan aplikasi ditampilkan pada Tabel I.

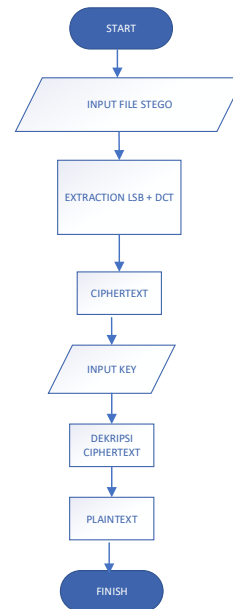
TABLE I. SOFTWARE SPECIFICATION

No	Software	Version
1	Visual Studio Code	1.41.0
2	Python	3.8.0
3	Pip	19.3.1
No	Requirements Python Library	Version
1	opencv-python	3.8.0
2	xlwt	1.3.0
3	numpy	1.17.4
4	matplotlib	3.1.2
5	pillow	6.1.2
6	pathlib	1.0.1
7	openapi-codec	1.3.2
8	cycler	0.10.0
9	decorator	4.0.11
10	networkx	1.11
11	olefile	0.44
12	pyparsing	2.2.0
13	python-dateutil	2.6.0
14	pytz	2017.2
15	PyWavelets	0.5.2
16	scikit-image	0.13.0
17	scipy	0.19.0
18	Six	1.10.0

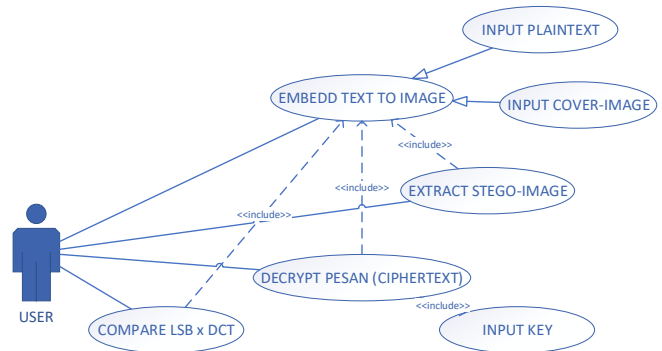
Tahapan berikutnya membuat rancangan umum aplikasi. Berikut ini flowchart proses dan use case aplikasi.



Gambar 6 Flowchart Embed dan Enkripsi Pesan Rahasia

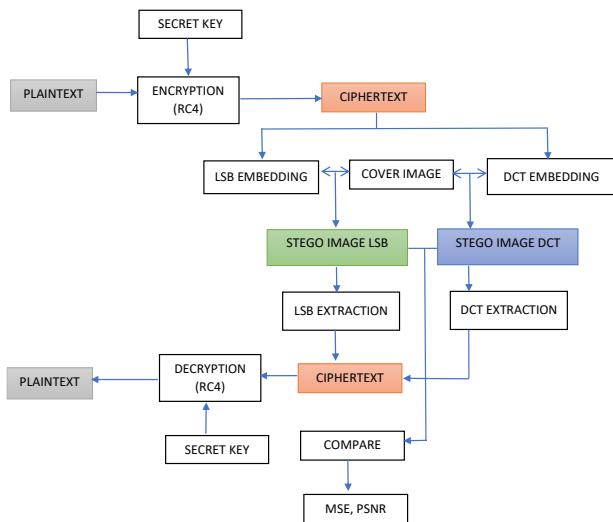


Gambar 7 Flowchart Extraksi dan Dekripsi Pesan Rahasia



Gambar 8 Use Case Aplikasi

Setelah rancangan selesai dibuat maka, tahap berikutnya dilakukan implementasi atau pengembangan aplikasi (coding). Hasil dari aktivitas ini menghasilkan aplikasi yang dapat mengenkripsi dan menyembunyikan pesan, serta memperlihatkan perbandingan kualitas hasil dari dua algoritma steganografi yakni LSB dan DCT. Berikut ini diagram alur proses kombinasi kriptografi dan steganografi yang dibuat.

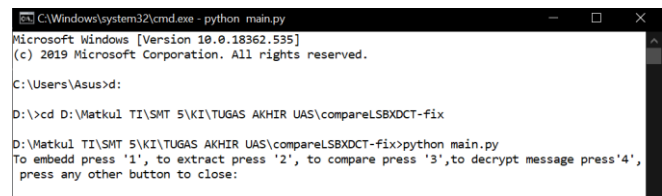


Gambar 9 Diagram Alur Proses Kombinasi Kriptografi dan Steganografi

Pengembangan aplikasi penelitian ini menggunakan Python 3.8.0. dan menggunakan PIL untuk transformasi domain Spasial yakni LSB, dan OpenCV untuk transformasi domain Frekuensi yakni DCT. Pesan rahasia dienkripsi kemudian disembunyikan ke dalam gambar yang disebut cover-image, dan menghasilkan stego-image (gambar yang mengandung pesan rahasia). Kemudian stego image di extract sehingga dapat mengungkap isi dari pesan yang telah dirahasiakan sebelumnya. Pesan yang muncul akan berbentuk ciphertext, karena pesan rahasia asli dienkripsi terlebih dahulu dengan algoritma RC4. Untuk mengetahui pesan rahasia harus dilakukan ekstraksi pesan terlebih dahulu. Perbandingan kualitas gambar menggunakan perbandingan MSE dan PSNR. Hasilnya disimpan ke dalam spreadsheet excel.

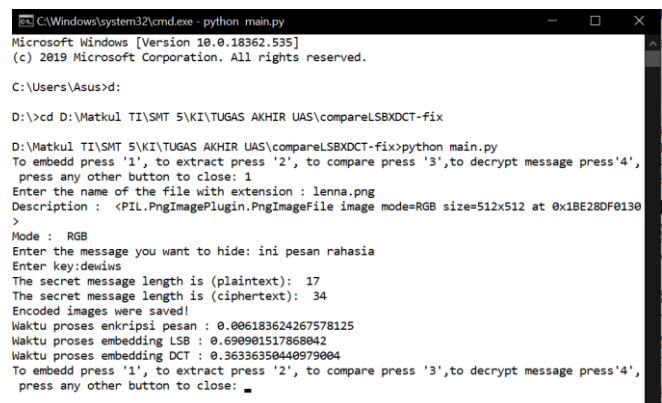
IV. RESULT AND ANALYSIS

Implementasi dan pengembangan aplikasi yang dibuat pada penelitian ini, memiliki 4 fitur utama yaitu fitur embed text to image, extract stego-image, decrypt pesan rahasia, dan compare atau membandingkan hasil stego image antara algoritma LSB dengan DCT. Aplikasi yang dibuat dijalankan pada command prompt dengan menjalankan perintah python main.py di dalam direktori folder aplikasi. Berikut ini tampilan awal aplikasi:



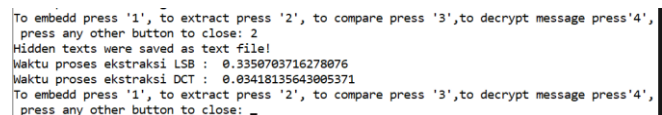
Gambar 10 tampilan awal

Untuk melakukan embedding pesan ke image, pilih 1 lalu enter, selanjutnya user diminta untuk menginputkan nama file gambar yang digunakan sebagai cover image yang telah disimpan di dalam folder Original_image, serta menginputkan pesan rahasia yang ingin disembunyikan beserta kunci rahasianya. Jika proses embedding berhasil, gambar stego dapat dilihat di dalam folder Encoded_image. Adapun jenis file gambar yang dapat diinputkan hanya yang memiliki ekstensi .png dengan mode RGB.



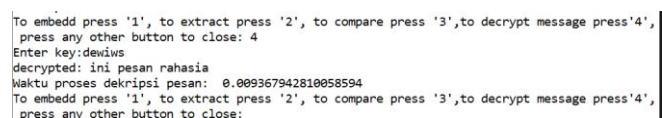
Gambar 11 embedding message

Untuk melakukan extract message dari stego image, pilih 2 lalu enter. Hasil text yang disembunyikan terdapat dalam folder Decoded_output. Adapun text yang di extract akan berbentuk ciphertext karena aplikasi yang dikembangkan menerapkan enkripsi algoritma RC4.



Gambar 12 extract message

Agar dapat mengetahui isi pesan sebenarnya, pilih 4 untuk melakukan decrypt message, kemudian masukkan kunci rahasianya.



Gambar 13 decrypt message

Adapun untuk membandingkan kualitas gambar stego algoritma LCB dan DCT yang dihasilkan, pilih 3 lalu enter. Maka hasil perbandingan kualitas keduanya yang berupa MSE dan PSNR terdapat pada file excel bernama Comparison.xlsx yang terdapat pada folder Comparison_result.

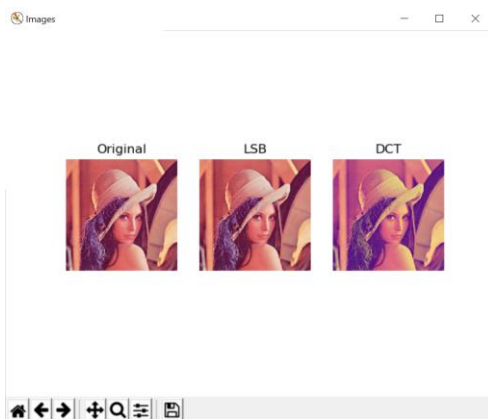
To embedd press '1', to extract press '2', to compare press '3',to decrypt message press'4', press any other button to close: 3
Comparison Results were saved as xls file!
To embedd press '1', to extract press '2', to compare press '3',to decrypt message press'4', press any other button to close: 3

Gambar 14 compare LSB DCT

	A	B	C	D	E	F	G
1	Original vs	MSE	PSNR				
2	LSB	0,476555	51,34967				
3	DCT	1888,319	15,37005				

Gambar 15 Comparison.xlsx

Selain itu, sebelum user menutup aplikasi, akan ditampilkan perbandingan kualitas gambar stego LSB dan DCT dengan gambar asli atau original, sehingga memudahkan user untuk melakukan Analisa.





Gambar 16 perbandingan hasil gambar stego

Terdapat beberapa pengujian untuk menganalisis perbandingan kualitas hasil steganografi antara algoritma LSB dan DCT dengan menerapkan algoritma enkripsi RC4 yaitu pengujian recovery dan fidelity menggunakan MSE, PSNR serta SSIM.

Berikut ini deskripsi file-file yang digunakan untuk penelitian:

TABLE II. DESKRIPSI FILE COVER-IMAGE

	Nama file	Ukuran file	Resolusi
	Lenna.png	462 KB (473.831 bytes)	512 x 512
	Pepper.png	280 KB (287.704 bytes)	512 x 384



Babylon.png 3,48 MB
(3.652.149 bytes) 2012 x 1484



asli5053.png 3,12 MB
(3.279.536 bytes) 3900 x 3900

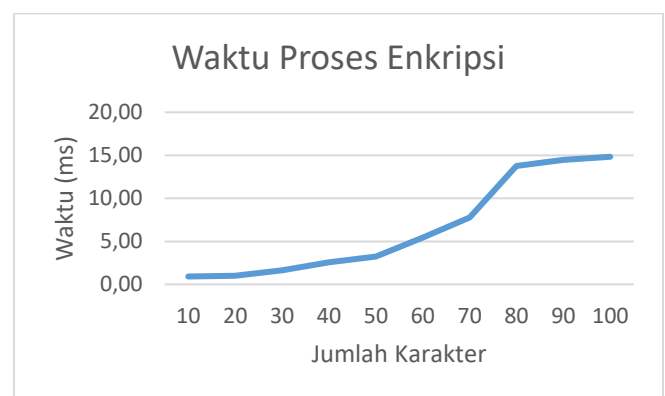
A. Uji Enkripsi dan Dekripsi RC4

Hasil pengujian enkripsi dan dekripsi dari segi waktu proses menggunakan gambar yang sama dengan jumlah karakter yang berbeda ketika disisipkan dalam gambar tersebut. Pengujian dilakukan antara 5 – 10 kali untuk mendapatkan hasil yang stabil, dan juga pengujian dilakukan dengan menggunakan karakter kelipatan 10 (dari 10 karakter hingga 100 karakter).

Berikut adalah tabel dan grafik hasil dari proses waktu enkripsi menggunakan algoritma RC4 yang dikombinasikan pada LSB dan DCT.

TABLE III. UJI ENKRIPSI RC4

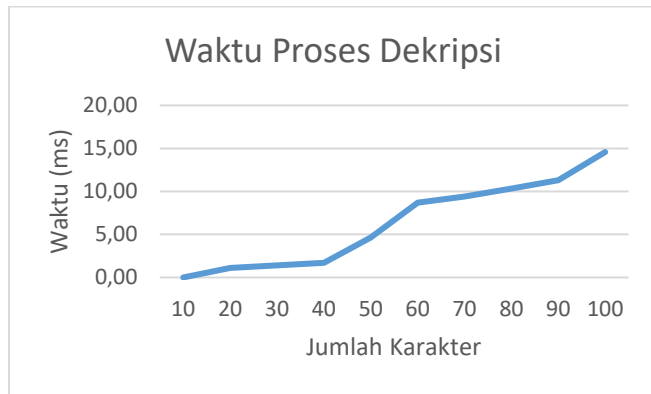
Jumlah Karakter	Waktu Rata-Rata Enkripsi (ms)	Waktu Rata-Rata Dekripsi (ms)
10	0,92	0,00
20	1,02	1,10
30	1,64	1,39
40	2,56	1,70
50	3,22	4,63
60	5,41	8,68
70	7,77	9,39
80	13,75	10,34
90	14,49	11,30
100	14,83	14,58



Gambar 17 Grafik Waktu Proses Enkripsi Pesan

Hasil grafik diatas menggambarkan waktu yang dibutuhkan untuk proses enkripsi menggunakan algoritma RC4 yang dikombinasikan pada metode LSB dan DCT. Sumbu x menyatakan jumlah karakter pesan rahasia yang dienkripsi dan sumbu y menyatakan lama waktu proses enkripsi dalam satuan millisecond. Pengujian dilakukan sebanyak 10 kali untuk tiap jumlah karakter pada tiap gambar

yang digunakan. Adapun nilai yang dimasukkan kedalam grafik merupakan nilai rata-rata supaya mendapatkan waktu yang konsisten, dikarenakan kinerja prosesor yang tidak stabil selama proses pengukuran waktu.



Gambar 18 Grafik Waktu Proses Dekripsi Pesan

Hasil grafik diatas menggambarkan waktu yang dibutuhkan untuk proses dekripsi menggunakan algoritma RC4 yang dikombinasikan pada metode LSB dan DCT. Sumbu x menyatakan jumlah karakter ciphertext hasil extract stego-image yang didekripsi dan sumbu y menyatakan lamanya proses dekripsi dalam satuan millisecond.

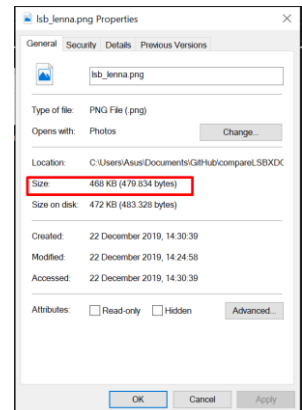
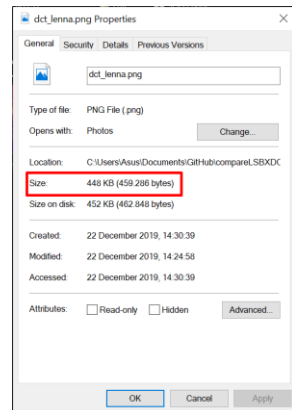
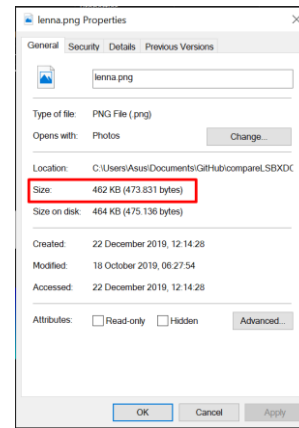
Dari tabel dan grafik hasil pengujian didapatkan kesimpulan bahwa waktu proses enkripsi dan dekripsi bergantung pada banyaknya jumlah karakter. Dengan kata lain, waktu proses enkripsi dan dekripsi berbanding lurus dengan banyaknya jumlah karakter yang digunakan.

B. Uji Recovery

Pengujian Recovery steganografi dikatakan baik dan berhasil jika pesan yang disembunyikan dalam stego image dapat diungkapkan kembali [11]. Tabel IV merupakan tabel perbandingan uji Recovery algoritma LSB dan DCT yang telah dilakukan. Dapat dilihat pada tabel bahwa uji recovery dinilai baik karena kesesuaian plainteks yang berhasil diekstraksi dari stego image.

TABLE IV. UJI RECOVERY Lenna.PNG

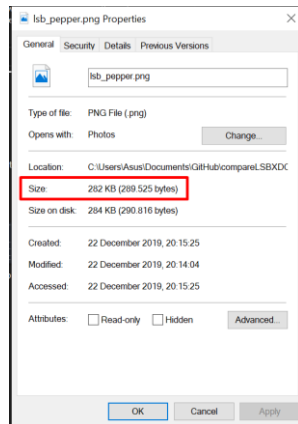
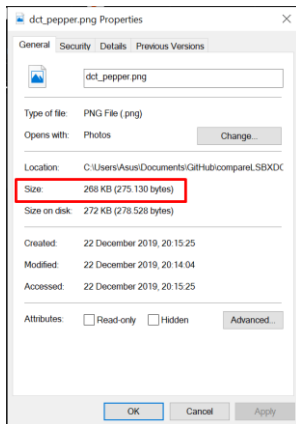
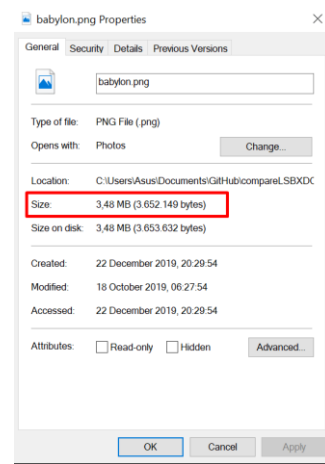
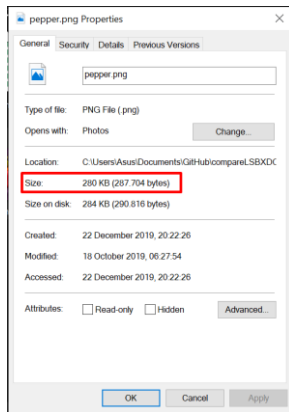
	LSB	DCT
Waktu proses embedding (ms)	687,326	251,537
Waktu proses ekstraksi (ms)	377,140	27,151
Ukuran file stego	468 KB (479.834 bytes)	448 KB (459.286 bytes)
Panjang karakter yang diinput ke cover-image	32	32
Panjang karakter hasil ekstraksi	32	32



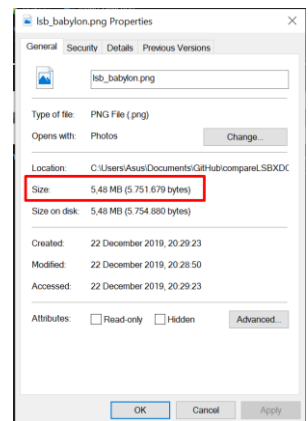
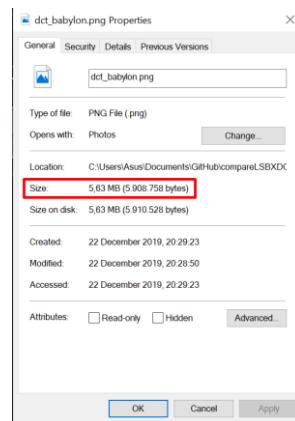
Gambar 19 perbandingan ukuran file asli dan file stego lenna.png

TABLE V. UJI RECOVERY PEPPER.PNG

	LSB	DCT
Waktu proses embedding (ms)	511,188	247,143
Waktu proses ekstraksi (ms)	299,924	21,620
Ukuran file stego	282 KB (289.525 bytes)	268 KB (275.130 bytes)
Panjang karakter yang diinput ke cover-image	68	68
Panjang karakter hasil ekstraksi	68	68



Gambar 20 perbandingan ukuran file asli dan file stego pepper.png



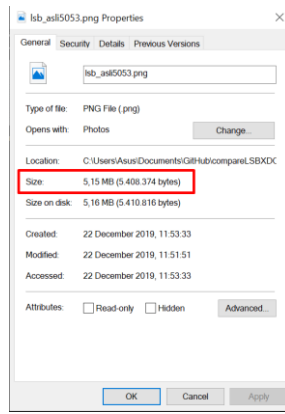
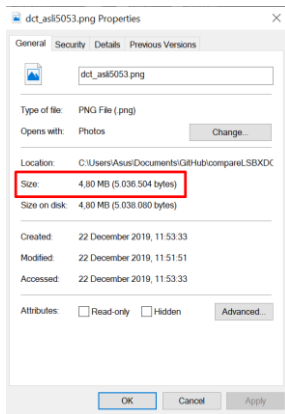
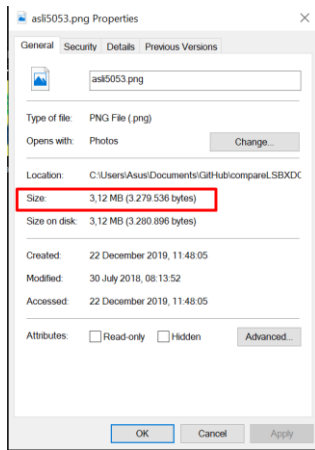
Gambar 21 perbandingan ukuran file asli dan file stego babylon.png

TABLE VI. UJI RECOVERY BABYLON.PNG

	LSB	DCT
Waktu proses embedding (ms)	7176,025	2909,394
Waktu proses ekstraksi (ms)	4135,073	314,449
Ukuran file stego	5,48 MB (5.751.679 bytes)	5,63 MB (5.908.758 bytes)
Panjang karakter yang diinput ke cover-image	130	130
Panjang karakter hasil ekstraksi	130	130

TABLE VII. UJI RECOVERY ASLI5053.PNG

	LSB	DCT
Waktu proses embedding (ms)	38550,809	14839,701
Waktu proses ekstraksi (ms)	22310,167	1872,154
Ukuran file stego	5,15 MB (5.408.374 bytes)	4,80 MB (5.036.504 bytes)
Panjang karakter yang diinput ke cover-image	32	32
Panjang karakter hasil ekstraksi	32	32



Gambar 22 perbandingan ukuran file asli dan file stego asli5053.png

Melalui tabel II serta tabel IV,V,VI,VII dapat diketahui bahwa 2 dari 4 percobaan embedding, ukuran file stego DCT menjadi lebih kecil daripada ukuran file gambar asli. Sedangkan ukuran file stego LSB menjadi lebih besar dari gambar asli. Meskipun pada dua percobaan lainnya, kedua file stego image LSB dan DCT sama-sama lebih besar daripada image asli, akan tetapi 1 dari 2 percobaan lainnya menyatakan file stego image DCT tetap lebih kecil daripada file stego image LSB. Stego image dengan menggunakan algoritma DCT menghasilkan gambar yang terkompresi, sehingga tidak menandakan adanya pesan rahasia didalamnya. Selain itu, waktu proses embedding dan ekstraksi lebih cepat menggunakan algoritma DCT.

C. Uji Fidelity

Tahap uji fidelity ditentukan dengan melakukan perhitungan Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), dan SSIM.

MSE

Mean Squared Error (MSE) adalah pengujian pertama yang dilakukan untuk mengukur kesamaan antara dua buah citra. Persamaan yang digunakan untuk mengukur MSE diperlihatkan pada persamaan 1. I adalah pixel gambar asli berukuran $m \times n$ sedangkan K adalah gambar yang telah disisipkan steganografi.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

PSNR

Pick Signal to Noise (PSNR) adalah parameter pengukuran distorsi sebuah file citra. Persamaan PSNR diperlihatkan pada persamaan 2. MAX adalah nilai sebuah pixel yang mungkin untuk sebuah pixel gambar.

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

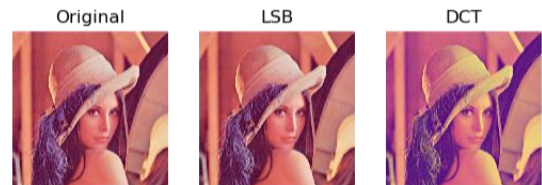
Berdasarkan pengujian MSE dan PSNR didapatkan bahwa nilai MSE yang dihasilkan kurang dari 1 dB dan PSNR di atas 50, berarti perubahan kualitas warna antara citra asli dengan stego image tidak mengalami perubahan yang signifikan, sehingga keberadaan dari file yang tersembunyi tidak mudah di deteksi oleh indra penglihatan manusia.

SSIM

Persamaan 3 memperlihatkan perhitungan *Structural Similarity* (SSIM), dimana μ , σ , dan σ_{xy} masing-masing adalah rata-rata, varians, dan kovarians dari suatu gambar, dan c_1 , c_2 adalah konstanta penyeimbang. SSIM memiliki nilai dengan rentang 0 – 1. SSIM dengan nilai mendekati 1 berarti citra yang diuji memiliki kedekatan terhadap citra aslinya.

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

Berikut ini perbandingan MSE, PSNR, dan SSIM dari beberapa file gambar yang diuji.



Gambar 23 perbandingan gambar original, LSB, dan DCT lenna.png

TABLE VIII. UJI FIDELITY LENNA.PNG

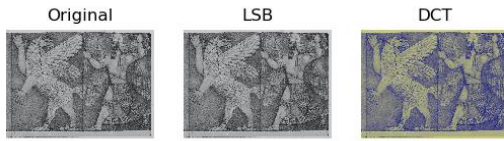
	LSB	DCT
MSE	0,466	1888,363
PSNR	51,445	15,369
SSIM	1,00	0,76



Gambar 24 perbandingan gambar original, LSB, dan DCT pepper.png

TABLE IX. UJI FIDELITY PEPPER.PNG

	LSB	DCT
MSE	0,0309	6860,618
PSNR	63,225	9,767
SSIM	1,00	0,75



Gambar 25 perbandingan gambar original, LSB, dan DCT babylon.png

TABLE X. UJI FIDELITY BABYLON.PNG

	LSB	DCT
MSE	0,86	6866,539
PSNR	54,656	9,7634
SSIM	1,00	0,75



Gambar 26 perbandingan gambar original, LSB, dan DCT asli5053.png

TABLE XI. UJI FIDELITY ASLI5053.PNG

	LSB	DCT
MSE	0,09	6864,421
PSNR	55,265	9,7647
SSIM	1,00	0,60

Berdasarkan pengujian aspek fidelity dari 4 sampel percobaan, stego image algoritma LSB menghasilkan MSE kurang dari 1 dB, serta nilai PSNR lebih dari 50 sehingga perubahan kualitas warna antara citra asli dengan stego image tidak mengalami perubahan yang signifikan. Akibatnya keberadaan file yang tersembunyi tidak mudah di deteksi oleh indra penglihatan manusia. Selain itu, nilai SSIM gambar stego LSB semuanya bernilai 1,00 yang artinya citra yang diuji memiliki kedekatan terhadap citra aslinya. Berbeda dengan stego image LSB, stego image DCT menghasilkan MSE lebih dari 1 dB bahkan angkanya mencapai ribuan, artinya kualitas warna citra asli mengalami perubahan yang sangat signifikan dan terlihat perbedaannya oleh indra penglihatan manusia. Nilai PSNR-nya pun kurang dari 50, sehingga mengakibatkan file tersembunyi lebih mudah untuk dideteksi. Selain itu, nilai SSIM stego image SSIM kurang dari 1 yang berarti citra yang diuji kurang memiliki kedekatan terhadap citra aslinya.

V. CONCLUSIONS

Dalam penelitian ini diterapkan keamanan berlapis pada steganografi dengan menerapkan kriptografi RC4. Pesan rahasia dienkripsi terlebih dahulu kemudian dimasukkan ke dalam cover citra dengan metode Least Significant Bit (LSB) yaitu setiap bit pesan rahasia yang dimasukkan ke dalam bit terakhir dari gambar digital serta metode Discrete Cosine Transform (DCT) yang mengubah data digital ke dalam bentuk domain frekuensi. Dari hasil pengujian enkripsi dan dekripsi RC4 didapatkan kesimpulan bahwa waktu proses enkripsi dan dekripsi bergantung pada banyaknya jumlah karakter. Dengan kata lain, waktu proses enkripsi dan dekripsi berbanding lurus dengan banyaknya jumlah karakter yang digunakan. Hasil pengujian recovery menunjukkan algoritma DCT memiliki kecepatan waktu proses dan kompresi file yang lebih baik. Stego image dengan menggunakan algoritma DCT menghasilkan gambar yang terkompresi, sehingga tidak menandakan adanya pesan rahasia didalamnya. Selain itu, waktu proses embedding dan ekstraksi lebih cepat menggunakan algoritma DCT dibandingkan dengan metode LSB. Adapun hasil pengujian fidelity memperlihatkan metode LSB lebih baik dibandingkan dengan DCT. Pengujian aspek fidelity dari 4 sampel percobaan, stego image algoritma LSB menghasilkan MSE kurang dari 1 dB, serta nilai PSNR lebih dari 50 sehingga perubahan kualitas warna antara citra asli dengan stego image tidak mengalami perubahan yang signifikan. Akibatnya keberadaan file yang tersembunyi tidak mudah di deteksi oleh indra penglihatan manusia. Selain itu, nilai SSIM gambar stego LSB semuanya bernilai 1,00 yang artinya citra yang diuji memiliki kedekatan terhadap citra aslinya.

Perlu diskusi lebih mendalam karena pada penelitian ini belum dilakukan percobaan robustness (tahan terhadap berbagai operasi manipulasi yang dilakukan pada citra penampung) serta diharapkan agar media yang disisipkan pesan rahasia bisa berupa lebih banyak tipe image serta bisa berupa file audio atau video. Selain itu, diharapkan pula untuk dilakukan pengembangan aplikasi berbasis GUI (*Graphical User Interface*)

VI. REFERENCES

- [1] A. Rahmatulloh, H. Sulastri, and R. Nugroho, "Keamanan RESTful Web Service Menggunakan JSON Web Token (JWT) HMAC SHA-512," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 2, 2018.
- [2] A. M. Elhanafi, "Implementasi Pengamanan Informasi Pesan Rahasia Menggunakan Color Ordering and Mapping Pada Media Citra Digital," *J. Ilm. Teknol. Harapan*, vol. 6, no. 02, pp. 87–91, 2017.
- [3] M. A. A. Pujari and M. S. S. Shinde, "Data Security using Cryptography and Steganography," *IOSR J. Comput. Eng.*, vol. 18, no. 04, pp. 130–139, 2016.
- [4] R. Sulaiman and B. Isnanto, "Peningkatan Keamanan Pesan Dengan Kriptografi RC4 dan Steganografi LSB Pada File JPEG," *Konf. Nas. Sist. Inf. 2018*, pp. 8–9, 2018.
- [5] B. W. Santoso and F. R. Alhadi, "PERBANDINGAN HASIL IMPELEMENTASI STEGANOGRAFI

DAN KRIPTOGRAFI MENGGUNAKAN LSB (LEAST SIGNIFICANT BIT) DENGAN EOF (END OF FILE),” *J. ICT Learn.*, vol. Vol. 3, no. 01, pp. 81–97, 2017.

- [6] J. Zainal, A. Pagar, and N. K. Bandarlampung, “Implementasi Teknik Steganografi Least Significant Bit (Lsb) Dan Kompresi Untuk Pengamanan Data Pengiriman Surat Elektronik,” vol. 10, no. 2, pp. 1–7, 2016.
- [7] N. Anwar, “Perancangan Steganografi Hidden Message Dengan Metode Least Significant Bit Insertion (Lsb) Berbasis Matlab,” *J. Algoritm. Log. dan Komputasi*, vol. 1, no. 1, pp. 25–30, 2018.
- [8] R. Kasmala, A. Budimansyah, and U. T. Lenggana, “Kompresi Citra Dengan Menggabungkan Metode Discrete Cosine Transform (DCT) dan Algoritma Huffman,” *J. Online Inform.*, vol. 2, no. 1, p. 1, 2017.
- [9] A. R. Saputra, “Color Image Compression Using Discrete Cosinus Transform (DCT),” *J. Ilm. Inform. Komput. Univ. Gunadarma*, no. 100, 2017.
- [10] J. I. Polinema *et al.*, “IMPLEMENTASI STEGANOGRAPHY MENGGUNAKAN ALGORITMA DISCRETE COSINE TRANSFORM,” *J. Inform. Polinema*, vol. 2, no. 1, pp. 35–39, 2015.
- [11] H. A. Damanik and M. Anggraeni, “Teknik Pengujian Keamanan Data Text Bertingkat Dengan Metode Steganography Lsb Dan Teknik Enkripsi,” *J. Penelit. Pos dan Inform.*, vol. 8, no. 2, p. 109, 2018.