Jurnal EurekaMatika

Journal homepage: https://ejournal.upi.edu/index.php/JEM

Pengamanan Citra *Grayscale* Menggunakan Penggabungan Kriptografi Visual *Secret Sharing* dan Steganografi *Enhanced Least Significant Bit*

Salman Al Ghifary Sujono, Rini Marwati, Cece Kustiawan*

Program Studi Matematika, Fakultas Pendidikan dan Ilmu Pengetahuan Alam,
Universitas Pendidikan Indonesia, Indonesia

*Correspondence: E-mail: rinimarwati@upi.edu

ABSTRAK

Penelitian ini mengkaji tentang penggabungan kriptografi visual *Secret Sharing* dan *steganografi Enhanced Least Significant Bit* dalam mengamankan citra *grayscale*. Citra pesan disamarkan menjadi beberapa bagian dengan kriptografi visual *Secret Sharing* kemudian disembunyikan ke citra lain dengan steganografi *Enhanced Least Significant Bit*. *Enhanced Least Significant Bit* merupakan metode *Least Significant Bit* yang ditingkatkan di mana bit yang digunakan sebagai tempat penyembunyian pesan bukanlah bit LSB terakhir melainkan dua atau tiga bit LSB terakhir. Hasil penelitian menunjukkan bahwa penggabungan ini memiliki tingkat keamanan yang tinggi karena memerlukan dua teknik keamanan data dan banyak citra yang sesuai untuk mendapatkan pesan yang sudah disamarkan.

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima 19 Juli 2024
Direvisi, 17 Oktober 2024
Disetujui 26 Oktober 2024
Tersedia online 9 November 2024
Dipublikasikan 9 November 2024

Kata Kunci:
Citra Grayscale,
Enhanced LSB,
Kriptografi Visual,
Skema Secret Sharing,
Steganografi.

ABSTRACT

This study examines the combination of Visual Secret Sharing cryptography and Enhanced Least Significant Bit steganography in securing grayscale images. The image message is disguised into several parts using Visual Secret Sharing cryptography and then hidden in another image using Enhanced Least Significant Bit steganography. Enhanced Least Significant Bit is an improved method of the Least Significant Bit where the bits used for hiding the message are not the last LSB bits but last two or last three LSB bits. The study results show that this combination has a high level of security as it requires two data security techniques and the appropriate number of images to retrieve the disguised message.

Keywords:
Enhanced LSB,
Grayscale Image,
Secret Sharing Scheme,
Steganography,
Visual Cryptography.

1. PENDAHULUAN

Seiring perkembangan zaman, komunikasi digital menjadi suatu hal yang umum dan sering dilakukan oleh semua orang untuk saling mengirim informasi. Salah satu bentuk informasi yang sering dikirimkan adalah citra, termasuk citra *grayscale* (keabuan) yang sering digunakan dalam berbagai aplikasi seperti medis dan pengolahan gambar. Citra yang dikirim secara digital terutama yang bersifat rahasia rawan terhadap penyerangan dan penyadapan oleh pihak-pihak yang tidak bertanggung jawab (Azanuddin *et al*., 2022). Ancaman terhadap citra tersebut dapat dicegah dengan menggunakan teknik kriptografi visual.

Kriptografi visual merupakan sebuah teknik kriptografi yang memungkinkan informasi citra dienkripsi menjadi beberapa bagian dan hanya dapat didekripsi jika memiliki semua bagian dari citra tersebut (Zahra *et al.*, 2021). Salah satu teknik pada kriptografi visual adalah *Secret Sharing*. Proses enkripsi pada teknik ini dilakukan dengan membagi citra menjadi *n share* yang di mana setiap *share* tersebut adalah subset dari citra awal dan proses dekripsi dilakukan dengan menggabungkan seluruh *n share* sehingga diperoleh kembali citra awal (Naor & Shamir, 1995).

Beberapa penelitian sebelumnya mengenai kriptografi visual telah dilakukan, di antaranya oleh Pella *et al* (2021) dan Bunga *et al.* (2023) yang menyimpulkan bahwa citra yang dienkripsi menggunakan skema kriptografi visual *Secret Sharing* dapat tersamarkan dengan baik dalam bentuk citra abstrak dan dapat diperoleh kembali citra awalnya apabila dilakukan proses dekripsi. Meskipun *share* yang dihasilkan sudah berupa citra abstrak, tetapi hal tersebut menimbulkan kecurigaan bagi pihak lain yang melihat selain pihak penerima. Oleh karena itu, diperlukan adanya keamanan tambahan dari hasil kriptografi visual *Secret Sharing*.

Steganografi adalah teknik penyembunyian informasi ke dalam informasi lain (*cover object*) sehingga keberadaannya tidak diketahui. Salah satu metode steganografi adalah LSB (*Least Significant Bit*). Metode LSB merupakan salah satu metode yang paling umum digunakan dalam steganografi karena sederhana dan efektif (Lutfi *et al*., 2018). Metode LSB memanfaatkan nilai bit yang tidak berarti sebagai tempat penyembunyian informasi sehingga tidak mengubah *cover object*. Metode LSB dapat digunakan untuk semua jenis citra digital.

Enhanced LSB merupakan metode LSB yang ditingkatkan atau varian dari metode LSB di mana banyak bit yang digunakan sebagai tempat penyembunyian informasi divariasikan untuk meningkatkan kapasitas penyembunyian dan kekuatan keamanan. Beberapa penelitian sebelumnya mengenai *Enhanced* LSB telah dilakukan, di antaranya oleh Kurniasih *et al.* (2023) dan Alqadi *et al.* (2019) yang menyembunyikan informasi berupa teks ke dalam citra. Bit yang digunakan sebagai penyembunyian pesan berturut-turut adalah bit ke-6 dan dua bit terakhir (bit ke-7 dan bit ke-8). Hasil kedua penelitian menyimpulkan bahwa hasil citra yang digunakan sebagai tempat penyembunyian (*stego image*) memiliki nilai PSNR tinggi yang berarti pesan tersembunyi dengan baik dan tidak terjadi perubahan yang signifikan pada *stego image*.

Untuk penyembunyian informasi berupa citra ke dalam citra sudah dilakukan sebelumnya oleh Rawat *et al.* (2023) yang menggunakan dua skema metode *Enhanced* LSB. Skema pertama mengganti dua bit LSB *cover image* dengan dua bit MSB (*Most Significant Bit*) citra rahasia. Skema kedua mengganti satu bit LSB dari setiap piksel merah, dua bit LSB dari setiap piksel hijau dan tiga bit LSB dari setiap piksel biru pada *cover image* dengan satu bit MSB, dua bit MSB dan tiga bit MSB pada citra rahasia. Hasil penelitian menyimpulkan bahwa *stego image* yang dihasilkan memiliki nilai PSNR lebih dari 30 *dB* yang berarti citra rahasia tersembunyi dengan baik tanpa merubah *stego image*.

Berdasarkan uraian di atas, penulis menggabungkan kriptografi visual *Secret Sharing* dan steganografi *Enhanced Least Significant Bit* untuk mengamankan citra digital berupa citra

grayscale. Seluruh proses tersebut akan dikonstruksi menjadi sebuah program aplikasi komputer menggunakan bahasa pemrograman Python versi 3.12 64-bit.

2. METODE

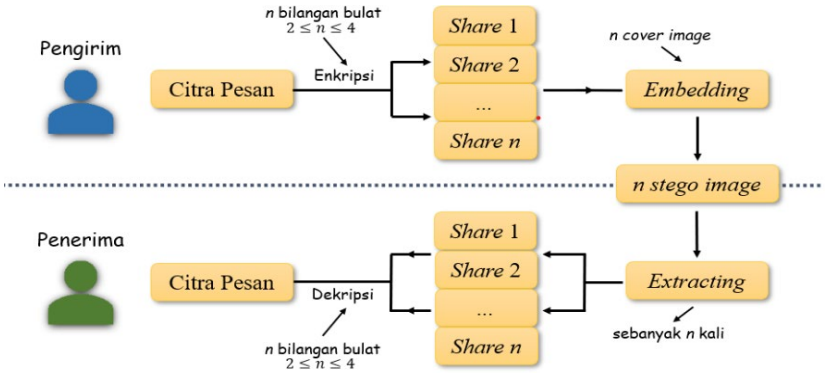
Informasi rahasia yang digunakan dalam penelitian ini adalah citra *grayscale*. Teknik kriptografi yang digunakan adalah kriptografi visual *Secret Sharing*, di mana citra rahasia akan dienkripsi menggunakan aritmatika modulo untuk menghasilkan *n* citra abstrak (*share)*. Begitu juga dengan proses dekripsi yang membutuhkan *n share* untuk mendapatkan citra rahasia kembali.

Teknik steganografi yang digunakan adalah *Enhanced Least Significant Bit*, yang merupakan metode LSB yang ditingkatkan atau varian dari metode LSB. *Enhanced* LSB yang digunakan pada penelitian ini menggunakan dua bit terakhir LSB dan tiga bit terakhir LSB dari *cover object* sebagai tempat penyembunyian citra sehingga dapat menyimpan lebih banyak informasi. *Cover object* yang digunakan adalah citra berwarna (RGB).

2.1 Model Dasar

Proses enkripsi diawali dengan pengirim memilih citra pesan yang akan digunakan dalam bentuk *.png dan *n* bilangan bulat untuk menghasilkan berapa banyak *share*. Setiap piksel pada *share n-1* dibuat secara acak kemudian ditambahkan masing-masing ke setiap piksel pada *share n* yang dibatasi dengan modulo 256. Setiap *n share* yang dihasilkan akan disembunyikan ke dalam citra berwarna (*cover image*). *Cover image* yang digunakan harus dalam bentuk file dan ukuran yang sama dengan citra rahasia dan *share*. Proses penyembunyian (*embedding)* dilakukan menggunakan dua skema. Skema pertama mengganti setiap dua bit LSB terakhir pada *cover image* dengan dua bit MSB pada *share*. Skema kedua mengganti setiap setiap tiga bit LSB terakhir pada *cover image* dengan tiga bit MSB pada *share*. Proses *embedding* dilakukan sebanyak *n* kali. Hasil dari proses *embedding* yaitu *n stego image* dikirim ke penerima.

Proses *extracting* dan dekripsi dimulai dari penerima menerima *n stego image* kemudian mengekstraksi *share* yang ada sesuai dengan skema yang dilakukan pada proses *embedding*. Proses *extracting* dilakukan sebanyak *n* kali. Seluruh *share* yang diperoleh disatukan melalui proses dekripsi dengan cara untuk setiap piksel pada *share n* dikurangi dengan setiap piksel pada *share n-1* yang dibatasi dengan modulo 256. Proses dekripsi menghasilkan citra pesan awal. Skema penggabungan dapat dilihat pada Gambar 1.



Gambar 1. Skema Penggabungan Kriptografi Visual *Secret Sharing* dan Steganografi *Enhanced Least Significant Bit*

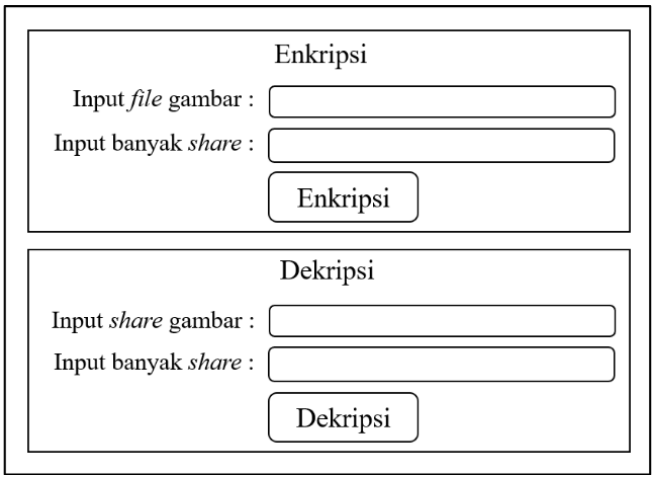
2.2 Konstruksi Program Aplikasi

Program aplikasi terdiri dari tiga buah halaman, yang terdiri dari kriptografi, steganografi dan uji citra. Halaman kriptografi digunakan untuk melakukan proses enkripsi citra pesan menjadi *n share* dan proses dekripsi *n share* menjadi citra pesan kembali. Halaman steganografi berfungsi untuk melakukan proses *embedding share* ke dalam *cover image* sehingga menghasilkan *stego image* dan proses *extracting stego image* sehingga mendapatkan *share*. Halaman uji citra berfungsi untuk melakukan pengujian terhadap *stego image* dan citra pesan hasil proses dekripsi. Rancangan masukan (*input*) serta luaran (*output*) program dapat dilihat pada Tabel 1.

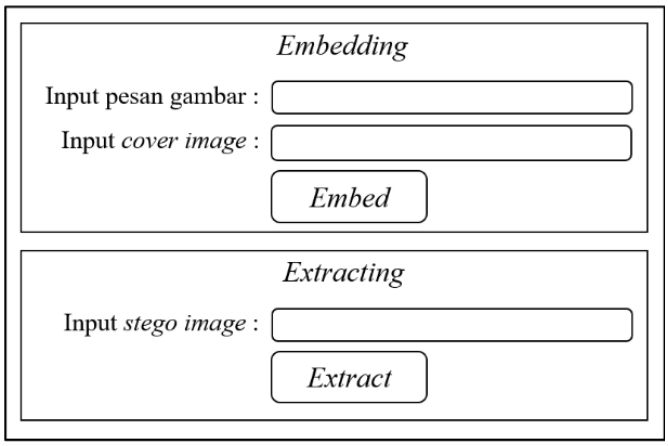
Tabel 1. Rancangan Masukan dan Luaran Program

Keterangan	Kriptografi	Steganografi	Uji Citra
Masukan (*input*)	Enkripsi • Citra pesan • *n* bilangan bulat ($2 \leq n \leq 4$) Dekripsi • *Folder share* • *n* bilangan bulat ($2 \leq n \leq 4$)	*Embedding* • *n share* • *Cover Image* *Extracting* • *n stego Image*	• *Stego Image* • Citra pesan hasil proses dekripsi
Luaran (output)	Enkripsi • *n share* Dekripsi • Citra pesan	*Embedding* • *n stego Image* *Extracting* • *n share*	• Nilai PSNR • Nilai NCC

Rancangan tampilan program aplikasi dapat dilihat pada Gambar 2 dan 3.



Gambar 2. Rancangan Tampilan Kriptografi



Gambar 3. Rancangan Tampilan Steganografi

2.3 Proses Validasi

Proses validasi dilakukan untuk memvalidasi program aplikasi yang telah dibuat. Program aplikasi tervalidasi jika citra pesan dapat diperoleh kembali pada proses *extracting* dan dekripsi dan melakukan pengujian citra pesan tersebut menggunakan uji *Normalized Cross-Correlation* (NCC) untuk mengetahui keidentikan dengan citra pesan awal citra serta melakukan pengujian *stego image* yang diperoleh menggunakan uji *Peak Signal-to-Noise Ratio* (PSNR) untuk mengetahui kualitas *stego image*.

2.3.1 *Normalized Cross-Correlation* (NCC)

Normalized Cross-Correlation (NCC) adalah metode untuk mengukur kemiripan suatu citra berdasarkan fungsi korelasi. Berikut persamaan untuk menghitung NCC (Kaso, 2018):

$$NCC = \frac{\sum_{k=1}^{n}(x_{ik} - \bar{x}_i) \cdot (x_{jk} - \bar{x}_j)}{\sqrt{\left[\sum_{k=1}^{n}(x_{ik} - \bar{x}_i)^2 \sum_{k=1}^{n}(x_{jk} - \bar{x}_j)^2\right]}}$$

$$\text{di mana } \bar{x}_i = \frac{1}{n}\sum_{k=1}^{n} X_{ik} \text{ dan } \bar{x}_j = \frac{1}{n}\sum_{k=1}^{n} X_{jk}$$

dengan:
NCC = nilai korelasi antara dua buah citra
X_{ik} = nilai piksel ke-k pada citra i
X_{jk} = nilai piksel ke-k pada citra j
$\bar{x}_i$ = rata-rata nilai piksel citra i
$\bar{x}_j$ = rata-rata nilai piksel citra j
n = jumlah piksel pada suatu citra

Suatu citra dikatakan memiliki tingkat keidentikan yang baik dengan citra lain apabila nilai NCC yang diperoleh berada di antara rentang 0 sampai 1, di mana jika semakin mendekati nol maka kedua citra semakin tidak identik (Kliwati, 2018).

2.3.2 *Peak Signal-to-Noise Ratio* (PSNR)

Peak Signal-to-Noise Ratio (PSNR) adalah parameter yang menunjukkan rasio tingkat toleransi *noise* tertentu terhadap banyaknya *noise* pada suatu piksel citra. Noise adalah kerusakan piksel pada bagian tertentu dalam sebuah citra sehingga memengaruhi kualitas dari piksel tersebut, sehingga PSNR digunakan untuk mengetahui kualitas suatu citra. Perhitungan PSNR memerlukan MSE (Mean Squared Error) yaitu parameter yang

menunjukkan tingkat kesalahan piksel-piksel citra hasil pemrosesan terhadap citra asli. Berikut persamaan untuk menghitung MSE dan PSNR:

$$MSE = \frac{1}{mn}\sum_{x=1}^{m}\sum_{y=1}^{n}[I(x,y) - I'(x,y)]^2$$

dengan:
MSE = nilai *Mean Squared Error* antara *cover image* dan *stego image*
m = panjang citra (dalam piksel)
n = lebar citra (dalam piksel)
$I(x,y)$ = nilai piksel dari *cover image*
$I'(x,y)$ = nilai piksel dari *stego image*

$$PSNR = 10 \cdot \log\left(\frac{MAX_I^2}{MSE}\right)$$

dengan:
PSNR = nilai PSNR *stego image* (dalam *dB*)
MAX_1^2 = maksimum nilai piksel
MSE = nilai *Mean Squared Error*

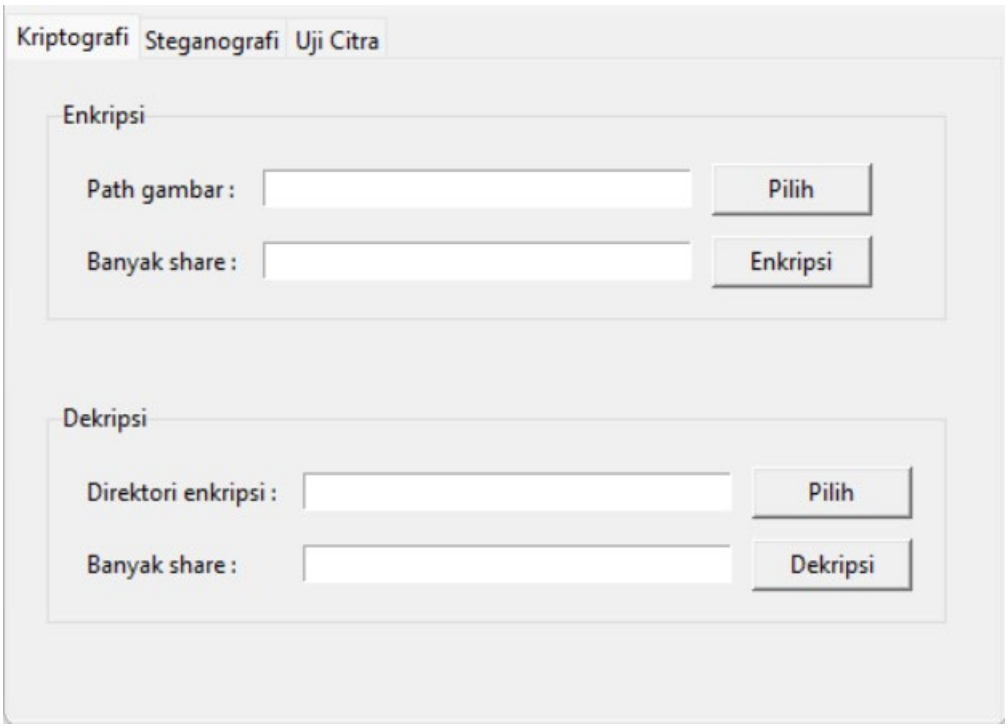
Suatu *stego image* atau citra dapat dikatakan memiliki kualitas yang baik dan dapat diterima apabila nilai PSNR yang diperoleh adalah lebih besar dari 30 dB. Semakin tinggi nilai PSNR yang diperoleh maka semakin tinggi kualitas dari *stego image* (Hore & Ziou, 2010).

3. HASIL DAN PEMBAHASAN

Program aplikasi dibuat untuk mempermudah user dalam melakukan proses enkripsi dan *embedding* untuk menghasilkan *stego image*, melakukan proses *extracting* dan dekripsi untuk mendapatkan citra pesan kembali serta menguji keidentikan citra hasil proses dekripsi dan menguji kualitas *stego image*. Program dibuat menggunakan bahasa pemrograman Python versi 3.12 64-bit.

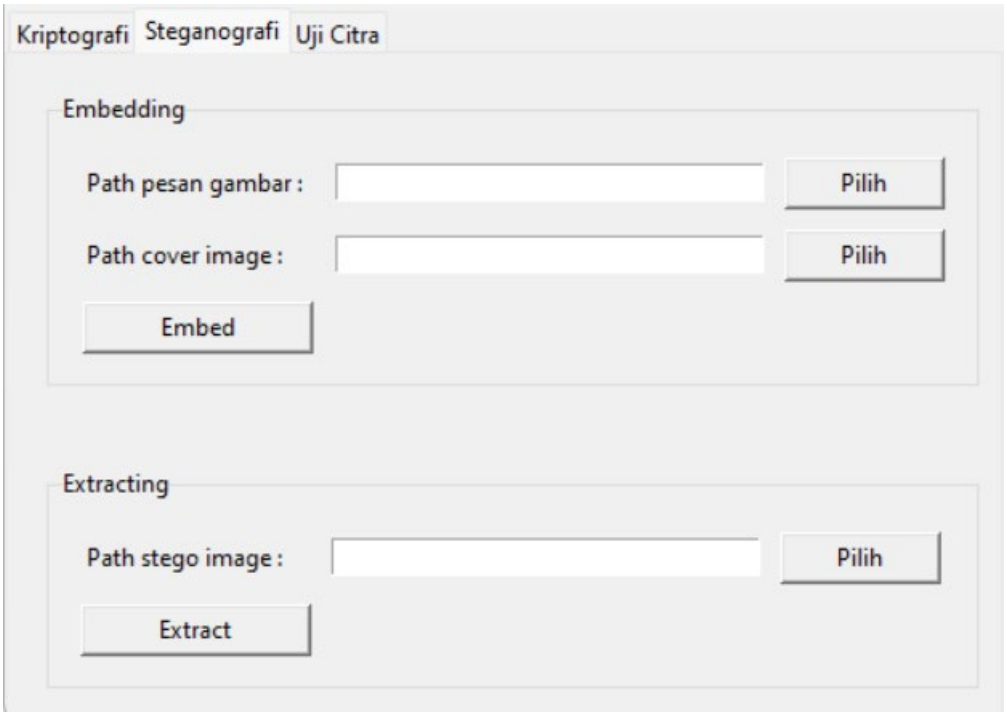
3.1 Tampilan Program Aplikasi

Pada tampilan kriptografi yang dapa dilihat pada Gambar 4, terdapat dua buah *frame* yaitu *frame* enkripsi dan *frame* dekripsi. Pada *frame* enkripsi, *button* Pilih berfungsi untuk memilih citra pesan yang akan dienkripsi dan *button* Enkripsi berfungsi untuk memulai proses enkripsi. Sedangkan pada *frame* dekripsi, *button* Pilih berfungsi untuk memilih *folder* tempat penyimpanan *share* dan *button* Dekripsi berfungsi untuk memulai proses dekripsi.

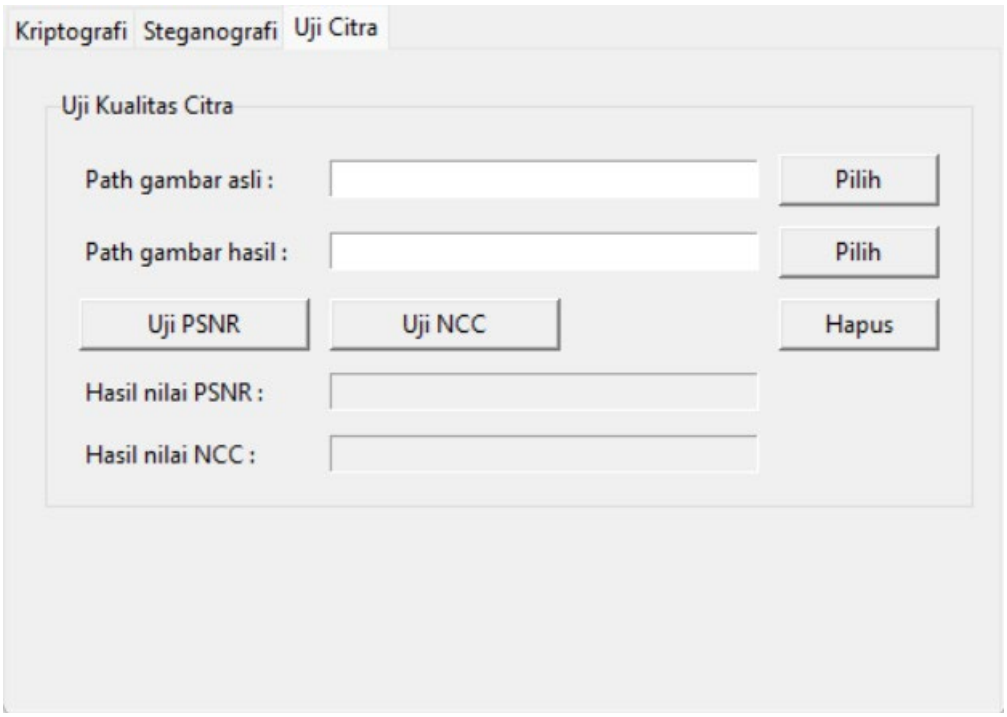


Gambar 4. Tampilan Kriptografi

Pada tampilan steganografi yang dapat dilihat pada Gambar 5, terdapat dua buah *frame* yaitu *frame embedding* dan *frame extracting*. Pada *frame embedding*, *button* Pilih berfungsi untuk memilih *share* dan *cover image* yang akan digunakan dalam proses *embedding* dan *button Embed* berfungsi untuk memulai proses *embedding*. Sedangkan pada *frame extracting*, *button* Pilih berfungsi untuk memilih *stego image* dan *button Extract* berfungsi untuk memulai proses *extracting*.



Gambar 5. Tampilan Steganografi



Gambar 6. Tampilan Uji Citra

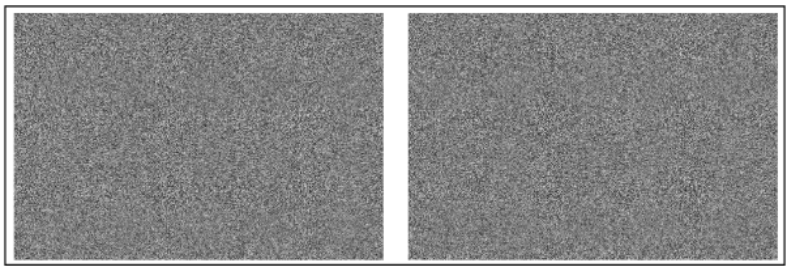
Pada tampilan uji citra yang dapat dilihat pada Gambar 6, terdapat *button* Pilih yang berfungsi untuk memilih *cover image* dan *stego image*, serta citra pesan awal dan citra pesan hasil proses dekripsi. *Button* Uji PSNR berfungsi untuk menghitung nilai PSNR dan *button* Uji NCC berfungsi untuk menghitung nilai NCC.

3.2 Validasi Program Aplikasi

Validasi program aplikasi dilakukan untuk mengetahui apakah *stego image* yang dihasilkan dari proses enkripsi dan *embedding* dapat dikembalikan menjadi citra pesan awal melalui proses *extracting* dan dekripsi atau tidak. Validasi dilakukan dengan meng-*input*-kan citra pesan yang dapat dilihat pada Gambar 7 dan nilai *n* bilangan bulatnya adalah 2. Hasil enkripsi dari program berupa *share* dapat dilihat pada Gambar 8. *Cover image* yang digunakan dapat dilihat pada Gambar 9.

Dalam teknik steganografinya, validasi dilakukan dengan dua skema. Skema pertama menggunakan dua bit LSB terakhir dan skema kedua menggunakan tiga bit LSB terakhir. Hasil proses dengan skema pertama menghasilkan *stego image* dan citra pesan hasil proses dekripsinya dapat dilihat pada Gambar 10. Sedangkan *stego image* hasil proses dengan skema kedua dan citra pesan hasil proses dekripsinya dapat dilihat pada Gambar 11. Hasil yang didapatkan dari kedua skema tersebut adalah citra pesan yang serupa dengan citra pesan yang di-*input*-kan sebelumnya.

Gambar 7. Citra Pesan



Gambar 8. Hasil Enkripsi

Gambar 9. *Cover Image*

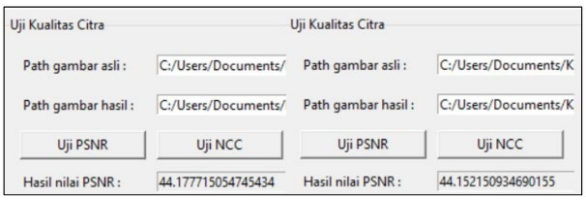
Gambar 10. *Stego Image* dan Citra Hasil Dekripsi Skema Pertama

Gambar 11. *Stego Image* dan Citra Hasil Dekripsi Skema Kedua

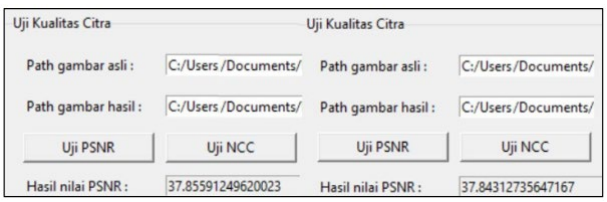
3.3 Validasi Pengujian Citra Hasil Pemrosesan

Setelah validasi program aplikasi dilakukan, validasi citra akan dilakukan untuk mengetahui kualitas dari *stego image* dan keidentikan dari citra pesan hasil proses dekripsi dengan citra pesan awal. Validasi pertama dilakukan dengan meng-*input*-kan *cover image* dan *stego image* yang diperoleh pada Gambar 10 dan Gambar 11 ke dalam tampilan uji citra seperti pada Gambar 6, kemudian menekan *button* Uji PSNR.

Gambar 12 menunjukkan bahwa nilai PSNR dari *stego image* pada Gambar 10 yang menggunakan skema pertama adalah 44,1777 *dB* dan 44,1521 *dB*. Sedangkan Gambar 13 menunjukkan bahwa nilai PSNR dari *stego image* pada Gambar 11 yang menggunakan skema kedua adalah 37,8559 *dB* dan 37,8431. Lebih lengkapnya, nilai PSNR yang dihasilkan dengan menggunakan dua skema berbeda dan nilai *n* bilangan bulat (banyak *share*) yang berbeda dapat dilihat pada Tabel 2.



Gambar 12. Nilai PSNR *Stego Image*



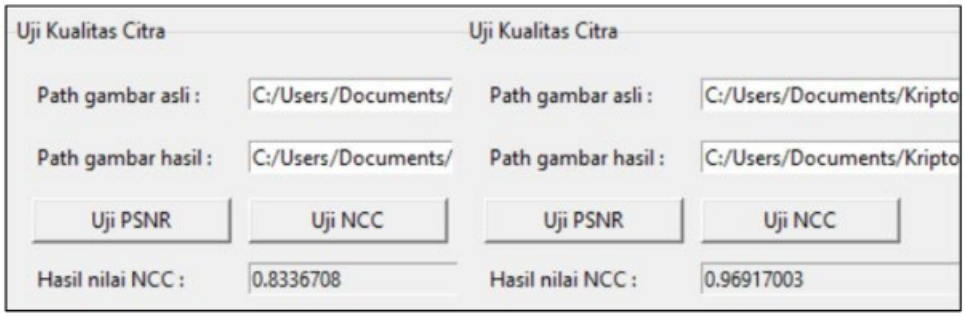
Gambar 13. Nilai PSNR *Stego Image*

Tabel 2. Nilai PSNR *Stego Image*

Skema Pertama			Skema Kedua		
Banyak *Share* (*n*)	***Stego Image***	**Nilai PSNR**	**Banyak *Share* (*n*)**	***Stego Image***	**Nilai PSNR**
2	*Stego Image 1*	44,1777	2	*Stego Image 1*	37,8559
	Stego Image 2	44,1521		*Stego Image 2*	37,8431
3	*Stego Image 1*	44,1753	3	*Stego Image 1*	37,8577
	Stego Image 2	44,1516		*Stego Image 2*	37,8445
	Stego Image 3	44,1583		*Stego Image 3*	37,9141
4	*Stego Image 1*	44,1733	4	*Stego Image 1*	37,8597
	Stego Image 2	44,1533		*Stego Image 2*	37,8449
	Stego Image 3	44,1506		*Stego Image 3*	37,9183
	Stego Image 4	44,1456		*Stego Image 4*	37,9101

Berdasarkan Tabel 2, hasil nilai PSNR yang didapatkan oleh *stego image* dengan menggunakan masukan (*input*) banyak *share* yang berbeda dan skema yang berbeda adalah lebih dari 30 *dB*. Hal ini berarti *stego image* memiliki kualitas yang baik dan dapat diterima.

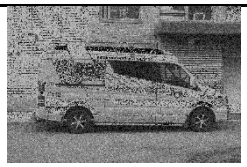
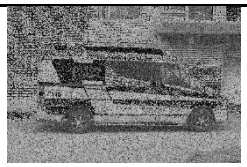
Validasi kedua dilakukan dengan meng-*input*-kan citra pesan awal dan citra pesan hasil proses dekripsi yang diperoleh pada Gambar 10 dan Gambar 11 ke dalam tampilan uji citra seperti pada Gambar 6, kemudian menekan *button* Uji NCC. Hasil uji NCC dapat dilihat pada Gambar 14.



Gambar 14. Hasil Nilai NCC Skema Pertama (kiri) dan Skema Kedua (kanan)

Pada Gambar 14, nilai NCC yang diperoleh dengan skema pertama adalah 0,8336 dan nilai NCC yang diperoleh dengan skema kedua adalah 0,9691. Lebih lengkapnya, nilai NCC yang dihasilkan dengan menggunakan dua skema berbeda dan nilai *n* bilangan bulat (banyak *share*) yang berbeda dapat dilihat pada Tabel 3.

Tabel 3. Nilai NCC Citra Hasil Dekripsi

Skema Pertama			Skema Kedua		
Banyak *Share* (*n*)	**Citra Hasil**	**Nilai NCC**	**Banyak *Share* (*n*)**	**Citra Hasil**	**Nilai NCC**
2		0,8336	2		0,9691
3		0,4182	3		0,9111
4		-0,0490	4		0,7457

Berdasarkan Tabel 3, hasil nilai NCC yang didapatkan oleh citra hasil dekripsi dengan menggunakan skema kedua untuk masukan (*input*) banyak *share* yang berbeda lebih baik dibandingkan dengan skema pertama. Hal ini berarti citra hasil dekripsi dengan skema kedua lebih identik dengan citra pesan awal karena nilai NCC yang diperoleh lebih mendekati 1.

Program aplikasi dapat menghasilkan citra pesan kembali setelah dilakukan proses *extracting* dan dekripsi dan nilai NCC yang diperoleh dengan skema kedua menunjukkan bahwa citra pesan tersebut identik dengan citra pesan awal serta nilai PSNR dari *stego image* yang dihasilkan dari proses *embedding* lebih dari 30 *dB*. Oleh karena itu, program dinyatakan berhasil.

4. KESIMPULAN

Penggabungan teknik kriptografi visual *Secret Sharing* dan steganografi *Enhanced Least Significant Bit* mampu meningkatkan keamanan dan kerahasiaan citra *grayscale* yang dikirim. Citra *grayscale* dienkripsi dengan kriptografi visual *Secret Sharing* menjadi citra abstrak dan disembunyikan ke dalam citra lain menggunakan steganografi *Enhanced Least Significant Bit* kemudian dikirimkan kepada pihak penerima. Hal ini membuat pihak-pihak yang tidak berwenang kesulitan untuk mengakses informasi yang tersembunyi karena mereka tidak hanya harus memahami teknik steganografi yang digunakan, tetapi juga harus mampu membalikkan proses kriptografi visual untuk mengembalikan citra *grayscale* asli. Program yang dihasilkan dari penggabungan kedua teknik tersebut selain dapat mengenkripsi dan meng-*embed*, mengektraksi dan mendekripsi citra, tetapi juga dapat mengecek kualitas dari *stego image* dengan menghitung nilai PSNR-nya dan mengetahui keidentikan dari citra hasil proses dekripsi dengan citra pesan awal dengan menghitung nilai NCC-nya.

5. DAFTAR PUSTAKA

Alqadi, Z., Zahran, B., Jaber, Q., Ayyoub, B., & Al-Azzeh, J. (2019). Enhancing the capacity of LSB method by introducing LSB2Z method. *International Journal of Computer Science and Mobile Computing*, *8*(3), 76-90.

Azanuddin, A., Yakub, S., & Prayudha, J. (2022). Implementasi keamanan citra menggunakan algoritma AES-128 dengan aplikasi Client-Server. *Jurnal Riset Sistem Informasi dan Teknik Informatika*, *7*(1), 51-61.

Bunga, P., Pella, S., & Odja, M. (2023). Implementasi secret sharing berbasis kriptografi visual skema (k, n) pada citra biner menggunakan GUI Matlab. *TRANSISTOR Elektro dan Informatika*, *5*(1), 14-20.

Hore, A., & Ziou, D. (2010). Image Quality Metrics: PSNR vs. SSIM. In *2010 20th international conference on pattern recognition* (pp. 2366-2369). IEEE.

Kaso, A. (2018). Computation of the normalized cross-correlation by fast fourier transform. *PloS one*, *13*(9), e0203434.

Kliwati, S. (2018). Algoritma deteksi frekuensi DTML menggunakan korelasi silang untuk telekomando wahana terbang. *Jurnal Teknologi Dirgantara*, *14*(1), 1-8.

Kurniasih, F., Marwati, R., & Sispiyati, R. (2023). Penggabungan Affine Cipher dan Least Significant Bit-2 untuk penyisipan pesan rahasia pada gambar. *Jurnal EurekaMatika*, *11*(2), 79-88.

Lutfi, S., & Rosihan, R. (2018). Perbandingan metode steganografi Least Significant Bit dan Most Significant Bit untuk menyembunyikan informasi rahasia ke dalam citra digital. *Jurnal Informatika dan Komputer*, *1*(1), 34–42.

Naor, M., & Shamir, A. (1995). Visual cryptography. In Advances in Cryptology--EUROCRYPT'94: Workshop on the Theory and Application of 38 Cryptographic Techniques Perugia, Italy, May 9–12, 1994 *Proceedings* 13 (pp. 1-12). Springer Berlin Heidelberg.

Pella, S. I., & Lami, H. F. (2021). Implementasi teknik kriptografi visual pada citra keabuan dan berwarna untuk otentikasi pengguna pada transaksi online. *Jurnal Media Elektro*, *10*(2), 65-72.

Rawat, D., & Bhandari, V. (2013). A steganography technique for hiding image in an image using LSB method for 24-bit color image. *International Journal of Computer Applications*, *64*(20), 15-19.

Zahra, A. D., Marwati, R., & Sispiyati, R. (2021). Kriptografi visual pada gambar berwarna (RGB) menggunakan algoritma Elliptic Curve Cryptography. *Jurnal EurekaMatika*, *9*(2), 141-150.