

PENYISIPAN PESAN PADA CITRA DIGITAL DENGAN STEGANOGRAFI MENGGUNAKAN METODE LSB (LEAST SIGNIFICANT BIT) DAN ALGORITMA AFFINE CIPHER

F. Hilyawan¹, Ramadhan Renaldy² dan M. Najiy Yullah³

^{1,2,3}*Jurusan Informatika, Fakultas Teknik dan Informatika, Universitas PGRI Semarang Gedung B Lantai 3, Kampus 1 Jl. Sidodadi Timur 24, Semarang*

E-mail : feldihilyawan27@gmail.com¹, ramadhanrenaldy@upgris.ac.id², najimuhammad67@gmail.com³

Abstrak

Seiring dengan berkembangnya komunikasi digital, meningkat pula ancaman terhadap data dan akses tidak sah, sehingga perlindungan data menjadi sangat penting. Steganografi memungkinkan penyembunyian pesan dalam citra yang tidak tampak secara visual, sementara kriptografi mengenkripsi pesan agar tidak dapat dibaca oleh pihak yang tidak berwenang. Dengan menggabungkan metode LSB (Least Significant Bit) dan algoritma Affine Cipher, penelitian ini bertujuan untuk memberikan pendekatan keamanan berlapis, di mana pesan terlebih dahulu dienkripsi menggunakan algoritma Affine Cipher dan kemudian pesan disisipkan di dalam citra menggunakan metode LSB. Hasil penelitian menunjukkan bahwa metode ini dapat mengamankan pesan rahasia dengan efektif tanpa merusak kualitas citra, sehingga memberikan tingkat perlindungan data yang lebih tinggi. Namun, efektivitas metode ini dapat berkurang dalam kondisi tertentu, seperti kompresi atau manipulasi citra, sehingga lebih cocok digunakan di lingkungan yang membutuhkan perlindungan data tingkat menengah tanpa banyak modifikasi file.

Kata Kunci: Steganografi, Kriptografi, Metode Least Significant Bit (LSB), Algoritma Affine Cipher

I. PENDAHULUAN

Perkembangan teknologi informasi dan komunikasi telah memberikan banyak kemudahan dalam pertukaran data, namun juga diiringi dengan meningkatnya potensi ancaman terhadap keamanan data. Salah satu ancaman terbesar adalah penyadapan dan pencurian. Oleh karena itu, perlindungan terhadap data sangat dibutuhkan. Salah satu solusi untuk mengamankan data adalah dengan menggunakan teknik kriptografi dan steganografi. Kriptografi berfungsi mengenkripsi pesan agar tidak dapat dibaca oleh pihak yang tidak berwenang, sedangkan steganografi menyembunyikan eksistensi pesan tersebut dalam suatu media.

Salah satu metode steganografi yang populer adalah *Least Significant Bit* (LSB), yang memungkinkan penyisipan data dalam sebuah gambar digital, sehingga tidak mudah terdeteksi. Namun, kelemahan metode ini adalah kerentanannya terhadap analisis statistik sederhana yang dapat mengungkapkan adanya penyisipan data. Untuk mengatasi kelemahan tersebut, penggunaan algoritma enkripsi sebelum proses steganografi dapat meningkatkan keamanan data yang disisipkan. Algoritma *Affine Cipher* merupakan salah satu teknik enkripsi yang dapat digunakan untuk mengacak pesan sebelum disembunyikan dalam media digital. Kombinasi antara *Affine Cipher* dan metode LSB menawarkan pendekatan yang lebih aman, sehingga pesan lebih sulit untuk diungkap.

Beberapa penelitian sebelumnya telah mengeksplorasi pendekatan ini, yang menunjukkan bahwa penggabungan metode LSB dan algoritma *Affine Cipher* mampu meminimalisir terjadinya kriptanalisis [1]. Selain itu, aplikasi steganografi yang telah dihasilkan dari implementasi metode LSB (*Least Significant Bit*) dapat digunakan dengan sangat baik untuk menyembunyikan pesan rahasia ke dalam sebuah gambar [2]. Penelitian-penelitian tersebut menunjukkan bahwa penggunaan metode LSB dan algoritma untuk enkripsi dapat menjadi solusi dalam mengamankan pesan rahasia di era digital.

Penelitian ini bertujuan untuk mengembangkan sistem penyisipan pesan pada citra digital menggunakan metode LSB dan dikombinasikan dengan algoritma *Affine Cipher* untuk meningkatkan keamanan data serta memastikan keberhasilan ekstraksi pesan.

II. METODOLOGI PENELITIAN

1. Studi Literatur

Studi literatur digunakan untuk mengumpulkan referensi yang terkait penelitian ini [3]. Referensi didapat dari berbagai sumber seperti jurnal, dokumen, dan artikel.

2. Perancangan Sistem

Pada tahap ini dilakukan perancangan diagram *Flowchart* atau diagram alur kerja sistem serta tampilan antar muka pengguna untuk memudahkan proses implementasi. Alur sistem ini dibagi menjadi dua yaitu, proses encode dan proses decode. Proses encode terdiri dari enkripsi pesan lalu menyisipkannya ke dalam gambar. Proses decode terdiri dari ekstraksi pesan dari gambar lalu mendekripsinya menjadi pesan asli.

3. Implementasi

Tahap ini mencakup pengembangan sistem berdasarkan rancangan yang telah dibuat sebelumnya. Sistem ini dibangun untuk dapat menyisipkan pesan kedalam gambar menggunakan kombinasi algoritma *Affine Cipher* dan metode *Least Significant Bit*. Algoritma *Affine Cipher* digunakan untuk mengenkripsi serta mendekripsi pesan, dan metode *Least Significant Bit* digunakan untuk menyisipkan pesan ke dalam gambar.

4. Pengujian

Tahap ini dilakukan untuk menguji hasil dari sistem yang telah dibuat. Hasil dari sistem ini yaitu gambar yang telah disisipkan pesan. Pengujian ini meliputi perbandingan gambar asli dan gambar hasil steganografi (stego image), serta uji ketahanan stego image seperti cropping dan editing.

III. HASIL DAN PEMBAHASAN

1. Perancangan Sistem

Berikut langkah-langkah dalam pengoprasi sistem steganografi :

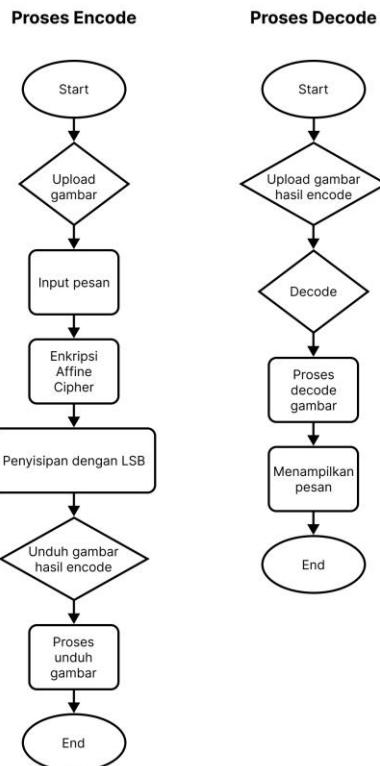
Proses Penyisipan/Encode pada gambar:

1. Mulai.
2. Pengguna memilih Gambar dan upload dari penyimpanan perangkat pengguna.
3. Pengguna memasukkan pesan.
4. Sistem melakukan enkripsi pada pesan menggunakan algoritma *Affine Cipher*.
5. Sistem menyisipkan pesan yang terenkripsi menggunakan metode *Least Significant Bit*.

6. sistem menyediakan gambar hasil encode.
7. Pengguna mengunduh gambar hasil encode.

Proses menampilkan pesan/Decode pada gambar :

1. Pengguna mengupload gambar hasil encode dari perangkat pengguna.
2. Sistem mengekstrak pesan dari dalam gambar menggunakan metode *Least Significant Bit*.
3. Sistem mendekripsi pesan tersebut menggunakan algoritma *Affine Cipher*.
4. Sistem menampilkan pesan asli yang telah didekripsi.



Gambar 1. Diagram *Flowchart encode dan decode*

2. Implementasi Sistem

2.1. Proses Penyisipan Pesan

Sistem ini dirancang untuk menyisipkan pesan menjadi tidak terdeteksi secara visual. Berikut adalah penjelasan proses penyisipan pesannya :

a. Enkripsi Pesan

Algoritma *Affine Cipher* digunakan untuk mengenkripsi pesan dalam sistem ini. Algoritma *Affine Cipher* merupakan salah satu bentuk pengembangan dari *Caesar Cipher*, algoritma ini mengubah setiap huruf dalam pesan asli (*plaintext*) menjadi huruf lain (*ciphertext*) berdasarkan fungsi matematika modular [4]. *Affine Cipher*

memanfaatkan dua kunci yaitu, kunci a dan b. Berikut contoh langkah-langkah untuk mengenkripsi pesan “PGRI” :

Rumus enkripsi :

$$E(x) = (ax + b) \bmod 26 \quad (1)$$

Keterangan :

x = Nilai numerik dari huruf, contoh : A=0, B=1, C=2, dan seterusnya.

a = Kunci a

b = Kunci b

1. Menentukan kunci a dan b

$$a = 5, b = 8$$

2. Mengubah huruf pesan ke angka

$$P = 15, G = 6, R = 17, I = 8$$

3. Mengenkripsi pesan menggunakan rumus algoritma Affine Cipher

$$E(15) = (5 \times 15 + 8) \bmod 26 = 83 \bmod 26 = 5 \text{ (F)}$$

$$E(6) = (5 \times 6 + 8) \bmod 26 = 38 \bmod 26 = 12 \text{ (M)}$$

$$E(17) = (5 \times 17 + 8) \bmod 26 = 93 \bmod 26 = 15 \text{ (P)}$$

$$E(8) = (5 \times 8 + 8) \bmod 26 = 48 \bmod 26 = 22 \text{ (W)}$$

Hasil enkripsinya adalah “FMPW”

b. Penyisipan Pesan yang Telah Terenkripsi ke Dalam Gambar

Pada tahap ini, metode LSB (*Least Significant Bit*) digunakan untuk menyisipkan pesan ke dalam gambar. Metode *Least Significant Bit* merupakan salah satu teknik steganografi yang paling umum digunakan untuk menyembunyikan data dalam media gambar digital. Teknik ini memanfaatkan bit paling terakhir dari setiap *pixel* dalam gambar untuk menyisipkan pesan [5]. Gambar yang dihasilkan akan terlihat sama dengan gambar aslinya, karena dalam proses penggantianya dilakukan pada bit terakhir sehingga tidak akan nampak perbedaan pada gambarnya.

Berikut penjelasan langkah-langkah penyisipan pesan ke dalam gambar :

Gambar yang digunakan yaitu gambar berwarna. Setiap satu pixel terdiri dari komponen RGB (Red, Green, Blue) yang masing-masing memiliki 8 bit, jadi satu pixel total mempunyai 24 bit / 3 byte untuk gambar berwarna.

1. Langkah pertama

Pesan hasil enkripsi yaitu “FMPW” diubah ke satuan ASCII lalu diubah lagi ke bentuk bit

Tabel 1. Konversi huruf ke bentuk bit

Huruf	ASCII	Bit
F	70	01000110
M	77	01001101
P	80	01010000
W	87	01010111

2. Langkah kedua

Bit terakhir dari masing-masing warna RGB (Red, Green, Blue) diganti dengan bit dari pesan. Karena 1 huruf dalam pesannya itu berjumlah 8 bit maka jumlah byte yang dibutuhkan adalah 8 byte, dan jumlah pixel yang dibutuhkan yaitu ada 3 pixel. Untuk lebih jelasnya seperti contoh berikut ini :

Untuk huruf M yang berbentuk bit = 01001101, dimasukkan ke dalam bit terakhir dari masing-masing warna RGB :

Tabel 2. Penyisipan bit pesan ke dalam bit gambar

Pixel 1	Pixel 2	Pixel 3
R = 1001001 <u>0</u>	R = 1001001 <u>0</u>	R = 1001001 <u>0</u>
G = 1001010 <u>1</u>	G = 1001010 <u>1</u>	G = 1001010 <u>1</u>
B = 1000101 <u>0</u>	B = 1000101 <u>1</u>	B = 10001010

Untuk proses selanjutnya juga sama seperti diatas, tinggal melanjutkan saja yaitu dari warna B dan seterusnya. Setelah selesai, pesan telah tersimpan ke dalam gambar dan tidak terdeteksi secara visual.

c. Ekstraksi pesan

Proses ekstraksi pesan juga menggunakan metode LSB (*Least Significant Bit*). Dengan menggunakan metode LSB, sistem membaca byte-byte di dalam gambar, mengambil angka bit yang telah disisipkan sebelumnya, dan merangkainya kembali menjadi bit-bit pesan. Setelah dirangkai, bit-bit pesan diubah kedalam satuan ASCII lalu diubah lagi ke bentuk pesan semula yaitu “FMPW”.

d. Dekripsi pesan yang terenkripsi

Pesan didekripsi menggunakan algoritma *Affine Cipher*.

Rumus dekripsi :

$$D(x) = a^{-1} (x - b) \bmod 26 \quad (2)$$

Keterangan :

x = Nilai numerik dari huruf, contoh : A=0, B=1, C=2, dan seterusnya.

a^{-1} = Invers modular

b = Kunci kedua

1. Mengubah huruf pesan terenkripsi ke bentuk angka

$F = 5, M = 12, P = 15, W = 22$

2. Menentukan a^{-1} yaitu 21

3. Dekripsi pesan

$$D(5) = 21 \times (5-8) \bmod 26 = 21 \times (-3) \bmod 26 = 15 \text{ (P)}$$

$$D(12) = 21 \times (12-8) \bmod 26 = 21 \times 4 \bmod 26 = 6 \text{ (G)}$$

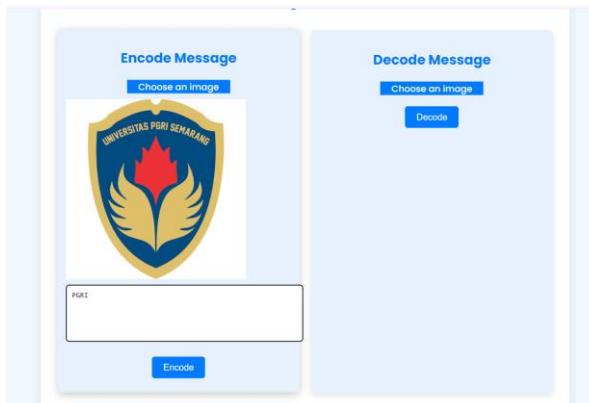
$$D(15) = 21 \times (15-8) \bmod 26 = 21 \times 7 \bmod 26 = 17 \text{ (R)}$$

$$D(22) = 21 \times (22-8) \bmod 26 = 21 \times 14 \bmod 26 = 8 \text{ (I)}$$

Hasil dekripsinya yaitu “PGRI”

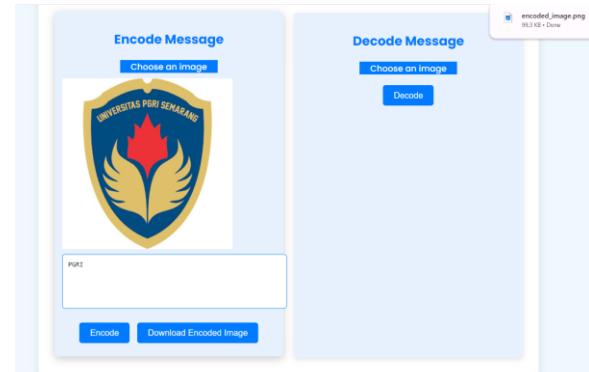
2.2. Tampilan Sistem

- Mengupload gambar dan input pesan untuk di encode



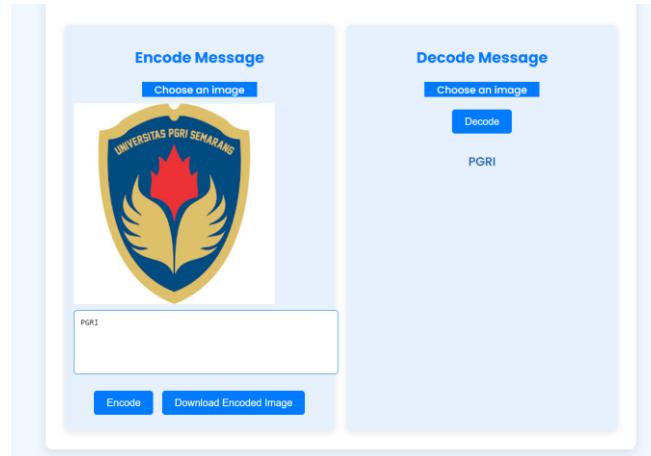
Gambar 2. Proses encode

- Mengunduh gambar hasil encode



Gambar 3. Unduh gambar

- c. Mengupload gambar hasil encode untuk di decode



Gambar 4. Hasil decode

2.3. Pengujian

Tahap ini dilakukan pengujian terhadap gambar yang telah disisipkan pesan menggunakan kombinasi algoritma *Affine Cipher* dan metode LSB (*Least Significant Bit*).

a. Perbandingan Gambar Asli dan *Stego Image*



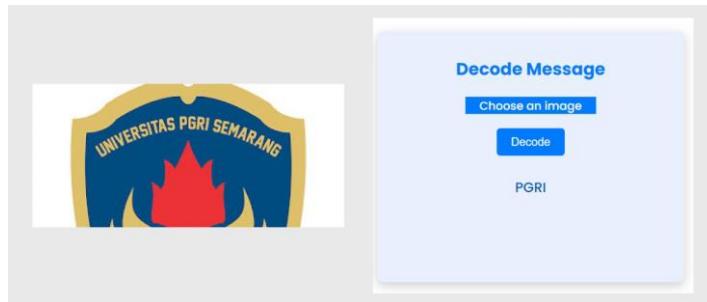
Gambar 5. Perbandingan gambar asli dan *stego image*

Terlihat tidak ada perbedaan antara kedua gambar tersebut, karena perubahan pada bit ini memiliki dampak yang sangat kecil pada tampilan visual gambar, dan pesan yang disisipkan tidak mudah terdeteksi secara visual oleh mata manusia.

b. Uji Ketahanan *Stego Image*

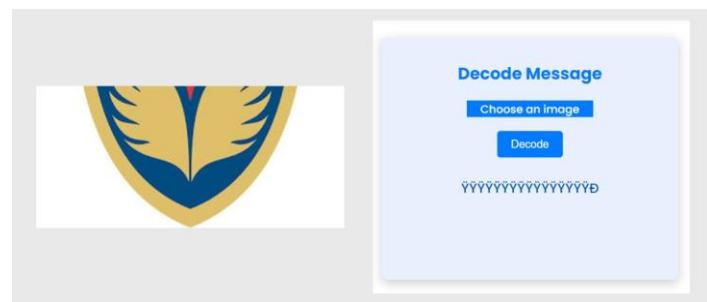
Stego image diuji ketahanannya dengan di *cropping* dan ditingkatkan kecerahannya.

1. Cropping bawah dan atas



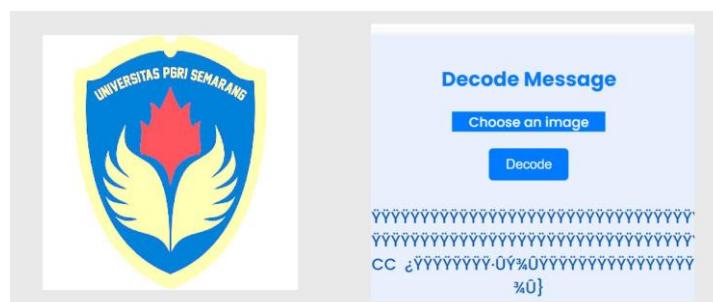
Gambar 6. *Cropping* bawah stego image

Ketika *stego image* dipotong bagian bawahnya, lalu di decode untuk menampilkan pesan yang disisipkan, pesan asli akan muncul karena pesan disisipkan pada bagian atas gambar. Sebaliknya, pesan asli tidak akan muncul ketika *stego image* dipotong bagian atasnya seperti gambar dibawah ini :



Gambar 7. *Cropping* atas stego image

2. *Stego image* ditingkatkan kecerahannya



Gambar 8. Uji kecerahan *Stego image*

Ketika *stego image* di uji kecerahannya, lalu di decode, maka pesan asli tidak akan muncul, karena warna dari *stego image* berubah sehingga bit pesan yang disisipkan di dalamnya juga berubah.

IV. KESIMPULAN

Berdasarkan hasil penelitian, penyisipan pesan pada citra digital dengan steganografi menggunakan metode LSB (*Least Significant Bit*) dan algoritma *Affine Cipher* menunjukkan bahwa kombinasi ini dapat digunakan untuk menyisipkan pesan dalam gambar tanpa terlihat secara visual oleh mata manusia. Namun, terdapat beberapa kelemahan yaitu ketika *stego image* dimodifikasi seperti ditingkatkan kecerahannya dan dipotong bagian atasnya, lalu ketika hasil modifikasi *stego image* di decode, maka pesan asli tidak akan muncul.

VI. REFERENSI

- [1] Kurniasih, F., Mawarti, R., & Sispiyati, R. (2023). Penggabungan *Affine Cipher* dan *Least Significant Bit-2* untuk Penyisipan Pesan Rahasia pada Gambar. *Jurnal EurikaMatika*, 11(2), 79-88.
- [2] Malese, L, P. (2021). Penyembunyian Pesan Rahasia Pada Citra Digital dengan Teknik Steganografi Menggunakan Metode Least Significant Bit (LSB). *Jurnal Ilmiah Wahana Pendidikan*, 7(5), 343-354.
- [3] Al Jum'ah, M, N., & Sarimuddin. (2024). Implementasi Steganografi Metode Least Significant Bit (LSB) untuk Menyembunyikan File Pesan dalam Gambar. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 6(1), 102-108.
- [4] Zalukhu, Y, P, N., Waruwu, F, T., & Siburian, H, K. (2024). Penyisipan Pesan Terenkripsi *Affine Cipher* Pada Citra Digital Dengan Menggunakan Metode Pixel Value Differencing. *KETIK : Jurnal Informatika*, 1(4), 22-33.
- [5] Nur'aini, S. (2019). Steganografi Pada Digital Image Menggunakan Metode Least Significant Bit Insertion. *Walisongo Journal of Information Technology*, 1(1), 75-90.