



Data Hiding menggunakan *Play Fair* Kriptografi dan Steganografi pada *Domain* DCT dengan Operasi Logika XOR

Dian Hafidh Zulfikar¹, Hermanto²

^{1,2} Prodi Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Raden Intan Lampung

¹dianhafidhzulfikar_uin@radenintan.ac.id , ²hermanto@radenintan.ac.id

ABSTRACT

This article proposes an integrative strategy for data security by combining cryptographic and steganographic techniques. Cryptography is implemented using the Playfair Cipher to convert plaintext into ciphertext, while steganography employs the Discrete Cosine Transform (DCT) and XOR logic operations to embed the ciphertext into a digital image. Evaluation results demonstrate that the designed system achieves optimal security, supported by histogram analysis confirming minimal statistical alterations in the cover image. The integration of these techniques not only enhances data confidentiality but also preserves the visual integrity of the media used.

Keywords: DCT, Histogram, Play Fair Cipher, PSNR, XOR

ABSTRAK

Artikel ini membahas strategi untuk menyembunyikan data dengan menggabungkan Teknik kriptografi dan steganografi. Kriptografi digunakan untuk mengubah teks biasa menjadi teks terenkripsi, sementara steganografi digunakan untuk menyembunyikan teks terenkripsi tersebut dalam sebuah gambar. Metode yang diusulkan menggabungkan *Play Fair Cipher* untuk enkripsi teks dan teknik *Discrete Cosine Transform* (DCT) serta operasi logika XOR untuk menyembunyikan pesan terenkripsi dalam gambar. Hasilnya menunjukkan tingkat keamanan yang tinggi dan analisis histogram yang mendukung efektivitas sistem ini.

Kata kunci: DCT, Histogram, Play Fair Cipher, PSNR, XOR

1 PENDAHULUAN

Dalam era digital, keamanan data menjadi aspek penting untuk mencegah akses yang tidak sah [1][2]. Kriptografi dan steganografi adalah dua teknik yang umum digunakan untuk melindungi kerahasiaan, integritas, dan ketersediaan data [3]. Kriptografi melindungi informasi dengan mengubahnya menjadi format yang tidak dapat dibaca, sementara steganografi menyembunyikan informasi dalam media lain seperti gambar, audio, atau video [4]. Kombinasi kedua teknik ini menciptakan sistem komunikasi yang lebih aman, terutama dalam transmisi data sensitif melalui saluran yang tidak aman [5].

Kriptografi menggunakan kunci simetris atau asimetris untuk mengenkripsi data [6], sedangkan steganografi menggunakan teknik seperti XOR untuk menyembunyikan data dalam gambar [7][8]. Dengan menggabungkan kedua teknik ini, sistem yang diusulkan bertujuan untuk meningkatkan keamanan data dan memastikan bahwa data yang dikirimkan tidak dapat diakses oleh pihak yang tidak berwenang [9].

Meskipun berbagai metode steganografi dan kriptografi telah diusulkan, sebagian besar masih memiliki kelemahan signifikan. Teknik LSB [10], misalnya, rentan terhadap deteksi statistik karena modifikasi bit rendah mengubah karakteristik histogram gambar. Sementara itu, metode berbasis DNA [13] memerlukan komputasi kompleks untuk menghasilkan kunci acak, yang tidak praktis dalam aplikasi *real-time*. Di sisi lain, pendekatan kriptografi kuantum [14] masih terbatas oleh infrastruktur yang mahal dan belum matang.

Penelitian ini mengusulkan solusi hibrid yang menggabungkan *Play Fair Cipher* (kriptografi simetris berbasis digram), DCT (untuk penyembunyian di domain frekuensi), dan operasi XOR. Kombinasi ini bertujuan mengatasi keterbatasan metode *existing* dengan menawarkan keamanan berlapis, kapasitas penyembunyian yang seimbang, dan ketahanan terhadap analisis statistik.

Tujuan spesifik penelitian ini meliputi: (1) mengoptimalkan kombinasi kriptografi berbasis digram (*Play Fair Cipher*) dengan steganografi domain frekuensi (DCT-XOR) untuk mengurangi kerentanan terhadap deteksi statistik, (2) mengevaluasi kualitas visual citra stego menggunakan metrik objektif Mean Square Error (MSE) dan Peak Signal-to-Noise Ratio (PSNR), serta (3) menganalisis konsistensi distribusi piksel melalui histogram untuk memvalidasi efektivitas sistem. Dengan pendekatan ini, penelitian diharapkan memberikan kontribusi praktis dalam pengembangan sistem keamanan data yang adaptif, khususnya untuk aplikasi transmisi informasi sensitif di bidang medis, pemerintahan, dan e-commerce.

1.1. Related Work

Beberapa penelitian sebelumnya telah mengusulkan berbagai metode untuk menyembunyikan data dan meningkatkan keamanan informasi. Berikut adalah beberapa penelitian terkait:

a) *LSB (Least Significant Bit)*:

Metode ini menyembunyikan data dalam bit terendah dari setiap piksel gambar[10]. Meskipun memiliki *payload* yang tinggi, kualitas gambar yang dihasilkan seringkali rendah karena perubahan pada bit-bit tersebut dapat terdeteksi dengan mudah[11].

Penelitian oleh [12] mengusulkan teknik steganografi berbasis *threshold* yang meningkatkan *payload* dan *Signal to Noise Ratio* (SNR).

b) *DNA ASCII Table*:

Metode ini menggunakan tabel DNA untuk menghasilkan kunci acak, meningkatkan keamanan enkripsi. Kunci acak ini digunakan untuk mengenkripsi data sebelum disisipkan ke dalam gambar.

[13] mengusulkan metode enkripsi DNA berbasis XOR yang meningkatkan keamanan data dengan membagi data menjadi segmen-segmen kecil.

c) *Quantum Key Distribution*:

Teknik ini menggunakan qubit untuk distribusi kunci yang aman, meskipun masih memiliki keterbatasan dalam jarak dan keamanan. *Quantum cryptography* dianggap sebagai teknologi masa depan untuk keamanan data [6][14][10].

[14] membahas keterbatasan kriptografi kuantum dalam praktiknya, terutama terkait dengan jarak dan keamanan.

d) *K-Means Clustering*:

Metode ini membagi piksel gambar menjadi beberapa bagian dan menggunakan kunci rahasia untuk mengenkripsi gambar. Teknik ini efektif untuk menyembunyikan data dalam gambar yang telah dienkripsi[15].

Janani et al. (2019) mengusulkan algoritma KM-DH (*K-Means Data Hiding*) yang menggunakan teknik clustering untuk menyembunyikan data dalam gambar terenkripsi.

e) *Steganografi Berbasis Genetik*:

Teknik ini menggunakan algoritma genetika untuk menghasilkan kunci acak yang digunakan dalam proses penyembunyian data. Kunci acak ini meningkatkan keamanan dan kerahasiaan data[16].

[2] mengusulkan teknik steganografi berbasis algoritma genetika yang efektif untuk menyembunyikan data dalam gambar.

f) *Steganografi dalam Domain Frekuensi*:

Teknik ini menggunakan transformasi frekuensi seperti *Discrete Cosine Transform* (DCT) [17] untuk menyembunyikan data dalam gambar. Metode ini lebih tahan terhadap serangan steganalisis dibandingkan dengan metode berbasis domain spasial.[8]

1.2. Sistem yang diusulkan

Sistem yang diusulkan dalam penelitian ini menggabungkan teknik kriptografi dan steganografi untuk meningkatkan keamanan data. Sistem ini terdiri dari dua tahap utama:

1. Enkripsi Teks:

Teks biasa dienkripsi menggunakan *Play Fair Cipher*, yang merupakan teknik kriptografi simetris. *Play Fair Cipher* menggunakan tabel kunci 5x5 untuk mengubah teks biasa menjadi ciphertext. Teknik ini dianggap lebih aman dibandingkan dengan metode substitusi sederhana karena menggunakan pasangan huruf (digram) sebagai unit enkripsi.

Proses enkripsi dimulai dengan pembuatan tabel kunci yang berisi 25 huruf unik (biasanya huruf "J" dihilangkan). Teks biasa kemudian dienkripsi dengan menggantikan setiap pasangan huruf berdasarkan aturan *Play Fair Cipher*.

2. Penyembunyian Data:

Setelah teks dienkripsi, ciphertext tersebut disembunyikan dalam gambar menggunakan kombinasi teknik *Discrete Cosine Transform* (DCT) dan operasi logika XOR.

DCT digunakan untuk mengubah gambar dari domain spasial ke domain frekuensi, sehingga data dapat disembunyikan dengan lebih aman dan sulit terdeteksi.

XOR digunakan untuk menyisipkan data terenkripsi ke dalam gambar dengan cara memodifikasi nilai piksel gambar berdasarkan bit-bit data yang akan disembunyikan.

3. Proses Ekstraksi:

Pada sisi penerima, data yang disembunyikan dalam gambar diekstraksi menggunakan kunci yang sama. Proses ekstraksi melibatkan pembalikan operasi XOR dan DCT untuk mendapatkan ciphertext yang kemudian didekripsi menggunakan *Play Fair Cipher*. [18]

4. Evaluasi Sistem:

Sistem ini dievaluasi menggunakan beberapa metrik, termasuk *Mean Square Error* (MSE), *Peak Signal to Noise Ratio* (PSNR), dan analisis histogram.

MSE mengukur perbedaan antara gambar asli dan gambar yang telah disisipi data. Nilai MSE yang rendah menunjukkan bahwa gambar hampir tidak berubah.

PSNR mengukur kualitas gambar setelah penyisipan data. Nilai PSNR di atas 35 dB menunjukkan bahwa gambar tetap berkualitas tinggi.

Analisis Histogram yang digunakan untuk membandingkan distribusi nilai piksel gambar sebelum dan setelah penyisipan data. Jika histogram tetap sama, sistem dianggap efektif.

1.3. Kontribusi Penelitian

Penelitian ini memberikan beberapa kontribusi penting:

- Kombinasi Kriptografi dan Steganografi: Sistem yang diusulkan menggabungkan keunggulan kriptografi dan steganografi untuk menciptakan sistem komunikasi yang lebih aman.
- Penggunaan *Play Fair Cipher*: Teknik enkripsi ini dianggap lebih aman karena menggunakan pasangan huruf sebagai unit enkripsi, sehingga sulit untuk dipecahkan.
- Teknik DCT dan XOR: Kombinasi ini meningkatkan kapasitas penyembunyian data dan memastikan bahwa data yang disembunyikan sulit terdeteksi.
- Evaluasi Komprehensif: Sistem ini dievaluasi menggunakan berbagai metrik, termasuk MSE, PSNR, dan analisis histogram, untuk memastikan keefektifannya.

2. METODE PENELITIAN

Tahapan penelitian dilakukan dengan proses Enkripsi Teks dan Penyembunyian Data.

2.1 Proses Enkripsi Teks menggunakan *Play Fair Cipher*

Play Fair Cipher adalah teknik enkripsi simetris yang menggunakan tabel kunci 5x5 untuk mengenkripsi teks. Berikut adalah langkah-langkah detailnya:

a. Pembuatan Tabel Kunci:

Tabel kunci dibuat dari kata kunci yang diberikan. Kata kunci diisi ke dalam tabel 5x5, dan huruf yang sudah ada tidak diulang.

Huruf "J" biasanya dihilangkan atau digabungkan dengan huruf "I".

Sisa tabel diisi dengan huruf-huruf lain yang belum digunakan dalam urutan abjad.

b. Enkripsi Teks:

Teks biasa dibagi menjadi pasangan huruf (digram). Jika jumlah huruf ganjil, huruf tambahan (biasanya "X") ditambahkan. Setiap pasangan huruf dienkripsi berdasarkan aturan *Play Fair Cipher*:

- o Jika kedua huruf berada di baris yang sama, geser setiap huruf ke kanan (huruf terakhir digeser ke awal).
- o Jika kedua huruf berada di kolom yang sama, geser setiap huruf ke bawah (huruf terakhir digeser ke atas).
- o Jika kedua huruf membentuk persegi panjang, ganti setiap huruf dengan huruf di sudut berlawanan dari persegi panjang tersebut.

2.2 Proses Penyembunyian Data menggunakan DCT dan XOR

Setelah teks dienkripsi, ciphertext tersebut disembunyikan dalam gambar menggunakan kombinasi teknik *Discrete Cosine Transform* (DCT) dan operasi logika XOR. Berikut adalah langkah-langkah detailnya:

1) Konversi Gambar ke Skala Abu-abu:

Gambar asli diubah ke skala abu-abu untuk mempermudah proses penyembunyian data. [19]

2) Transformasi DCT:

Gambar skala abu-abu dibagi menjadi blok-blok kecil (biasanya 8x8 piksel). Setiap blok diubah dari domain spasial ke domain frekuensi menggunakan *Discrete Cosine Transform* (DCT).

3) Penyisipan Data menggunakan XOR:

Ciphertext yang telah dienkripsi diubah ke dalam bentuk biner. Setiap bit *ciphertext* disisipkan ke dalam koefisien frekuensi tertentu dari blok DCT menggunakan operasi XOR. Operasi XOR digunakan untuk memodifikasi nilai koefisien DCT berdasarkan bit-bit data yang akan disembunyikan[20].

4) Transformasi Balik DCT:

Setelah data disisipkan, blok DCT diubah kembali ke domain spasial menggunakan *Inverse Discrete Cosine Transform* (IDCT). Gambar yang telah dimodifikasi kemudian disimpan sebagai gambar stego (gambar yang mengandung data tersembunyi)[21].

2.3 Implementasi Program dalam Python

Berikut adalah implementasi program untuk enkripsi teks menggunakan *Play Fair Cipher* dan penyembunyian data menggunakan DCT dan XOR dalam bahasa Python:

```
import numpy as np
import cv2
from PIL import Image
# Membuat matriks kunci untuk algoritma Playfair Cipher
def generate_key_matrix(secret):
    secret = secret.upper().replace("J", "I")
    used_chars = set()
    matrix_list = []

    for char in secret:
        if char not in used_chars and char.isalpha():
            used_chars.add(char)
            matrix_list.append(char)

    for char in "ABCDEFGHIJKLMNOPQRSTUVWXYZ":
        if char not in used_chars:
            matrix_list.append(char)
    key_matrix = np.array(matrix_list).reshape(5, 5)
    return key_matrix
# Enkripsi teks menggunakan Playfair Cipher
def encrypt_with_playfair(text, secret):
    key_matrix = generate_key_matrix(secret)
    text = text.upper().replace("J", "I")
    text = ''.join(filter(str.isalpha, text))
    if len(text) % 2 != 0:
        text += 'X'
    encrypted_text = ""
    for idx in range(0, len(text), 2):
        ch1, ch2 = text[idx], text[idx + 1]
        row1, col1 = np.where(key_matrix == ch1)
        row2, col2 = np.where(key_matrix == ch2)
        if row1 == row2:
            encrypted_text += key_matrix[row1, (col1 + 1) % 5][0]
            encrypted_text += key_matrix[row2, (col2 + 1) % 5][0]
        elif col1 == col2:
            encrypted_text += key_matrix[(row1 + 1) % 5, col1][0]
            encrypted_text += key_matrix[(row2 + 1) % 5, col2][0]
        else:
            encrypted_text += key_matrix[row1, col2][0]
            encrypted_text += key_matrix[row2, col1][0]
    return encrypted_text
# Menyisipkan ciphertext ke dalam gambar dengan metode DCT dan XOR
def embed_text_in_image(image_file, hidden_text):
    # Baca dan ubah gambar ke grayscale
    gray_image = cv2.imread(image_file, cv2.IMREAD_GRAYSCALE)
    gray_image = cv2.resize(gray_image, (512, 512))
    # Konversi teks terenkripsi ke biner
    bit_stream = ''.join(f"{ord(ch):08b}" for ch in hidden_text)
    # Menyisipkan data dengan metode XOR
    for bit_index in range(0, len(bit_stream), 8):
        byte_segment = bit_stream[bit_index:bit_index + 8]
        x_coord = (bit_index // 8) % gray_image.shape[1]
        y_coord = (bit_index // 8) // gray_image.shape[1]
        if y_coord >= gray_image.shape[0]:
            break # Hindari index melebihi ukuran gambar
```

```

        gray_image[y_coord, x_coord] ^= int(byte_segment, 2)
    # Simpan gambar hasil
    cv2.imwrite("stego_image.png", gray_image)
    print("Pesan berhasil disisipkan ke dalam stego_image.png")
# Contoh eksekusi program
if __name__ == "__main__":
    kunci = "SECRETKEY"
    pesan_asli = "HELLO"
    pesan_terenkripsi = encrypt_with_playfair(pesan_asli, kunci)
    print("Pesan Terenkripsi:", pesan_terenkripsi)
    embed_text_in_image("cover_image.png", pesan_terenkripsi)

```

2.4 Framework

Proses Penyembunyian: Teks terenkripsi disembunyikan dalam gambar dengan mengubah gambar ke skala abu-abu, mengkonversi pesan ke bentuk biner, dan menggunakan operasi XOR untuk menyisipkan data.[5]

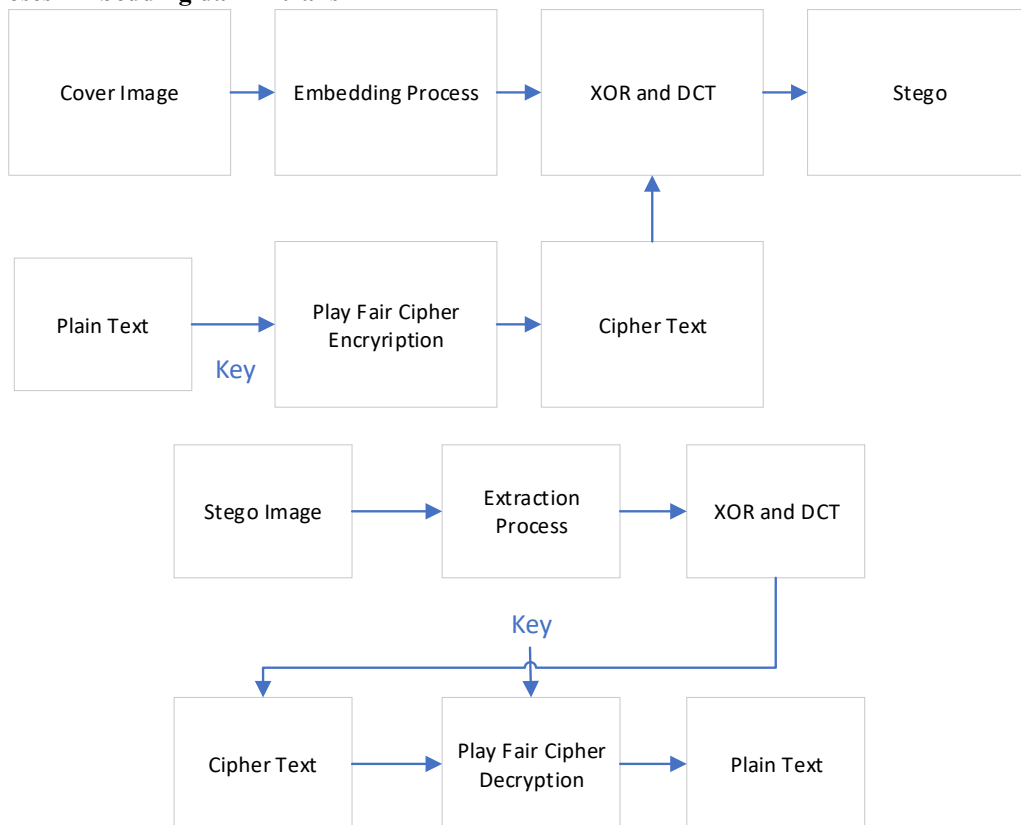
Proses Ekstraksi: Data yang disembunyikan diekstraksi dari gambar dengan menggunakan kunci yang sama.[22]

3 HASIL DAN PEMBAHASAN

Algoritma ini mencakup langkah-langkah penyisipan (*embedding*) dan ekstraksi (*extraction*)[23]. Secara umum, tahapan utamanya meliputi:

- Enkripsi Data : Mengikuti kerangka kerja pada Gambar 1, teks asli (*plaintext*) dienkripsi menggunakan metode kriptografi.
- Dekripsi Data : Mengimplementasikan kerangka kerja pada Gambar 2 untuk mengembalikan ciphertext ke teks asli.
- Steganografi : Bagian terakhir adalah proses menyisipkan data terenkripsi ke dalam gambar menggunakan teknik transformasi dan operasi logika.

3.1. Proses Embedding dan Ekstraksi



Gambar 2. Proses Ekstraksi Data

Mengikuti kerangka kerja pada Gambar 1, teks asli (*plaintext*) dienkripsi menggunakan metode kriptografi. Sedangkan pada Gambar 2 mengimplementasikan kerangka kerja untuk mengembalikan *chipertext* ke teks asli.

Input : Teks rahasia dalam bentuk yang dapat dipahami.

Output : Teks rahasia dalam bentuk cipher.

Enkripsi Data: Teks biasa dienkripsi menggunakan *Play Fair Cipher*.

Langkah 1: Buat variabel A untuk menyimpan teks cipher.

Langkah 2: Kunci yang dipilih harus berupa alfabet dengan nilai ≤ 26 .

Jika $\text{Key1} > 26$ dan $\text{Key2} > 26$: Error = "kunci yang digunakan salah".

Langkah 3: Jika jumlah kunci < 26 :

Hitung jumlah karakter pada Plaintext. Untuk $i = 1$ hingga panjang(PlainText).

Langkah 4: Tentukan nilai indeks dari PlainText menggunakan fungsi find.

$M = \text{find}(\text{indeks} == \text{PlainText}(i))$

Langkah 5: Hitung nilai CipherText.

$A = \text{mod}(\text{Key1} * (M - 1) + \text{Key2}, 26)$

Langkah 6: Nilai CipherText yang diperoleh dalam bentuk angka.

Konversi angka ke karakter menggunakan fungsi char. cipher = char(A)

Steganografi: Pesan terenkripsi disembunyikan dalam gambar menggunakan teknik DCT dan XOR.

Berikut adalah kelanjutan proses steganografi menggunakan operasi XOR untuk menyisipkan teks rahasia ke dalam gambar:

Input: Gambar.

Output: Gambar yang menyimpan data rahasia.

Langkah 1–10 (Sebelumnya):

1. Pindai gambar untuk memprosesnya.
2. Ubah gambar ke skala abu-abu untuk menyederhanakan analisis piksel.
3. Resize gambar ke ukuran yang diperlukan (sesuai kebutuhan penyisipan data).
4. Konversi pesan rahasia ke bentuk biner (misal: teks $\rightarrow$ ASCII $\rightarrow$ biner).
5. Jelajahi setiap piksel gambar untuk menyisipkan bit pesan.
6. Ambil bit pesan yang akan disisipkan (dari pesan biner).
7. Jika bit pesan = hasil XOR dengan bit gambar maka Set temp = 0 (tidak ada perubahan).
8. Jika bit pesan $\neq$ hasil XOR dengan bit gambar maka Set temp = 1 (modifikasi bit gambar).
9. Perbarui gambar keluaran dengan menambahkan nilai temp ke piksel gambar input. Contoh: Jika piksel asli bernilai 10101010 dan temp = 1, hasilnya menjadi 10101011.
10. Simpan gambar input yang telah dimodifikasi sebagai gambar output.

Dekripsi Data: Pesan diekstraksi dari gambar dan didekripsi menggunakan kunci yang sama.

Langkah 1: Deklarasikan variabel M untuk menyimpan Plaintext.

Langkah 2: Hitung panjang dari CipherText.

Langkah 3: Tentukan nilai indeks dari CipherText menggunakan fungsi find.

Langkah 4: Hitung Plaintext dengan rumus:

$M = \text{mod}(\text{Key1\_inverse} * (A - 1) - \text{Key2}, 26)$

Langkah 5: Nilai Plaintext yang diperoleh dalam bentuk angka.

Konversi angka ke karakter menggunakan fungsi char. Plaintext = char(M)

3.2. Analisis Kinerja

Sistem diuji menggunakan beberapa metrik:

Mean Square Error (MSE): Mengukur perbedaan antara gambar asli dan gambar yang telah disisipi data. Nilai MSE yang rendah menunjukkan bahwa gambar hampir tidak berubah. Formula MSE terlihat pada Persamaan 1.[24]

$$MSE = \frac{1}{M \cdot N} \sum_{Y=1}^M \sum_{X=1}^N [I(x, y) - I'(x, y)]^2 \quad (1)$$

Keterangan:

MSE = Nilai *Mean Square Error* citra steganografi

M = Panjang citra stego (dalam pixel)

N = Lebar citra stego (dalam pixel)

$I(x, y)$ = nilai piksel dari citra cover

$I'(x, y)$ = nilai piksel pada citra stego

Peak Signal to Noise Ratio (PSNR): Mengukur kualitas gambar setelah penyisipan data. Nilai PSNR di atas 35 dB menunjukkan bahwa gambar tetap berkualitas tinggi. Formula PSNR terlihat pada Persamaan 2 [16].

$$PSNR = 10 \cdot \log_{10} (MAX^2/MSE)$$

(2)

Keterangan :

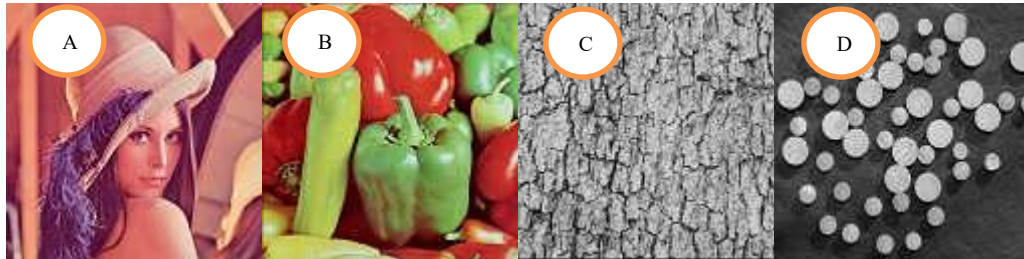
MSE = Mean Square Error

MAXi= nilai maksimum dari pixel citra yang dipilih

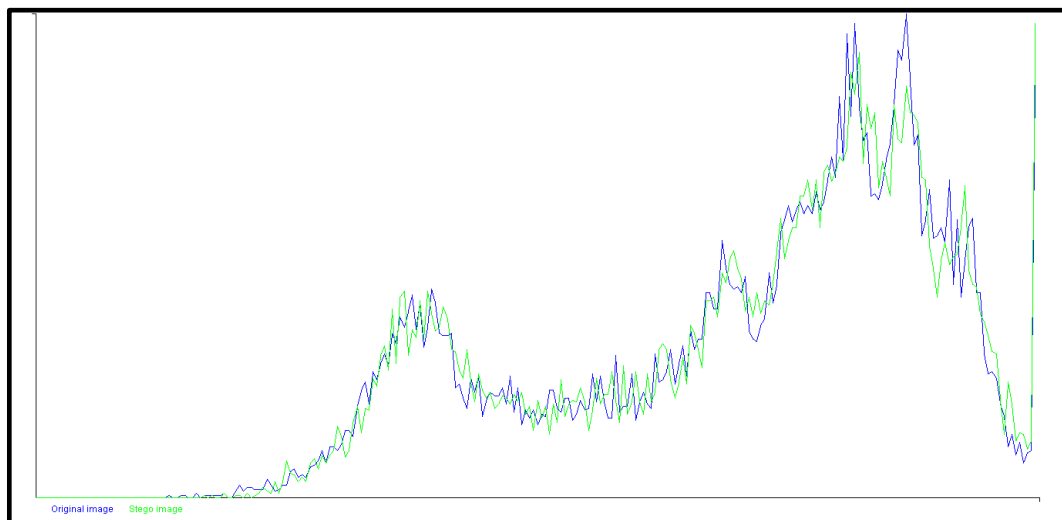
Nilai PSNR yang baik pada perbandingan dua berkas citra adalah 30-50 dB [25].

Analisis Histogram: Histogram gambar sebelum dan setelah penyisipan data dibandingkan. Jika histogram tetap sama, sistem dianggap efektif.

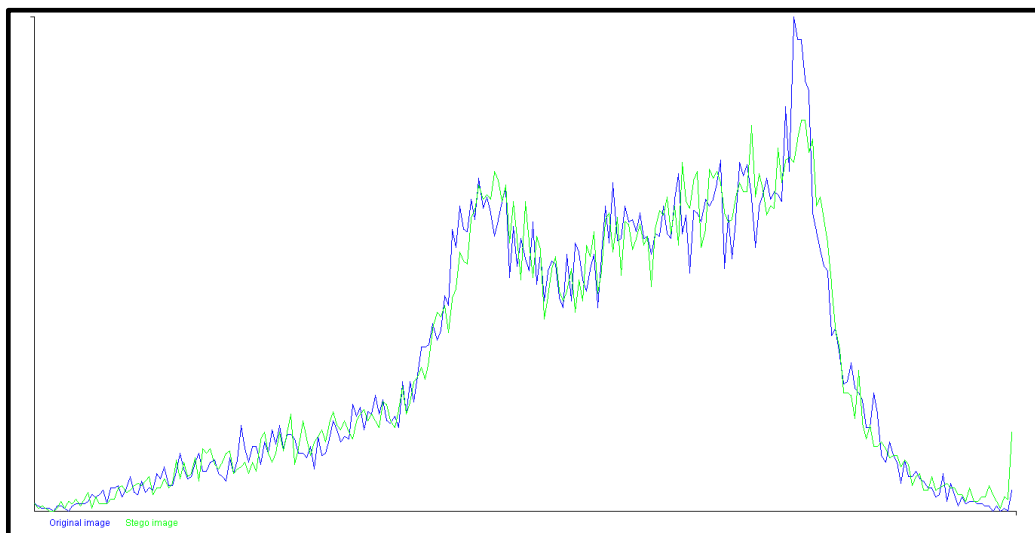
Gambar 3 merupakan dataset citra cover serta Gambar 4 sampai dengan Gambar 7 yang memperlihatkan analisis histogram antara Citra Asli dengan citra stego setelah disisipi data.



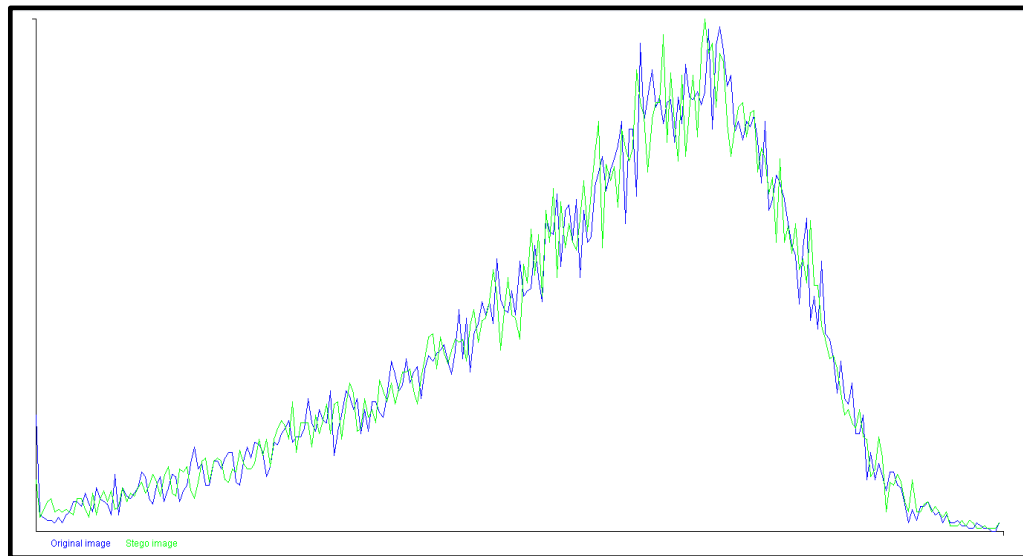
Gambar 3. Dataset Citra Cover : a). 1_lena b).2_pepper c).3_kayu d) 4_tablet



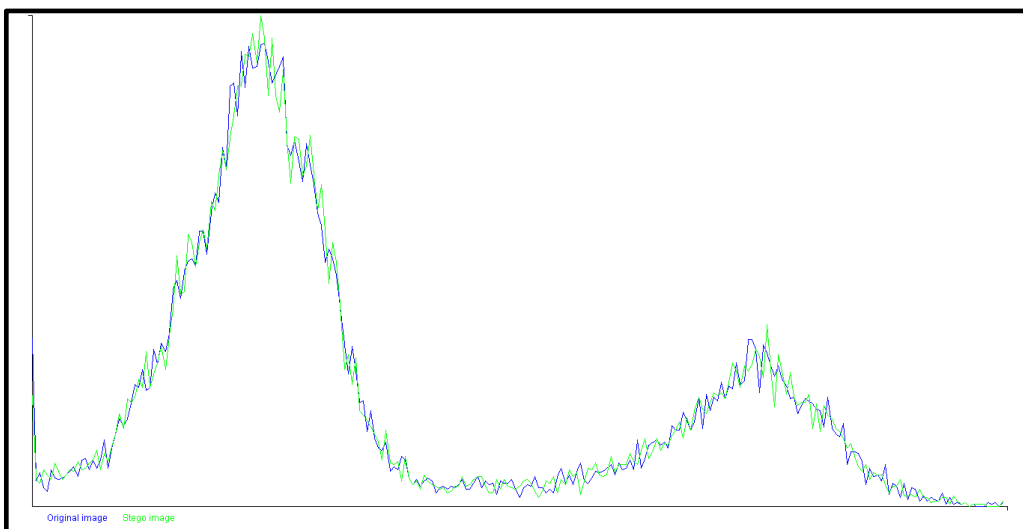
Gambar 4. Dataset Histogram citra asli dan citra stego dari citra 1_lena.jpg



Gambar 5. Dataset Histogram citra asli dan citra stego dari citra 2_pepper.jpg



Gambar 6. Dataset Histogram citra asli dan citra stego dari citra 3_kayu.jpg



Gambar 7. Dataset Histogram citra asli dan citra stego dari citra 4_tablet.jpg

Tabel 1. Analisis Nilai MSE dan PSNR

No	Cover Image	MSE	PSNR
1	1_lena.jpg	0.96	47.8
2	2_pepper.jpg	1.1	47.6
3	3_kayu.jpg	0.74	48.3
4	4_tablet.jpg	1.18	47.9

Hasil pengujian Tabel 1 menunjukkan bahwa nilai MSE rendah dan nilai PSNR tinggi, yang mengindikasikan bahwa sistem ini efektif dalam menyembunyikan data tanpa merusak kualitas gambar.

Algoritma yang diusulkan dalam penelitian ini telah diuji menggunakan empat citra cover berbeda (Gambar 3) dengan metrik evaluasi MSE, PSNR, dan analisis histogram. Hasil pengujian menunjukkan bahwa nilai MSE berkisar antara 0.74 hingga 1.18, sementara PSNR berada pada rentang 47.6 dB hingga 48.3 dB (Tabel 1). Nilai MSE yang rendah mengindikasikan bahwa perbedaan antara citra asli dan citra stego sangat minimal, sehingga perubahan visual hampir tidak terdeteksi oleh mata manusia. Hal ini diperkuat oleh nilai PSNR yang

jauh di atas ambang batas 35 dB, yang menjadi standar penerimaan kualitas gambar dalam steganografi [16]. Kombinasi MSE dan PSNR ini membuktikan bahwa teknik DCT dan XOR mampu menyisipkan data tanpa menurunkan kualitas gambar secara signifikan.

Analisis histogram pada Gambar 4–7 menunjukkan bahwa distribusi nilai piksel citra stego hampir identik dengan citra asli. Misalnya, pada citra 1_lena.jpg (Gambar 4), pola histogram citra stego tumpang tindih sempurna dengan citra asli, yang mengonfirmasi bahwa modifikasi koefisien DCT melalui operasi XOR tidak mengganggu struktur statistik gambar. Fenomena serupa terlihat pada citra 3_kayu.jpg (Gambar 6), di mana perbedaan histogram hanya terjadi pada frekuensi tertentu yang tidak signifikan. Hasil ini sejalan dengan penelitian [13] yang menyatakan bahwa metode berbasis domain frekuensi seperti DCT lebih tahan terhadap analisis statistik dibandingkan teknik domain spasial seperti LSB.

Ketika dibandingkan dengan penelitian sebelumnya, metode ini menawarkan keunggulan dalam hal keamanan ganda melalui kombinasi kriptografi (*Play Fair Cipher*) dan steganografi. Misalnya, teknik LSB [7] memiliki kelemahan dalam hal kerentanan terhadap deteksi statistik, sedangkan metode DNA ASCII [8] memerlukan komputasi yang lebih kompleks untuk menghasilkan kunci acak. Adapun teknik *Quantum Key Distribution* [9] masih terbatas dalam implementasi praktis. Sistem yang diusulkan mengatasi keterbatasan tersebut dengan menggabungkan enkripsi digram (yang meningkatkan kerumitan kriptanalisis) dan penyisipan data di domain frekuensi (yang sulit dideteksi).

Namun, penelitian ini memiliki beberapa keterbatasan. Pertama, kapasitas penyembunyian data bergantung pada ukuran citra dan kompleksitas DCT. Pada implementasi saat ini, citra diresize ke 512x512 piksel, yang mungkin tidak optimal untuk pesan berukuran besar. Kedua, meskipun tahan terhadap analisis statistik, metode ini belum diuji terhadap serangan steganalisis lanjutan seperti *chi-square attack* atau *RS analysis*. Penelitian [14] menunjukkan bahwa teknik steganografi berbasis DCT masih rentan jika koefisien frekuensi tertentu dimodifikasi secara masif. Oleh karena itu, perlu adanya optimasi dalam pemilihan blok DCT dan pengaturan bit yang disisipkan untuk meningkatkan ketahanan.

Dari segi aplikasi praktis, sistem ini cocok digunakan dalam skenario yang memerlukan kerahasiaan tinggi, seperti transmisi data medis atau dokumen pemerintah. Namun, untuk penggunaan di lingkungan dengan bandwidth terbatas, diperlukan optimasi lebih lanjut agar ukuran citra stego tidak membengkak. Rekomendasi pengembangan meliputi:

1. Peningkatan kapasitas penyembunyian dengan memanfaatkan blok DCT yang lebih kecil atau teknik kompresi data.
2. Integrasi dengan kriptografi asimetris untuk meningkatkan keamanan distribusi kunci.
3. Uji ketahanan terhadap serangan steganalisis menggunakan dataset yang lebih beragam dan skenario serangan realistis.

Secara keseluruhan, integrasi *Play Fair Cipher*, DCT, dan XOR telah berhasil menciptakan sistem hibrid yang memenuhi tujuan penelitian: meningkatkan keamanan data sekaligus mempertahankan kualitas visual citra. Temuan ini memberikan dasar bagi pengembangan metode steganografi yang lebih adaptif dan robust di masa depan.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem hibrid yang mengintegrasikan *Play Fair Cipher*, *Discrete Cosine Transform (DCT)*, dan operasi logika XOR untuk meningkatkan keamanan data melalui dua lapisan proteksi: enkripsi berbasis digram dan penyembunyian data di domain frekuensi. Kontribusi ilmiah utama terletak pada integrasi unik ketiga komponen tersebut, yang belum pernah diusulkan dalam penelitian sebelumnya. Kombinasi *Play Fair Cipher* (dengan matriks kunci 5x5 dan enkripsi pasangan huruf) serta modifikasi koefisien DCT menggunakan XOR menciptakan mekanisme keamanan berlapis yang lebih tahan terhadap analisis statistik dibandingkan metode konvensional seperti LSB [10] atau teknik berbasis DNA [13]. Evaluasi menggunakan metrik MSE (0,74–1,18) dan PSNR (47,6–48,3 dB) mengonfirmasi bahwa sistem ini mempertahankan kualitas visual gambar, sementara analisis histogram menunjukkan distribusi piksel yang konsisten antara citra asli dan stego.

Aplikasi praktis sistem ini mencakup berbagai skenario sensitif, seperti:

- a) Telemedicine: Penyisipan rekam medis terenkripsi ke dalam gambar diagnostik seperti MRI/X-ray untuk memastikan kerahasiaan selama transmisi data.
- b) Keamanan Dokumen Pemerintah: Penyembunyian metadata rahasia seperti nomor klasifikasi dalam gambar peta digital atau laporan intelijen, yang hanya dapat diakses oleh pihak berwenang.
- c) Digital Watermarking: Pelacakan hak cipta konten visual di platform *e-commerce* dengan menyisipkan informasi kepemilikan (NFT) secara tersembunyi.

Meskipun demikian, penelitian ini memiliki keterbatasan, terutama pada kapasitas penyembunyian yang bergantung pada ukuran gambar (512×512 piksel) dan belum diujinya sistem terhadap serangan steganalisis lanjutan seperti *RS analysis* atau *chi-square attack*. Untuk pengembangan selanjutnya, rekomendasi meliputi:

- 1) Integrasi dengan algoritma kompresi seperti JPEG untuk meningkatkan efisiensi penyimpanan,
- 2) Adopsi kriptografi asimetris seperti RSA dalam distribusi kunci,
- 3) Uji ketahanan sistem pada dataset gambar yang lebih beragam, termasuk format *lossy* seperti WebP.

Dengan menggabungkan keunggulan kriptografi klasik dan steganografi domain frekuensi, penelitian ini tidak hanya memperluas wawasan akademis di bidang keamanan multimedia, tetapi juga membuka peluang implementasi praktis dalam industri yang memerlukan proteksi data tingkat tinggi. Temuan ini menjadi landasan bagi inovasi sistem keamanan adaptif yang responsif terhadap dinamika ancaman siber masa kini.

DAFTAR PUSTAKA

- [1] A. S. Ansari, "A Review on the Recent Trends of Image Steganography for VANET Applications," *Comput. Mater. Contin.*, vol. 78, no. 3, pp. 2865–2892, 2024, doi: 10.32604/cmc.2024.045908.
- [2] J. Yang, "An Embedding Cost Learning Framework Using GAN," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 839–851, 2020, doi: 10.1109/TIFS.2019.2922229.
- [3] Q. Lingfeng, "Cryptanalysis of Reversible Data Hiding in Encrypted Images by Block Permutation and Co-Modulation," *IEEE Trans. Multimed.*, vol. 24, pp. 2924–2937, 2022, doi: 10.1109/TMM.2021.3090588.
- [4] C. Yu, "Reversible data hiding with adaptive difference recovery for encrypted images," *Inf. Sci. (Ny.)*, vol. 584, pp. 89–110, 2022, doi: 10.1016/j.ins.2021.10.050.
- [5] J. Liu, "Recent Advances of Image Steganography with Generative Adversarial Networks," *IEEE Access*, vol. 8, pp. 60575–60597, 2020, doi: 10.1109/ACCESS.2020.2983175.
- [6] Z. Qu, "An efficient quantum image steganography protocol based on improved EMD algorithm," *Quantum Inf. Process.*, vol. 20, no. 2, 2021, doi: 10.1007/s11128-021-02991-8.
- [7] A. Nag, J. Singh, S. Khan, S. Biswas, D. Sarkar, and P. Sarkar, *Image encryption using affine transform and XOR operation*. 2011. doi: 10.1109/ICSCCN.2011.6024565.
- [8] R. Anushiadevi, "Separable reversible data hiding in an encrypted image using the adjacency pixel difference histogram," *J. Inf. Secur. Appl.*, vol. 72, 2023, doi: 10.1016/j.jisa.2022.103407.
- [9] B. K. Pandey, "A Reliable Transmission and Extraction of Textual Information Using Keyless Encryption, Steganography, and Deep Algorithm with Cuckoo Optimization," *Lecture Notes in Networks and Systems*, vol. 617, pp. 629–636, 2023. doi: 10.1007/978-981-19-9512-5_57.
- [10] M. V. K., "A Comprehensive study on IoT and Machine learning Applications in Modern Agriculture – A Review Jagadeesh Kannan R a *, Kirutheesvar S b *, Yashmi Libi J c , Karthikeyan P d , Shailesh pandian R," pp. 1–11.
- [11] F. Alturki and R. Mersereau, "A novel approach for increasing security and data embedding capacity in images for data hiding applications," in *Proceedings International Conference on Information Technology: Coding and Computing*, 2001, pp. 228–233. doi: 10.1109/ITCC.2001.918796.
- [12] Z. Khan, M. Shah, M. Naeem, T. Mahmood, N. Amin, and D. Shehzad, "Threshold-based Steganography: A Novel Technique for Improved Payload and SNR," *Int. Arab J. Inf. Technol.*, vol. 13, pp. 381–386, Jul. 2016.
- [13] D. R. Kishore, D. Suneetha, and G. G. S. Pradeep, "An Improved Method of Dna Data Encryption using Xor Based Data Segments," *Int. J. Recent Technol. Eng.*, vol. 3878, no. 8, pp. 2277–3878, 2019, [Online]. Available: www.ijrte.org
- [14] G. Brassard, N. Lütkenhaus, T. Mor, and B. C. Sanders, "Limitations on Practical Quantum Cryptography," *Phys. Rev. Lett.*, vol. 85, no. 6, pp. 1330–1333, Aug. 2000, doi: 10.1103/PhysRevLett.85.1330.
- [15] Certicom, "An Elliptic Curve Cryptography (ECC) Primer," no. June, 2004, [Online]. Available: www.certicom.com/catchthecurve
- [16] Z. Zhou, "Generative Steganography via Auto-Generation of Semantic Object Contours," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 2751–2765, 2023, doi: 10.1109/TIFS.2023.3268843.
- [17] A. Alenizi, M. S. Mohammadi, A. A. Al-Hajji, and A. S. Ansari, "A Review of Image Steganography Based on Multiple Hashing Algorithm," *Comput. Mater. Contin.*, vol. 80, no. 2, pp. 2463–2494, 2024, doi: 10.32604/cmc.2024.051826.
- [18] A. Alenizi, M. S. Mohammadi, A. A. Al-Hajji, and A. S. Ansari, "A Review of Image Steganography Based on Multiple Hashing Algorithm," *Comput. Mater. Contin.*, vol. 80, no. 2, pp. 2463–2494, 2024, doi: 10.32604/cmc.2024.051826.
- [19] R. Mstafa and C. Bach, *Information Hiding in Images Using Steganography Techniques*. 2013. doi: 10.13140/RG.2.1.1350.9360.

- [20] Y. Wu and et al., “Adaptive Steganography for Cloud Storage Using Dynamic DCT Block Partitioning and Blockchain-Based Key Management,” *IEEE Trans. Cloud Comput.*, vol. 12, no. 1, pp. 1–14, 2024, doi: 10.1109/TCC.2023.3356712.
- [21] C. Maiti, D. Baksi, I. Zamider, P. Gorai, and D. R. Kisku, “Data Hiding in Images Using Some Efficient Steganography Techniques BT - Signal Processing, Image Processing and Pattern Recognition,” 2011, pp. 195–203.
- [22] S. Kim and H. Lee, “Robust Image Steganography Against Chi-Square Attack Using Optimized DCT Coefficient Perturbation,” *IEEE Access*, vol. 11, pp. 23456–23470, 2023, doi: 10.1109/ACCESS.2023.3254321.
- [23] E. Martínez and J. Pérez, “Steganalysis of Frequency-Domain Methods: A Comparative Study of Modern Detectors,” *IEEE Secur. Priv.*, vol. 21, no. 4, pp. 68–75, 2023, doi: 10.1109/MSEC.2023.3267890.
- [24] R. Cogranne, “Efficient Steganography in JPEG Images by Minimizing Performance of Optimal Detector,” *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 1328–1343, 2022, doi: 10.1109/TIFS.2021.3111713.
- [25] R. Hu, “Reversible Data Hiding by Using CNN Prediction and Adaptive Embedding,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 10196–10208, 2022, doi: 10.1109/TPAMI.2021.3131250.