PROTEKSI *SECRET MESSAGE* PADA CITRA DIGITAL MENGGUNAKAN STEGANOGRAFI DWT-LSB DAN KRIPTOGRAFI *VIGENERE CIPHER*

TUGAS AKHIR

Diajukan Oleh :
NAJLA' HAURA ANDIKO
NIM. 190705045
Mahasiswa Fakultas Sains dan Teknologi
Program Studi Teknologi Informasi

PRODI TEKNOLOGI INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI AR-RANIRY
BANDA ACEH
2023 M / 1445 H

LEMBAR PERSETUJUAN

PROTEKSI *SECRET MESSAGE* PADA CITRA DIGITAL MENGGUNAKAN STEGANOGRAFI DWT-LSB DAN KRIPTOGRAFI *VIGENERE CIPHER*

TUGAS AKHIR

Diajukan Kepada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh
Sebagai Salah Satu Beban Studi Memperoleh Gelar Sarjana
pada Prodi Teknologi Informasi

Oleh:
NAJLA' HAURA ANDIKO
NIM. 190705045
Mahasiswa Fakultas Sains dan Teknologi
Program Studi Teknologi Informasi

Disetujui untuk Dimunaqasyahkan Oleh:

Pembimbing I,	Pembimbing II,
Malahayati, M.T	**Bahagi, M.T**
NIP. 198301272015032003	**NIP. 198802212022031001**

Mengetahui,
Ketua Program Studi Teknologi Informasi

Ima Dwitawati, MBA
NIP. 198210132014032002

LEMBAR PENGESAHAN

PROTEKSI *SECRET MESSAGE* PADA CITRA DIGITAL MENGGUNAKAN STEGANOGRAFI DWT-LSB DAN KRIPTOGRAFI *VIGENERE CIPHER*

TUGAS AKHIR

Telah Diuji Oleh Panitia Ujian Munaqasah Tugas Akhir
Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh dan Dinyatakan Lulus
Serta Diterima Sebagai Salah Satu Beban Studi Program Sarjana (S-1)
Pada Prodi Teknologi Informasi

Pada Hari/Tanggal : 22 Desember 2023
9 Jumadil Akhir 1445 H

di Darussalam, Banda Aceh
Panitia Ujian Munaqasah Tugas Akhir

Ketua,	Sekretaris,
Malahayati, M.T	**Baihaqi, M.T**
NIP. 198301272015032003	**NIP. 198802212022031001**
Penguji I,	Penguji II,
Nazaruddin Ahmad, M.T	**Hendri Ahmadian M.I.M**
NIP. 198206052014031002	**NIP. 198301042014031002**

Mengetahui,
Dekan Fakultas Sains dan Teknologi
UIN Ar-Raniry Banda Aceh

Dr. Ir. M. Dirhamsyah, M.T.,IPU
NIP. 196210021988111001

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama	: Najla' Haura Andiko
NIM	: 190705045
Program Studi	: Teknologi Informasi
Fakultas	: Sains dan Teknologi
Judul Tugas Akhir	: Proteksi *Secret Message* Pada Citra Digital Menggunakan Steganografi DWT-LSB dan Kriptografi *Vigènere Cipher*

Dengan ini menyatakan bahwa dalam penulisan tugas akhir ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggungjawabkan;
2. Tidak melakukan plagiasi terhadap naskah orang lain;
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya;
4. Tidak memanipulasi dan memalsukan data;
5. Mengerjakan sendiri karya ini dan mampu mempertanggungjawab atas karya ini;

Bila kemudian hari ini ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat mempertanggungjawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenakan sanksi berdasarkan aturan yang berlaku di Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh.

Demikian pernyataan ini saya buat dengan sesungguhnya dan tanpa paksaan dari pihak manapun.

Banda Aceh, 18 Desember 2023

Yang Menyatakan

(Najla' Haura Andiko)

ABSTRAK

Nama : Najla' Haura Andiko
NIM 190705045
Program Studi : Teknologi Informasi

Judul : Proteksi *Secret Message* Pada Citra Digital Menggunakan Steganografi DWT-LSB dan Kriptografi *Vigenere Cipher*
Tanggal Sidang : 22 Desember 2023
Jumlah Halaman : 64 Halaman
Pembimbing I : Malahayati, M.T
Pembimbing II : Baihaqi, M.T

Permasalahan pencurian informasi pada dunia digital sangat meresahkan banyak *user*, karena dinilai merugikan dari berbagai aspek. Berdasarkan permasalahan tersebut, masalah ini dapat di atasi salah satunya dengan menyisipkan informasi pada media lain. Fokus penelitian ini terhadap *secret message* berupa teks dan media yang digunakan adalah citra digital. Penelitian ini menggunakan metode steganografi DWT-LSB untuk teknik *embedding* dan menggunakan kriptografi *vigènere cipher* untuk mengenkripsi pesan rahasia.

Setelah dilakukan percobaan terhadap keempat citra, diperoleh nilai PSNR terbaik yaitu 71.72124017690822 dB dengan nilai MSE sebesar 0.004374781260936953. Berdasarkan hasil percobaan, program dapat mengamankan *secret message* dengan baik menggunakan algoritma *Vigènere Cipher* dan DWT-LSB. Jumlah maksimal karakter yang dapat di-*embed* ke dalam citra, tergantung pada resolusi citra tersebut. Semakin tinggi resolusinya, maka semakin banyak karakter yang dapat di-*embed*. *Secret Message* pada *stego-image* dapat diekstrak kembali dan didekripsi ke bentuk aslinya.

Kata Kunci : Steganografi, Kriptografi, DWT-LSB, *Vigènere Cipher*, MSE, PSNR

ABSTRACT

Name : Najla' Haura Andiko
Student ID 190705045
Department : Information Technology

Title : Protection of secret messages in digital images using DWT-LSB Steganography and Vigenère Cipher Cryptography

Date : December, 22nd 2023
Number of Pages : 64 Pages

Supervisor I : Malahayati, M.T
Supervisor II : Baihaqi, M.T

The problem of information theft in the digital world is very troubling to many users, because it is considered detrimental from various aspects. Based on these problems, this problem can be overcome one of them by inserting information in other media. The focus of this research is on the secret message in the form of text and the media used is a digital image. This research uses DWT-LSB steganography method for embedding technique and uses Vigènere cipher cryptography to encrypt the secret message.

After experimenting with the four images, the best PSNR value is 71.72124017690822 dB with an MSE value of 0.004374781260936953. Based on the experimental results, the program can secure the secret message well using the Vigènere Cipher and DWT-LSB algorithms. The maximum number of characters that can be embedded into an image depends on the resolution of the image. The higher the resolution, the more characters that can be embedded. The Secret Message in the stego-image can be extracted and decrypted to its original form.

Keywords : Steganography, Cryptography, DWT-LSB, *Vigènere Cipher*, MSE, PSNR

KATA PENGANTAR

Bismillahirrahmanirrahiim. Segala puji dan syukur selalu kita panjatkan kepada Allah s*ubhanahu wata'ala* yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir dengan judul **"Proteksi *Secret Message* Pada Citra Digital Menggunakan Steganografi Dwt-Lsb Dan Kriptografi *Vigenere Cipher*".** Shalawat serta salam selalu kita panjatkan kepada Rasulullah saw. yang mengantarkan manusia dari zaman kebodohan ke zaman yang penuh dengan ilmu pengetahuan.

Penyusunan tugas akhir ini diajukan sebagai salah satu syarat untuk menyelesaikan pendidikan tingkat sarjana pada Program Studi Teknologi Informasi, Universitas Islam Negeri Ar-Raniry Banda Aceh. Dalam proses penyusunan tugas akhir ini, dengan kerendahan hati penulis ingin mengungkapkan terimasih kepada :

1. Kedua orangtua yang senantiasa memberikan dukungan dan doa kepada saya dalam menyelesaikan tugas akhir ini.
2. Keluarga dan sahabat yang selalu menyemangati dan membantu penulis untuk menyelesaikan tugas akhir ini dari awal hingga akhir.
3. Bapak Dekan Fakultas Sains dan Teknologi Dr. Ir. Muhammad Dirhamsyah, M.T., IPU yang selalu mendukung dan memberi motivasi untuk kami.
4. Ibu Malahayati, M.T sebagai dosen pembimbing pertama dan Bapak Baihaqi, M.T sebagai dosen pembimbing kedua, yang telah meluangkan waktu dan mencurahkan pemikirannya dalam membimbing penulis untuk menyelesaikan tugas akhir ini.
5. Ibu Cut Ida Rahmadiana, S.Si selaku Staf Prodi Teknologi Informasi, yang senantiasa membantu penulis dalam mengurus segala berkas administrasi.
6. Ketua Prodi Teknologi Informasi Ibu Ima Dwitawati, M.B.A dan Sekretaris Prodi Teknologi Informasi Bapak Khairan AR., M.Kom., serta seluruh staf prodi yang ikut membantu proses penelitian ini hingga selesai.

7. Sahabat dan teman-teman seangkatan 2019 dan 2018 yang selalu memberikan semangat dan dukungan untuk menyelesaikan tugas akhir ini bersama-sama. Terkhusus untuk kalian Alissa, Cut, Farla, Izzia, Nabila, Maitsa dan Saed yang selalu jadi teman untuk meneyelesaikan tugas akhir ini.
8. Seluruh pihak yang ikut membantu dalam menyelesaikan tugas akhir ini yang tidak dapat saya sebutkan namanya satu persatu.

Penulis mengucapkan Terimakasih dan semoga jasa kita semua dicatat sebagai amal jariyah oleh Allah swt. Penulis menyadari bahwa tugas akhir ini masih jauh dari kata sempurna, untuk itu dengan segala kerendahan hati penulis menerima saran dan masukin yang bermanfaat untuk menyempurnakan tugas akhir ini. Semoga dengan selesainya tugas akhir ini dapat memberikan manfaat bagi penulis dan pembaca.

Aamiin yaa rabbal 'alamiin.

Banda Aceh, 19 Oktober 2023

Penulis

Najla' Haura Andiko

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR RUMUS

BAB I
PENDAHULUAN

1.1 Latar Belakang

Berbagai jenis informasi pada dunia digital dapat di-*transfer* dengan mudah dan cepat melalui *interconnected networking* (internet). Dalam penggunaan internet, ada hal yang harus diperhatikan seperti masalah keamanan informasi. Permasalahan pencurian informasi pada dunia digital sangat meresahkan banyak *user*, karena dinilai merugikan dari berbagai aspek. Informasi yang bersifat sensitif atau rahasia, harus diamankan sehingga pihak lain tidak mengetahuinya. Berdasarkan permasalahan tersebut, masalah ini dapat di atasi salah satunya dengan menyisipkan informasi pada media lain. Namun, penelitian ini hanya fokus terhadap *secret message* berupa teks dan media yang digunakan adalah citra digital. Teknik ini dikenal dengan istilah *embedding.*

Teknik *embedding* pada penelitian ini, hanya menyisipkan *secret message* ke dalam *cover image* dengan tujuan agar *secret message* tidak terlihat secara kasat mata oleh manusia. Pada dasarnya, algoritma komputer dapat mendeteksi *secret message* yang disisipkan dalam sebuah *cover image* tergantung pada tingkat kompleksitas dari teknik penyisipan yang digunakan, juga kemampuan dari algoritma komputer (steganalisis) yang digunakan.

Pada permasalahan ini terdapat beberapa teknik, seperti *Discrete Wavelet Transform* (DWT), *Discrete Cosine Transform* (DCT), *Discrete Fourier Transform* (DFT), *Spread Spectrum* (SS), *Least Significant Bit* (LSB) atau kebalikan dari LSB yaitu MSB yang merupakan singkatan dari *Most Significant Bit* (Agung, 2023). Namun solusi untuk permasalahan diatas, penulis menggabungkan metode DWT dan LSB untuk meningkatkan keamanan pada *encrypted message*.

Kombinasi kedua metode ini dinilai cukup efektif oleh beberapa peneliti, seperti penelitian yang dilakukan oleh Deddy Rudhistiar pada 2022 dengan judul

"Implementasi Pengamanan Citra Digital Berbasis Enkripsi 2D *Logistic Map* dan DNA *Encoding* Dengan Penyisipan LSB dan DWT". Tujuan dari penelitian ini adalah untuk mengimplementasikan teknik enkripsi *cover image* dengan mengkombinasikan antara algoritma enkripsi 2D *logistic Map* dan DNA *Encoding*, serta metode LSB dan DWT. Pengujian dilakukan terhadap *cover image* dengan format BMP yang berukuran 512x512 piksel. Hasil pengujian menunjukkan bahwa metode ini dapat menghasilkan citra yang terenkripsi dengan tingkat keamanan tinggi. Adapun teknik DWT-LSB berhasil menyembunyikan informasi rahasia pada *cover image* tanpa merusak kualitas citra tersebut (Rudhistiar, 2022).

Penerapan teknik DWT-LSB dimulai dengan mengubah *cover image* ke dalam domain *wavelet* (DWT). Transformasi ini memproses citra berdasarkan nilai frekuensi dari *sub-band* citra tersebut. Komponen *sub-band* didapatkan melalui proses penurunan dari level dekomposisi suatu citra. *Sub-band* dengan frekuensi paling rendah dan berisi informasi penting dipilih sebagai wadah penyisipan *encrypted message*. *Band* dengan frekuensi rendah ini dikenal dengan LL atau *lower low*. Implementasi DWT melibatkan *low pass filter* dan *high pass filter* untuk memisahkan sinyal citra menjadi 2 domain, yaitu domain dengan frekuensi rendah dan frekuensi tinggi. Hasil dekomposisi domain tersebut, menghasilkan *sub-band* yang lebih kecil, yaitu *lower low* (LL), *lower high* (LH), *higher low* (HL) dan *higher high* (HH) (Rudhistiar, 2022).

Pada penelitian ini *sub-band* yang dipilih adalah LL dengan alasan memiliki kualitas gambar terbaik diantara *sub-band* lainnya. Sehingga apabila nilai LSB pada *cover image* dimanipulasi, tidak terlalu berdampak pada kualitas citra secara keseluruhan.

Adapun peran LSB adalah untuk menyembunyikan *encrypted message* ke dalam *cover image*. Nilai piksel dalam *band* ini kemudian dikonversi menjadi bilangan biner 8 digit. Metode LSB mengganti bit terakhir (bit ke-8) dengan *secret message* yang juga telah dikonversi ke biner. Pada LSB dilakukan modifikasi dengan merubah urutan penyisipan yang seharusnya pada bit ke-8 dengan bit ke-7. Hal ini memberikan

variasi pada metode LSB dan meningkatkan keamanan terhadap *encrypted message* itu sendiri.

Encrypted message adalah pesan rahasia yang telah dienkripsi menggunakan kriptografi *vigènere cipher*. Peran kriptografi disini adalah sebagai pengaman pada *secret message*, sehingga apabila pihak ketiga atau steganalisis berhasil mengetahui keberadaan pesan rahasia tersebut di dalam citra, mereka tetap kesulitan untuk mendekripsi pesan tersebut, karena tidak mengetahui kuncinya.

Kombinasi teknik steganografi dan kriptografi dapat meningkatkan nilai efektivitas dalam memproteksi *secret message* pada *cover image*. Metode steganografi DWT untuk menentukan posisi pesan yang akan disembunyikan, sedangkan LSB untuk menyisipkan pesan ke dalam citra tanpa merubah kondisi citra. Sementara kriptografi untuk mengamankan pesan rahasia dari pihak yang tidak berwewenang. Pesan akan dienkripsi menggunakan kunci yang hanya diketahui oleh pengirim dan penerima pesan.

Cover image yang telah dimodifikasi untuk menyembunyikan *secret message* disebut dengan *stego image*. Tujuan dari *stego image* adalah untuk membuat informasi tersembunyi (*secret message*) sulit terdeteksi oleh orang yang tidak memiliki wewenang.

1.2 Rumusan Masalah

Berdasarkan penjelasan dari latar belakang, berikut rumusan permasalahan yang akan diselesaikan pada penelitian ini:

1. Bagaimana kualitas *stego image* dan *encrypted message*?
2. Berapa besar kapasitas *secret message* yang dapat disembunyikan pada *cover image*?
3. Apa *encrypt message* atau *ciphertext* dapat kembali ke bentuk *plaintext*?

1.3 Tujuan Penelitian

Penelitian ini dilakukan dengan tujuan sebagai berikut:

1. Mengevaluasi kualitas *stego-image* dan *secret message*, mengukur keefektifan enkripsi terhadap *secret message* untuk memastikan keamanan informasi yang disampaikan dan membandingkan *cover image* dan *stego-image*.
2. Mengindentifikasi batas maksimal kapasitas *secret message* yang dapat disembunyikan pada *cover image*.
3. Menilai kemampuan sistem untuk mengembalikan *encrypted message* ke bentuk *plaintext*.

1.4 Batasan Penelitian

Untuk menyesuaikan pembahasan antara judul dan latar belakang, maka penulis membatasi penelitian sebagai berikut:

1. *Secret message* dienkripsi menggunakan algoritma *vigenere cipher*.
2. Citra ditransformasi menggunakan teknik DWT-LSB.
3. Citra yang digunakan adalah citra RGB dengan format TIFF.
4. Program dibuat menggunakan bahasa python di Google Collaboration.

1.5 Manfaat Penelitian

Berdasarkan uraian dari latar belakang, rumusan masalah dan tujuan penelitian, maka manfaat dari penelitian ini adalah:

1. Meningkatkan keamanan informasi.
2. Sebagai pengujian keamanan sistem dan infrastruktur komputer.
3. Membantu individu menjaga privasi mereka dengan cara menyembunyikan informasi pribadi dalam media yang tidak mencurigakan.
4. Mendorong pengembangan algoritma-algoritma baru untuk menyembunyikan data dan mendeteksi pesan tersembunyi.
5. Menjadi dasar pengembangan teknik steganografi yang lebih baik untuk kedepannya dengan menggabungkan berbagai teknik lainnya.

BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Relevan

Penelitian relevan merupakan serangkaian penelitian yang telah dilakukan oleh orang lain kemudian dijadikan sebagai sumber atau bahan dalam membuat penelitian. Tujuan dari penelitian relevan adalah untuk mengetahui bagaimana metode penelitian dan hasil penelitian terdahulu yang dilakukan, sekaligus sebagai tolak ukur bagi penulis dalam menganalisis suatu penelitian.

Pertama, penelitian yang dilakukan oleh Chaerul Umam, Muslih & Daffa Fadhillah yang berjudul "Kombinasi Steganografi LSB dan Kriptografi AES dalam Sekuriti Teks Rahasia pada Citra Berwarna". Penelitian ini dilakukan dengan tujuan untuk mengamankan pesan rahasia dengan mengkombinasikan antara steganografi dan kriptografi. Metode steganografi LSB untuk menyisipkan *encrypted message* pada *cover image*, sedangkan kriptografi AES sebagai enkripsi dan dekripsi *secret message*. Penelitian ini dilakukan dengan menguji apakah pesan terenkripsi dapat didekripsi dengan baik sehingga *secret message* sampai kepada pihak penerima. Pengujian yang dilakukan adalah menggunakan 5 buah citra berbeda ukuran dengan pesan yang sama. Citra dengan ukuran paling kecil yaitu 128x128 piksel memiliki waktu tempuh enkripsi dan dekripsi rata-rata 0,008 detik, sedangkan citra dengan ukuran paling besar yaitu 512x512 piksel memiliki waktu tempuh rata-rata 0,039 detik. Hasil penelitian juga membuktikan bahwa terjadi perubahan ukuran pada *cover image* setelah dilakukan penyisipan pada *encrypted message*. (Chaerul Umam dkk., 2022).

Kedua, penelitian yang dilakukan oleh Ade Ibrahim dengan judul "Menggabungkan Teknik Steganografi *Discrete Wavelet Transform* Dua Dimensi (2-D) dan Algoritma Kriptografi RSA pada Perancangan dan Analisis Keamanan Pesan". Tujuan dari penelitian ini adalah perancangan dan analisis keamanan *secret message* dengan menggabungkan DWT dan RSA. Steganografi DWT untuk menyembunyikan

encrypted message kedalam *sub-band layer* yang telah di transformasi, sedangkan kriptografi RSA untuk mengenkripsi dan dekripsi *secret message* menggunakan *public key* dan *private key*. Untuk mengukur kualitas citra dan tingkat keamanan dari *secret message*, penulis melakukan pengujian MSE, PSNR dan BER. Hasil pengujian menunjukkan kombinasi DWT dan RSA dapat memberikan keamanan yang cukup baik dan menghasilkan *stego-image* dengan kualitas baik. Hasil *avalanche effect* sebesar 7,33%, PSNR sebesar 62,3657%, BER dan CER sebesar 0% (Ibrahim, 2021).

Ketiga, penelitian yang dilakukan oleh Christy Atika Sari dan Wellia Shinta Sari yang berjudul "Kombinasi *Least Significant Bit* (LSB-1) dan Rivest Shamir Adleman (RSA) dalam Kriptografi Citra Warna". Tujuan penelitian ini adalah untuk meningkatkan keamanan pesan rahasia yang disisipkan pada citra menggunakan LSB-1 dan RSA. Metode LSB-1 untuk menyisipkan pesan, sedangkan RSA untuk enkripsi pesan. Dari hasil pengujian didapatkan citra dengan nama Papper.bmp berhasil memperoleh nilai tertinggi sebesar 7,5694% dan mendekati nilai standari entropi yaitu 8. Pada percobaan waktu proses enkripsi yang dilakukan memperolah grafik yang tidak stabil. Perubahan dengan durasi terlama yaitu pada percobaan ke-11 dengan durasi waktu 0,0013 detik, sedangkan durasi terpendek yaitu 0,002 detik. Hasil pengujian yang didapatkan adalah PSNR lebih dari 40 dB, yaitu 45,1778 dB dan nilai MSE sebesar 1,9739 (Sari & Sari, 2022).

Keempat, penelitian yang berjudul "Implementasi *Vigenere Cipher* Sebagai Pengaman pada Proses Deskripsi Steganografi *Least Significant Bit*" oleh Tuti Alawiyah, Rian Ardianto dan Dini Silvi Purnia. Tujuan dari penelitian ini adalah untuk meningkatkan keamanan *encrypted message* yang disembunyikan dalam *cover image*. Penelitian ini menggabungkan 2 metode yaitu LSB dan *Vigènere Cipher*. Penelitian ini berhasil menerapkan teknik *embedding secret message* dengan baik, adapun *secret message* atau dokumen yang disisipkan dalam citra juga dapat diperoleh kembali secara utuh tanpa ada gangguan. Adapun kondisi citra setelah disisipkan pesan tidak mengalami perubahan bentuk yang signifikan, hanya ukuran *file stego-image* yang berubah (Alawiyah dkk., 2020).

Berikut ringkasan terkait penelitian relevan beserta kekurangannya yang dibuat dalam Tabel II.1.

Tabel II.1 Penelitian Relevan

No	Nama Peneliti (Tahun). Judul Penelitian	Metode Penelitian	Hasil Penelitian	Kekurangan
1	Chaerul Umam, Muslih & Daffa Fadhillah (2022). Kombinasi Steganografi LSB dan Kriptografi AES dalam Sekuriti Teks Rahasia pada Citra Berwarna	– Steganografi LSB – Kriptografi AES.	– Pesan terlindungi dengan baik. – Pesan yang berhasil ekstrak oleh pihak ketiga, akan kesulitan untuk membaca pesan tanpa kunci.	– Penyisipan hanya dapat dilakukan pada *file* citra.
2	Ade Ibrahim (2021). Menggabungkan Teknik Steganografi *Discrete Wavelet Transform* Dua Dimensi (2-D) dan Algoritma Kriptografi RSA pada Perancangan dan Analisis Keamanan Pesan	– Steganografi DWT – Kriptografi RSA	– Program ini berhasil mengamankan pesan dengan baik. – Penulis melakukan pengujian dengan hasil *avalanche effect* sebesar 7.33%, PSNR 62,3657%, BER dan *CER* sebesar 0%.	– Program dirancang khusus untuk citra dengan format BMP. – Penulis hanya menggunakan metode DWT yang memiliki kekurangan dalam hal kapasitas penyimpanan.
3	Christy Atika Sari & Wellia Shinta Sari (2022). Kombinasi Least Significant Bit (LSB-1) dan *Rivest*	– Metode LSB-1 – Kriptografi RSA	– Kombinasi LSB-1 dan RSA berhasil diterapkan dengan baik. – Pesan terenkripsi dengan baik.	– Kapasitas penyimpanan pesan yang terbatas

No	Nama Peneliti (Tahun). Judul Penelitian	Metode Penelitian	Hasil Penelitian	Kekurangan
	Shamir Adleman (RSA) dalam Kriptografi Citra Warna		– Hasil enkripsi pesan menghasilkan nilai entropi terbaik yaitu sebesar 7,56. – Modifikasi LSB-1 menghasilkan *stego image* dengan kualitas baik yang tidak terlihat perubahan secara signifikan. – Hasil pengujian memperoleh nilai PSNR 45,1778 dB dan nilai MSE 1,9739.	– Rentan terhadap serangan dari steganalisis.
4	Tuti Alawiyah, Rian Ardianto & Dini Silvi Purnia (2020). Implementasi *Vigenere Cipher* Sebagai Pengaman pada Proses Deskripsi Steganografi *Least Significant Bit*	– Metode LSB – Kriptografi *Vigènere Cipher*	– Penyisipan pesan rahasia berhasil dilakukan dengan baik. – Pesan dapat kembali diekstrak dan memperoleh hasil yang sama dengan sebelum pesan disisipkan pada citra. – Citra yang telah disisipkan pesan tidak berubah secara kasat mata. – *Stego image* mengalami perubahan pada ukuran *file* gambar. – Kombinasi LSB dan *vigènere cipher* memiliki hasil citra yang baik.	– Penelitian ini menggunakan kriptografi *vigènere cipher* yang tingkat keamanannya masih terbatas dan dapat dipecahkan menggunakan teknik kriptanalisis yang canggih.

2.2 Kerangka Berpikir

Kerangka berpikir penulis dituangkan pada Gambar II.1.

Gambar II.1 Kerangka Berpikir

2.3 Landasan Teori

2.3.1 Citra Digital

Sebuah gambar merupakan representasi visual dari suatu objek, orang atau pemandangan. Adapun definisi dari citra digital adalah fungsi dua dimensi $f(x,y)$ yang merupakan proyeksi pemandangan 3 dimensi menjadi sebuah bidang dengan proyeksi 2 dimensi, dimana x,y menyatakan lokasi dari elemen piksel dan mengandung intensitas nilai. Sebuah citra dapat dikatakan citra digital adalah apabila nilai x,y dan intensitas bersifat diskrit (Yanu dkk., 2022)

Secara matematis, citra digital adalah representasi matriks dari gambar dua dimensi menggunakan elemen sel titik dalam jumlah yang terbatas, atau dikenal dengan piksel. Setiap piksel diwakili oleh nilai numerik yang berbeda, berikut penjelasannya:

1. Citra biner atau *monochrome*

Citra biner hanya memiliki kedalaman 1-bit memori untuk menyimpan dua warna, yaitu 0 untuk hitam dan 1 untuk putih.

2. Citra *grayscale*

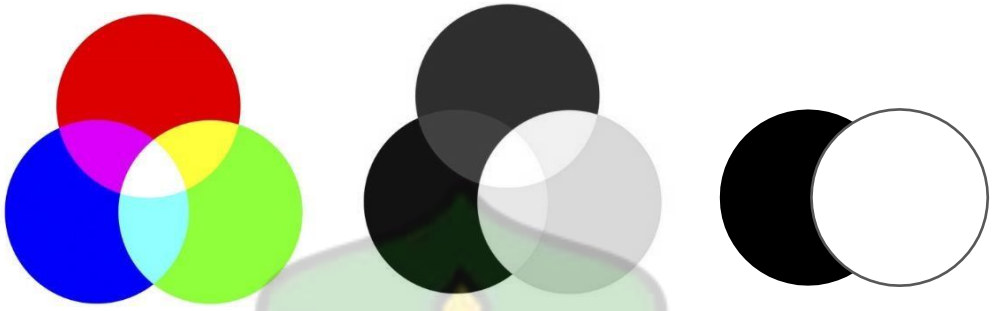
Citra ini hanya mempunyai warna tingkat keabuan yang disimpan dalam 8-bit dengan nilai intensitas pikselnya 0–255, yaitu 0 untuk hitam dan 255 untuk putih.

3. Citra Warna RGB

Citra warna terbentuk berdasarkan kombinasi warna primer yaitu *red*, *green* dan *blue* yang memiliki kedalaman 8 bit per kanalnya. Variasi warna pada setiap piksel adalah sebanyak 255 x 255 x 255 = 16.581.375 warna. Pada Tabel *2.2* dapat dilihat contoh warna dan nilai penyusun dari *red*, *green* dan *blue* (Marleny, 2021).

Berdasarkan penjelasan diatas, citra RGB menghasilkan paling banyak warna diantara citra lainnya. Citra biner hanya memiliki dua warna, yaitu hitam dan putih,

berbeda dengan citra *grayscale* yang memiliki warna dengan tingkat keabuan. Berikut penjelasannya pada Gambar II.2.



(a) Citra Warna RGB (b) Citra *Grayscale* (c) Citra Biner

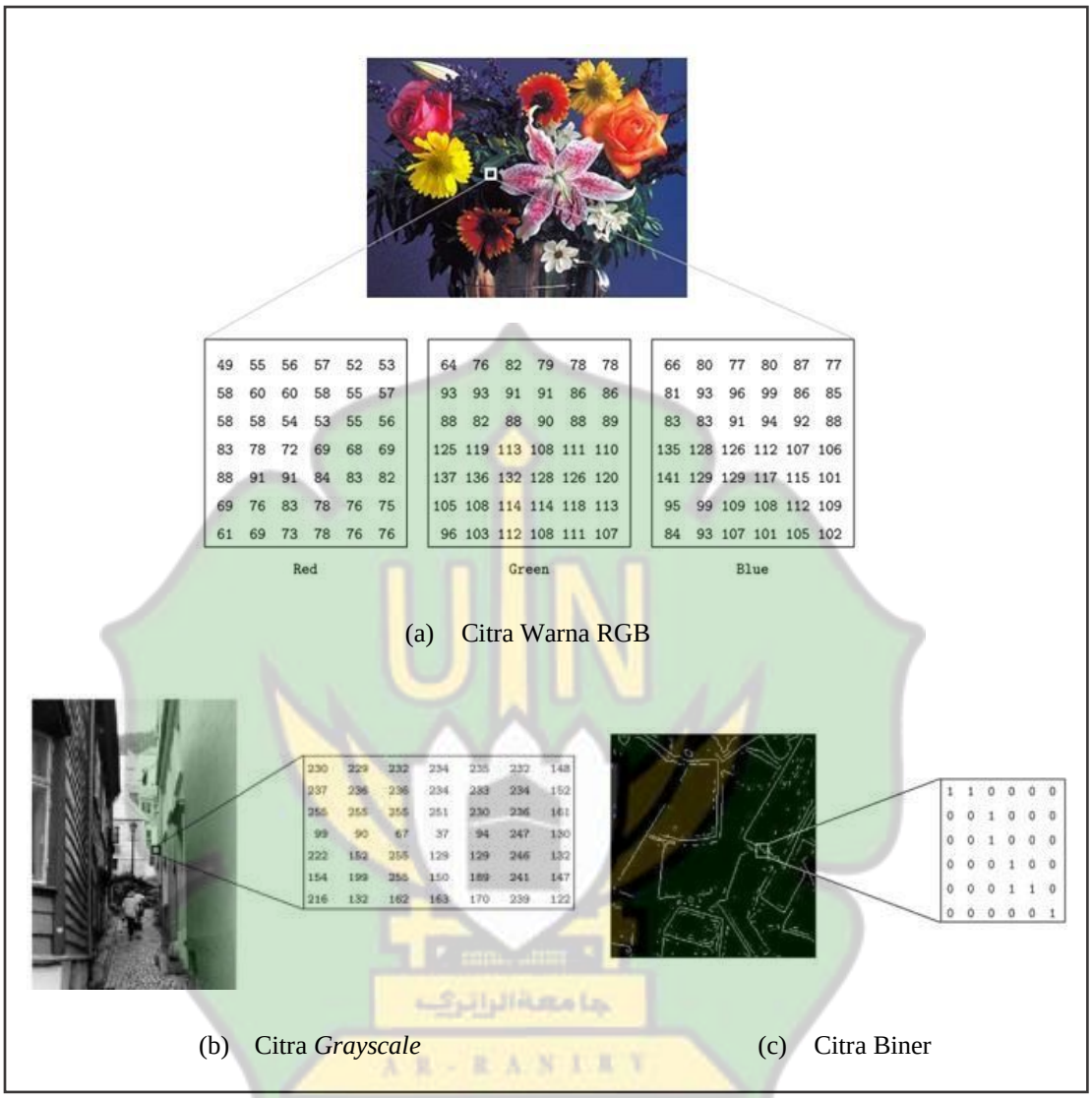
Gambar II.2 Warna Penyusun pada Citra RGB, *Grayscale* dan Biner

Tabel II.2 Nilai Penyusun Warna RGB dan *Grayscale*

Warna	R	G	B
Merah	255	0	0
Hijau	0	255	0
Biru	0	0	255
Hitam	0	0	0
Putih	255	255	255

Sumber : Pengolahan Citra Digital Menggunakan *Pyhton*

Contoh citra RGB, *grayscale* dan biner dapat dilihat pada Gambar II.2 yang merepresentasikan nilai dari setiap pikselnya. Ketiga citra dalam Gambar II.2 merupakan contoh representasi visual. Citra dapat berupa hasil dari fotografi, lukisan, gambar atau coretan pada layar monitor. Sebuah citra dapat dikatakan juga sebagai penyebaran dari variasi gelap-terang, redup-cerah dan atau variasi warna pada suatu bidang datar.



(a) Citra Warna RGB

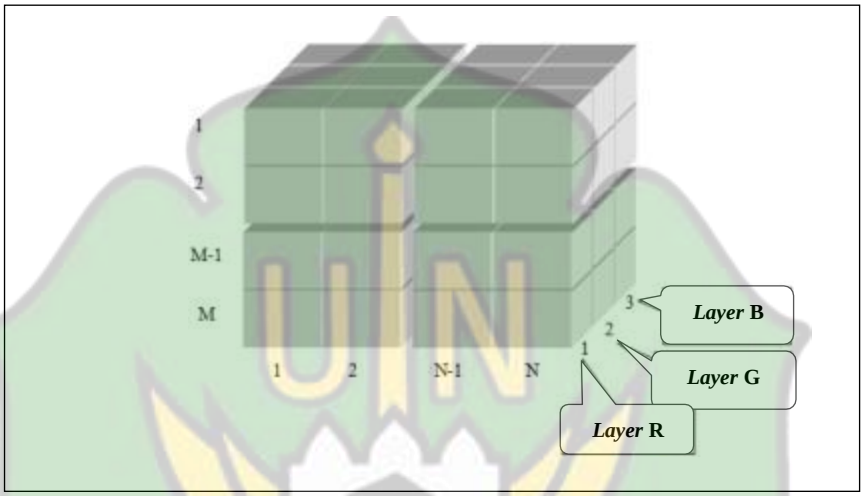
(b) Citra *Grayscale* (c) Citra Biner

Gambar II.3 Citra Berwarna dan Representasi Warna RGB
(Sumber: catatanpeneliti.wordpress.com)

Pada Gambar II.3, menunjukkan bahwa citra RGB terdiri dari 3 *layer*. Satu piksel citra berwarna terdiri dari 3 warna dasar, yaitu merah, hijau dan biru. Sehingga untuk dapat menghasilkan warna hijau, tidak selalu dengan komposisi warna R = 0, G = 255 dan B = 0. Namun, komposisi R = 55, G = 76 dan B = 80 juga akan menghasilkan

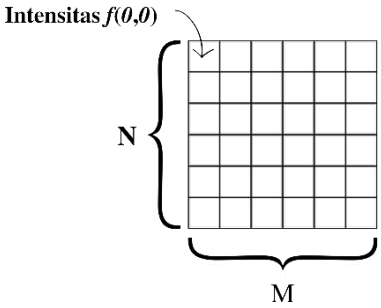
warna hijau yang lebih gelap. Citra *grayscale* hanya memiliki tingkatan warna keabuan dan citra biner yang memiliki 2 warna, hitam dan putih.

Citra berwarna merupakan larik berdimensi tiga yang dapat digambarkan seperti pada Gambar II.4 berikut. *Layer* pertama menyatakan komponen R, *layer* kedua menyatakan komponen G dan *layer* ketiga menyatakan komponen B.



Gambar II.4 *Layer* pada citra RGB
(Sumber: Menggabungkan Teknik Steganografi DWT 2-D dan Algoritma RSA pada Perancangan dan Analisis Keamanan Pesan)

Ketentuan yang ada pada sebuah citra yaitu variabel *x* untuk posisi horizontal dan variabel *y* untuk arah vertikal. Citra pada umumnya disajikan sebagai *f*(*x*,*y*), namun pada versi diskret citra di sistem komputer adalah *f*(*m*,*n*), dengan *m* dan *n* menjadi nomor urut posisi. Pada setiap posisi (*m*,*n*) terdapat detail yang disebut dengan *picture element* (piksel) atau unsur gambar. Gambaran atau representasi dari sebuah citra digital dapat diperhatikan pada Gambar II.5, terdapat perbedaan dari cara menyatakan posisi titik antara koordinat citra dengan koordinat grafik (kartesian).



Gambar II.5 Representasi Citra Digital

Sebuah citra digital yang mengalami proses menggunakan komputer, harus diterangkan secara numerik (angka) dan bernilai diskret. Proses citra digital terbagi menjadi dua yaitu *sampling* dan *quantization*. Proses *sampling* membagi citra asli ke dalam *grid* yang berbentuk bujur sangkar pada arah horizontal dan vertikal, sedangkan proses *quantization* membagi intensitas citra asli menjadi citra dengan intensitas yang ditentukan. Proses ini berdampak pada penurunan kualitas penampilan objek pada citra dibanding dengan penampilan objek aslinya.

Pada umumnya, citra digital berbentuk persegi panjang dengan dimensi lebar x tinggi (*M* x *N*). Dengan *M* menyatakan jumlah kolom (tinggi) dan *N* menyatakan jumlah baris (lebar) pada citra. Resolusi citra menyatakan jumlah baris atau kolom piksel persatuan pengukuran, misalnya dpi (*dot per inch*) yang menyatakan banyak piksel dalam skala 1 *inch* (F. Marpaung dkk., 2022).

2.3.2 *Discrete Wavelet Transform* (DWT)

Kata *wavelet* dicetuskan oleh Alex Grossmann dan Jean Morlet pada awal tahun 1980-an. *Wavelet* berasal dari bahasa Perancis, *ondelette* bermakna gelombang kecil dan *onde* berarti gelombang. Saat diterjemahkan ke dalam bahasa inggris menjadi *wave*. Kedua kata ini digabungkan sehingga terbentuk kata baru yaitu *wavelet*. Konsep *wavelet* adalah alat analisis yang biasa digunakan untuk menyajikan data atau fungsi

operator kedalam komponen frekuensi yang berlainan. Komponen tersebut kemudian dikaji dengan sebuah resolusi yang sesuai dengan skalanya (Ibrahim, 2021).

Transformasi *wavelet* digunakan untuk dekomposisi sinyal ke *wavelet* komponen. Konsep ini dapat memisahkan rincian halus dalam sinyal. *Wavelet* paling kecil digunakan untuk mengisolasi rincian yang baik disinyal, sedangkan *wavelet* besar dapat mengidentifikasi rincian yang kasar. Jenis transformasi DWT terdiri dari 1 dimensi dan 2 dimensi.

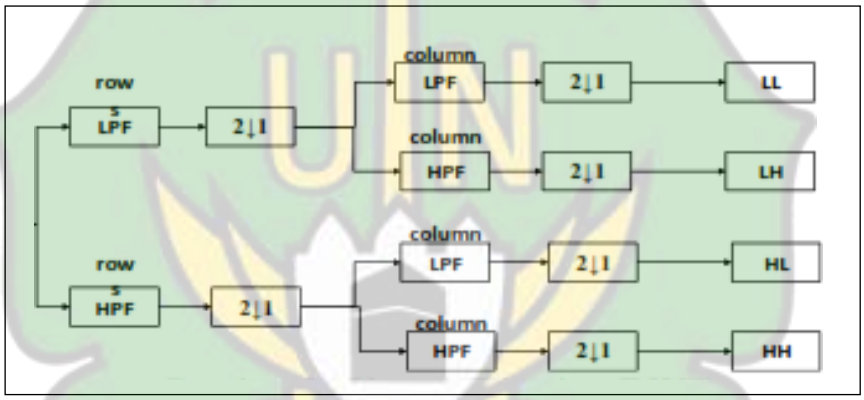
Konsep DWT tergolong sangat sederhana. Citra asli yang ditransformasi kemudian didekomposisi menjadi 4 sub-*image* baru untuk menggantikannya. Setiap sub berukuran seperempat kali dari citra asli. Sub-*image* pada posisi atas kanan, bawah kanan dan kiri akan terlihat seperti versi kasar dari citra asli karena berisi komponen frekuensi tinggi dari citra asli. Pada sub-*image* atas kiri, terlihat seperti citra asli dan lebih halus karena berisi komponen dengan frekuensi lebih rendah dari citra asli. Sub-*image* tersebut kemudian dibagi menjadi 4 sub baru, proses ini dapat diulangi seterusnya sesuai dengan tingkatan transformasi yang diinginkan (Ibrahim, 2021).

Contoh proses transformasi citra menggunakan *wavelet* dapat dilihat pada Gambar II.6.

Gambar II.6 Tahapan DWT pada citra digital
(Sumber: *Remote Sensing Digital Image Analysis*)

Citra pada Gambar II.6 telah direkonstruksi sepenuhnya dengan teknik DWT menggunakan bank filter rekonstruksi. Terlihat jelas bahwa setiap tingkatan dekomposisi mengandung semua informasi dari citra asli. Menggunakan teknik ini citra dapat dikompres tanpa kehilangan informasi yang signifikan pada citra. Hal ini memungkinkan data gambar disimpan atau dikirim menggunakan bit data digital yang jauh lebih sedikit daripada aslinya (Richards, 2022).

Pada proses dekomposisi transformasi *wavelet* diskret 2 dimensi dilakukan dengan memproses baris dan kolom secara terpisah. Berikut ilustrasi skema *filtering* DWT pada Gambar II.7.



Gambar II.7 Skema *Filtering* DWT

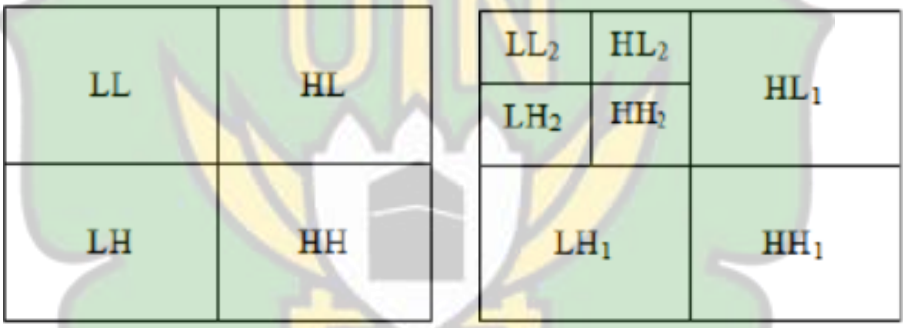
(Sumber: Implementasi Pengamanan Citra digital Berbassis Enkripsi 2D *Logistic Map* dan DNA *Encoding* dengan Penyisipan LSB dan DWT)

Implementasi Transformasi *Wavelet* Diskret memanfaatkan *low-pass filter* dan *high-pass filter* dengan melewatkan sinyal dan melakukan *down-sampling* pada keluaran setiap level *filter* pada tahap dekomposisi. Jika bernilai tinggi, *High-pass filter* dapat menentukan frekuensi tersebut, namun *low-pass filter* untuk menentukan frekuensi rendah (Rudhistiar, 2022).

Pada Gambar II.7, LL menyatakan bagian koefiesien yang didapatkan melalui proses *low-pass* dilanjutkan dengan *low-pass*. Citra bagian ini merupakan versi lebih halus dari citra aslinya sehingga disebut dengan komponen aproksimasi. LH

menyatakan bagian koefisien yang didapatkan melalui proses *low-pass* yang dilanjutkan dengan *high-pass*. HL diperoleh melalui proses *highpass* dan dilanjutkan dengan *low-pass*. Sedangkan HH diperoleh melalui *high-pass* kemudian dilanjutkan dengan *high-pass*. Komponen LH, HL dan HH merupakan komponen detail (Ibrahim, 2021).

Low-pass filter (LPF) digunakan untuk menghasilkan *sub-band* citra dengan resolusi rendah (LL), sementara *high-pass filter* (HPF) untuk menghasilkan *sub-band* dengan citra beresolusi tinggi, yaitu LH, HL dan HH. Pada proses ini, HPF bertindak sebagai operator yang mengekstraksi fitur pada citra asli, sementara LPF sebagai operator *smoothing* yang dapat menghilangkan detail kasar pada citra.



Gambar II.8 Dekomposisi Transformasi *Wavelet* Level 1 dan 2

(Sumber: Menggabungkan Teknik Steganografi DWT 2-D dan Algoritma RSA pada Perancangan dan Analisis Keamanan Pesan)

Pada Gambar II.8, citra asli didekomposisi menjadi 4 bagian dengan penjelasan sebagai berikut:

1. *Low-Low* (LL)

   Merepresentasikan informasi frekuensi rendah secara horizontal maupun vertikal pada citra asli. Bagian ini yang akan diambil sebagai citra asli yang baru untuk di dekomposisi kembali pada level yang lebih tinggi.

2. LH (*low-high*)

Menyatakan informasi frekuensi rendah secara horizontal dan frekuensi tinggi secara vertikal. Informasi horizontal pada bagian ini memiliki resolusi rendah, sedangkan informasi vertikalnya beresolusi tinggi.

3. HL (*high-low*)

Merepresentasikan informasi frekuensi rendah secara vertikal dan memiliki resolusi rendah, sedangkan frekuensi tinggi secara horizontal dan memiliki resolusi tinggi.

4. HH (*high-high*)

Merepresentasikan informasi frekuensi tinggi secara vertikal dan horizontal. Pada kedua arah tersebut, informasi pada *sub-band* ini memiliki resolusi yang tinggi.

2.3.3 Format Citra Digital

Standar format citra digital saat ini terdiri dari berbagai jenis dengan karakteristik yang berbeda. Berikut penjelasan beberapa format citra yang umum digunakan:

1. *Bitmap* (BMP)

BMP merupakan format penyimpanan tidak terkompres yang umumnya digunakan untuk menyimpan citra biner hingga citra berwarna.

2. *Tagged Image Format* (TIFF)

Format TIFF merupakan format gambar terbaik. Apabila format gambar ini diberikan suatu perlakuan, semua informasi di dalamnya tidak akan hilang. Format gambar ini pada umumnya digunakan dalam dunia percetakan dengan kualitas gambar yang sangat tinggi. Selain dapat menyimpan gambar dengan kualitas hingga 32-bit, TIFF juga dapat digunakan untuk melakukan transfer antar program.

3. *Portable Network Graphics* (PNG)

Format ini dikenal dengan format citra terkompresi yang dapat digunakan pada citra *grayscale* dan citra berwarna. PNG mampu menyimpan informasi sampai kanal *alpha* dengan kapasitas sebesar 1 hingga 16 bit per kanal.

4. *Joint Photographic Experts Group* (JPEG)

Ini merupakan format yang digunakan untuk melakukan transmisi data. Format ini biasanya digunakan untuk menyimpan gambar digital hasil kompresi dengan metode JPEG.

5. *Graphics Interchange Format* (GIF)

Format GIF digunakan pada citra berwarna dengan palet 8-bit pada aplikasi *web*. Kekurangan format ini terletak pada kualitasnya yang rendah.

Citra digital seringkali dikompresi untuk mengecilkan ukurannya atau untuk mengubah berbagai atribut, seperti jenis berkas, ukuran, resolusi dan kedalaman bit. Kompresi data terbagi menjadi 2 yaitu:

1. Kompresi *Lossless* (tanpa kehilangan data)

Teknik ini digunakan untuk mengurangi ukuran data tanpa mengurangi kualitas data. Memulihkan dan membangun kembali data dalam bentuk aslinya setelah *file* didekompresi. Saat ukuran *file* gambar dikompresi, kualitasnya tetap sama. Teknik *lossless* digunakan antara lain *Run-Length Encoding* (RLE), *Huffman Encoding* dan *Arithmetic Encoding*. Menggunakan metode ini, meskipun ukuran *file* diperkecil, pengurangannya lebih sedikit dibandingkan dengan pengurangan menggunakan kompresi *lossy*.

2. Kompresi *Lossy* (dengan kehilangan data)

Teknik ini mengurangi ukuran data dengan membuang informasi yang dianggap kurang penting, sehingga kualitas data yang dihasilkan lebih rendah dibandingkan dengan data asli. Data dalam *file* dihapus dan tidak dikembalikan

ke bentuk aslinya setelah dekompresi. *Lossy* menghapus data secara permanen dan mengurangi ukuran *file* jauh lebih banyak daripada ukuran *file* yang dicapai setelah kompresi *lossless*. Semakin banyak *file* yang dikompresi, semakin banyak degradasi yang terjadi. Teknik ini umumnya digunakan antara lain pada *Transform Coding* seperti DCT, DWT dan teknik pemrosesan sinyal seperti *Perceptual Coding* dan *Vector Quantization*.

2.3.4 Kriptografi

Kriptografi merupakan salah satu upaya untuk menjaga integritas data dengan mengacak data menjadi kode-kode yang tidak dapat dibaca. Kata kriptografi sendiri berasal dari Bahasa Yunani yaitu *crypto* yang berarti rahasia dan *graphia* yang berarti tulisan. Secara umum, kriptografi adalah ilmu yang mempelajari tentang teknik penyandian dengan cara mengubah pesan asli (*plaintext*) menjadi tulisan yang sulit dibaca (*ciphertext*). Berikut merupakan penjelasan dari komponen utama dalam kriptografi:

1. *Plaintext*, merupakan pesan yang dikirim atau simpan dalam bentuk asli dan dapat dibaca secara langsung.
2. *Ciphertext*, merupakan pesan yang telah di enkripsi sehingga tidak dapat di baca dan tidak memiliki makna.
3. Enkripsi, yaitu proses mengubah *plaintext* menjadi *ciphertext*.
4. Dekripsi, merupakan proses mengembalikan pesan *ciphertext* kembali ke dalam bentuk *plaintext*.
5. Kunci, adalah parameter untuk mengenkripsi dan mendekripsi. Kunci dapat berupa abjad, bilangan atau karakter.

Dalam dalam kriptografi ada beberapa unsur keamanan penting yang menjadi pelayanan dari kriptografi, unsur-unsur tersebut yaitu:

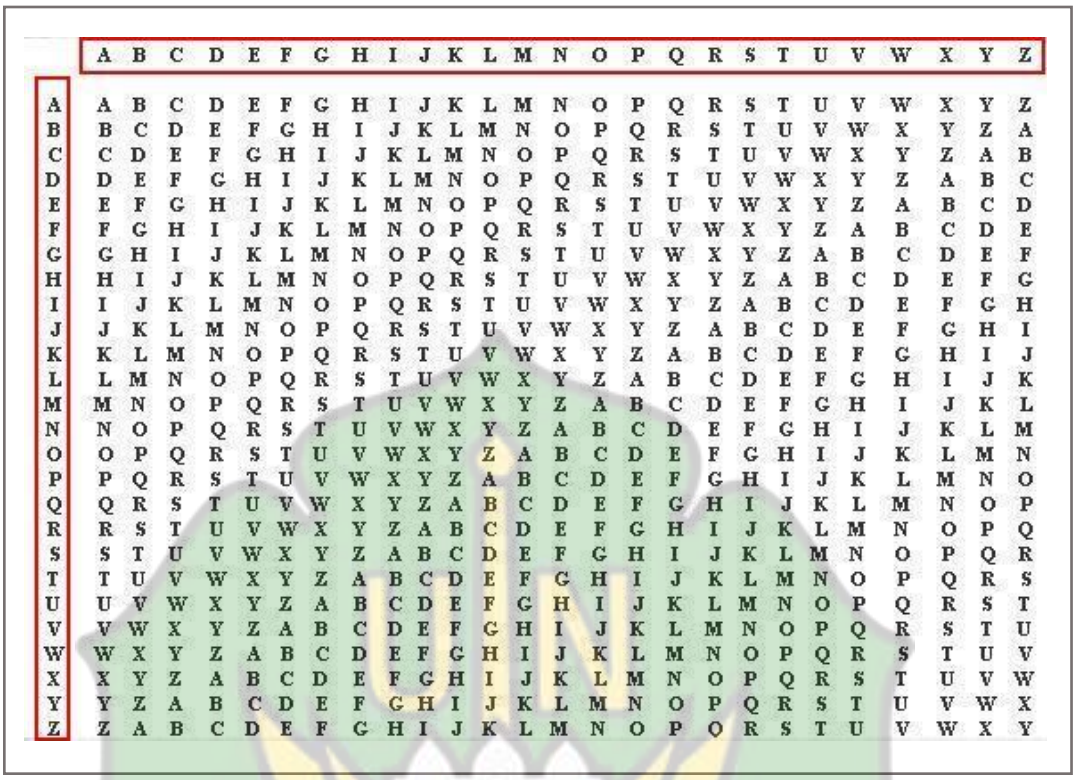
1. *Data Confidentiality*, adalah menjaga kerahasiaan pesan dari pihak yang tidak memiliki akses terhadap pesan tersebut.

2. *User Authentication*, dapat diartikan bahwa apakah pesan tersebut benar dari pihak yang memiliki izin atau dari pihak yang tidak bertanggungjawab.
3. *Message Authentication*, adalah tentang bagaimana menjaga keaslian pesan yang di kirim atau yang diterima.
4. *Nonrepudiation*, mengupayakan agar pesan yang diubah tidak dapat di sangkal oleh pengirimnya.

2.3.5 Vigènere Cipher

Algoritma ini ditemukan pertama kali oleh seorang kriptologis dari Perancis yang bernama Blaise de Vigenere pada 1586. Algoritma ini dapat dilakukan dengan menggunakan tabel yang berisi abjad dari *a-z*. Tabel ini disebut dengan *tabula recta* atau bujur sangkar *vigènere*. *Tabula recta* merupakan tabel persegi huruf yang terdiri dari a-z. Barisan tabel ini dibuat dengan menggeserkan huruf sebelumnya ke arah kiri. Seperti pada Gambar II.9 di bawah ini. Istilah ini ungkapkan pertama kali oleh seorang penulis dan biarawan Jerman yaitu Johannes Trithemius pada tahun 1508.

Berdasarkan aturannya, kolom paling kiri menyatakan huruf kunci dan baris paling atas menyatakan huruf *plaintext*. Hasil enkripsi *plaintext* dapat diperoleh dengan menyandingkan huruf *plaintext* dengan huruf kunci. Jika huruf kunci lebih sedikit, maka huruf kunci akan di ulang hingga sama banyaknya dengan *plaintext*. Sistem pengulangan ini disebut sistem periodik (Ruing, 2020).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Gambar II.9 *Tabula Recta*

(Sumber: wordpress.com)

Contoh enkripsi pesan menggunakan *tabula recta* dapat dilihat pada Tabel II.3. Huruf kunci *h* jika ditarik secara horizontal ke kanan dan huruf *plaintext i* ditarik secara vertikal ke bawah, maka perpotongannya akan bertemu pada huruf *p*. Dapat disimpulkan *ciphertext* dari *plaintext i* dan kunci *h* adalah *p*.

Tabel II.3 Hasil Enkripsi Pesan Dengan *Tabula Recta*

Komponen	***Secret Message***											
Plaintext	I	N	I	P	L	A	I	N	T	E	X	T
Kunci	H	A	I	H	A	I	H	A	I	H	A	I
Ciphertext	P	N	Q	W	L	I	P	N	B	L	X	B

Metode lain untuk mengenkripsi pesan yaitu menggunakan subtitusi angka yang menukarkan huruf dengan angka. Gambar II.10 merupakan tabel substitusi algoritma kriptografi *vigènere cipher*.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Gambar II.10 Substitusi Algoritma *Vigènere Cipher*

Setelah mengkonversi huruf *plaintext* dan kunci ke bilangan, selanjutnya adalah mengenkripsi menggunakan persamaan 1. Adapun proses dekripsi dapat dilihat pada persamaan 2. Keterangan C_i menyatakan karakter *cipher* ke i, P_i menyatakan karakter *plain* ke i dan K_i menyatakan kunci ke i.

$$C_i = (P_i + K_i) \, mod \, 26 \quad \text{..... (1)}$$

$$P_i = (C_i - K_i) \, mod \, 26 \quad \text{..... (2)}$$

Berikut merupakan contoh enkripsi pesan dengan *vigènere cipher* menggunakan table bujur sangkar dan rumus:

C_1 = (8 + 7) mod 26 = 15
C_2 = (13 + 0) mod 26 = 13
C_3 = (8 + 8) mod 26 = 16
C_4 = (15 + 7) mod 26 = 22
C_5 = (11 + 0) mod 26 = 11
C_6 = (0 + 8) mod 26 = 8
C_7 = (8 + 7) mod 26 = 15
C_8 = (13 + 0) mod 26 = 13
C_9 = (19 + 8) mod 26 = 1
C_{10} = (4 + 7) mod 26 = 11
C_{11} = (23 + 0) mod 26 = 23
C_{12} = (19 + 8) mod 26 = 1

Hasil enkripsi berdasarkan pesan dan kunci di atas adalah seperti pada Tabel II.4 berikut.

Tabel II.4 Hasil Enkripsi Pesan dengan Substitusi Angka

Komponen	***Secret Message***											
Plaintext	8	13	8	15	11	0	8	13	19	4	23	19
Kunci	7	0	8	7	0	8	7	0	8	7	0	8
Modulo 26	15	13	16	22	11	8	15	13	1	11	23	1
Ciphertext	P	N	Q	W	L	I	P	N	B	L	X	B

Metode substitusi angka di atas adalah dengan menukarkan huruf-huruf pesan dan kunci menjadi angka, kemudian dilakukan perhitungan antara $P_1 + K_1$ mod 26, $P_2 + K_2$ mod 26 dan begitu seterusnya, sehingga di dapatkan hasil sisa bagi dari keseluruhan pesan dan kunci. Kemudian diterjemahkan kedalam bentuk huruf, maka hasil *ciphertext* yang didapat adalah seperti pada Tabel II.4.

2.3.6 *American Standard Code for Information Interchange* (ASCII)

Sistem ASCII merupakan standar pengkodean dalam bertukar informasi. Pertukaran informasi dalam kode ASCII adalah salah satu standar internasional dalam bentuk kode huruf dan symbol yang bersifat universal. Sistem ini memiliki komposisi bilangan biner 8-bit mulai dari 00000000 hingga 11111111. Jika ditotalkan, kombinasi yang dihasilkan sebanyak 256 dimulai dari 0 hingga 255 (Marpaung dkk., 2023).

Pada penelitian ini, *ciphertext* yang telah dienkripsi dengan algortima *vigènere cipher* terdiri dalam bentuk abjad. Adapun nilai dari sebuah citra jika direpresentasikan terdiri dalam bentuk bilangan angka. Sedangkan proses *embedding* yang dilakukan adalah dengan metode LSB+1 yang terdiri dari bilangan biner. Oleh sebab itu, diperlukan kode ASCII untuk mengkonversikannya ke bilangan biner dengan tujuan

proses penyisipan dapat dilakukan. Kode ASCII yang dimaksud adalah seperti pada Gambar II.11, namun pada gambar ini hanya menampilkan dari desimal 0-127 saja.

Decimal - Binary - Octal - Hex – ASCII
Conversion Chart

Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII	Decimal	Binary	Octal	Hex	ASCII
0	00000000	000	00	NUL	32	00100000	040	20	SP	64	01000000	100	40	@	96	01100000	140	60	`
1	00000001	001	01	SOH	33	00100001	041	21	!	65	01000001	101	41	A	97	01100001	141	61	a
2	00000010	002	02	STX	34	00100010	042	22	"	66	01000010	102	42	B	98	01100010	142	62	b
3	00000011	003	03	ETX	35	00100011	043	23	#	67	01000011	103	43	C	99	01100011	143	63	c
4	00000100	004	04	EOT	36	00100100	044	24	$	68	01000100	104	44	D	100	01100100	144	64	d
5	00000101	005	05	ENQ	37	00100101	045	25	%	69	01000101	105	45	E	101	01100101	145	65	e
6	00000110	006	06	ACK	38	00100110	046	26	&	70	01000110	106	46	F	102	01100110	146	66	f
7	00000111	007	07	BEL	39	00100111	047	27	'	71	01000111	107	47	G	103	01100111	147	67	g
8	00001000	010	08	BS	40	00101000	050	28	(	72	01001000	110	48	H	104	01101000	150	68	h
9	00001001	011	09	HT	41	00101001	051	29	)	73	01001001	111	49	I	105	01101001	151	69	i
10	00001010	012	0A	LF	42	00101010	052	2A	*	74	01001010	112	4A	J	106	01101010	152	6A	j
11	00001011	013	0B	VT	43	00101011	053	2B	+	75	01001011	113	4B	K	107	01101011	153	6B	k
12	00001100	014	0C	FF	44	00101100	054	2C	,	76	01001100	114	4C	L	108	01101100	154	6C	l
13	00001101	015	0D	CR	45	00101101	055	2D	-	77	01001101	115	4D	M	109	01101101	155	6D	m
14	00001110	016	0E	SO	46	00101110	056	2E	.	78	01001110	116	4E	N	110	01101110	156	6E	n
15	00001111	017	0F	SI	47	00101111	057	2F	/	79	01001111	117	4F	O	111	01101111	157	6F	o
16	00010000	020	10	DLE	48	00110000	060	30	0	80	01010000	120	50	P	112	01110000	160	70	p
17	00010001	021	11	DC1	49	00110001	061	31	1	81	01010001	121	51	Q	113	01110001	161	71	q
18	00010010	022	12	DC2	50	00110010	062	32	2	82	01010010	122	52	R	114	01110010	162	72	r
19	00010011	023	13	DC3	51	00110011	063	33	3	83	01010011	123	53	S	115	01110011	163	73	s
20	00010100	024	14	DC4	52	00110100	064	34	4	84	01010100	124	54	T	116	01110100	164	74	t
21	00010101	025	15	NAK	53	00110101	065	35	5	85	01010101	125	55	U	117	01110101	165	75	u
22	00010110	026	16	SYN	54	00110110	066	36	6	86	01010110	126	56	V	118	01110110	166	76	v
23	00010111	027	17	ETB	55	00110111	067	37	7	87	01010111	127	57	W	119	01110111	167	77	w
24	00011000	030	18	CAN	56	00111000	070	38	8	88	01011000	130	58	X	120	01111000	170	78	x
25	00011001	031	19	EM	57	00111001	071	39	9	89	01011001	131	59	Y	121	01111001	171	79	y
26	00011010	032	1A	SUB	58	00111010	072	3A	:	90	01011010	132	5A	Z	122	01111010	172	7A	z
27	00011011	033	1B	ESC	59	00111011	073	3B	;	91	01011011	133	5B	[	123	01111011	173	7B	{
28	00011100	034	1C	FS	60	00111100	074	3C	<	92	01011100	134	5C	\	124	01111100	174	7C	|
29	00011101	035	1D	GS	61	00111101	075	3D	=	93	01011101	135	5D	]	125	01111101	175	7D	}
30	00011110	036	1E	RS	62	00111110	076	3E	>	94	01011110	136	5E	^	126	01111110	176	7E	~
31	00011111	037	1F	US	63	00111111	077	3F	?	95	01011111	137	5F	_	127	01111111	177	7F	DEL

Gambar II.11 Kode ASCII
(Sumber: bournetocode.com)

2.3.7 Steganografi

Dalam bahasa Yunani, kata steganografi berasal dari *steganos* yang berarti tersembunyi dan *graphien* yang berarti menulis atau menggambar. Steganografi dapat diartikan sebagai seni komunikasi rahasia dengan mengirim pesan yang telah disembunyikan pada suatu objek yang tidak mencurigakan. Gregory Kipper pada bukunya yang berjudul "*Investigator's Guide to Steganography*" memberikan dekripsi yang popular untuk steganografi yaitu *Hidden in Plain Sight* yang artinya tersembunyi di depan mata. Tujuan utama dari steganografi adalah menyembunyikan pesan ke

dalam sebuah pesan dengan cara yang tidak memungkinkan untuk dideteksi oleh musuh bahwasannya terdapat pesan kedua.

Teknik dasar yang digunakan dalam steganografi terdiri dari 7 teknik, yaitu:

1. *Injection*, yaitu menyisipkan pesan rahasia secara langsung ke dalam suatu media. Teknik ini sering disebut penyisipan atau *embedding*.
2. Substitusi, data normal yang diganti data rahasia. Teknik ini menurunkan kualitas media yang disisipi pesan.
3. *Transform Domain*, dinilai sangat efektif karena menyembunyikan pesan pada *transform space*.
4. *Spread Spectrum*, transmisi dengan *pseudo-noise code* berbentuk gelombang yang menyebarluaskan energi sinyal dalam sebuah jalur komunikasi yang lebih besar daripada sinyal jalur komunikasi informasi.
5. *Statistical Method*, disebut juga dengan teknik skema *steganography* 1 bit. Perubahan statistik ditunjukkan dengan indikasi 1 dan 0 untuk hasil statistic yang sama.
6. *Distortion*, menciptakan perubahan atas benda yang ditumpangi oleh data rahasia.
7. *Cover Generation*, memilih *cover object* sebagai media untuk menyembunyikan pesan.

Agar dapat dikatakan *stego image* yang baik, terdapat 4 kriteria yang harus diperhatikan dalam steganografi, yaitu:

1. *Imperceptibility*, antara *cover image* dan *stego image* harus tidak dapat dibedakan oleh indra manusia dengan tujuan agar pesan tidak terdeteksi.
2. *Fidelity*, kualitas citra sebelum dan setelah disisipi pesan tidak jauh berubah. Sehingga tidak menimbulkan kecurigaan pihak lain.
3. *Recovery*, pesan rahasia yang disembunyikan pada *cover image* harus dapat diekstrak kembali agar dapat digunakan oleh penerima.

2.3.8 *Least Significant Bit* (LSB-1)

LSB merupakan metode yang menukar bit piksel pada medium penampung dengan setiap bit pesan yang akan disembunyikan. Metode LSB adalah mengganti barisan bit paling kanan, sehingga pesan yang disembunyikan akan mudah terungkap. Kelemahan metode ini dapat di atasi dengan memodifikasi LSB menjadi LSB-1. Modifikasi ini bekerja dengan mengganti barisan bit ke tujuh (Afsari, dkk., 2022). Berikut contoh sembilan piksel dari citra RGB:

00110101	11010110	11101010
11110100	00111001	11100001
01110001	10010001	11100001

Pesan yang disisipkan adalah karakter "A" dengan nilai biner 01000001. Penyisipan pesan dilakukan menggunakan metode LSB-1, maka *stego image* yang dihasilkan adalah sebagai berikut:

001101**0**1	1101011**0**	1110100**0**
1111010**0**	0011100**0**1	1110000**0**1
0111000**0**1	1001001**1**	11100001

Pada contoh diatas, dapat dilihat bagaimana sistem dari metode LSB-1 pada citra asli ditukar dengan bit dari pesan yang akan disisipkan. Beberapa piksel mengalami perubahan dari piksel asal dan ada pula piksel yang tidak mengalami perubahan sama sekali.

Berdasarkan penelitian yang dilakukan oleh Citra Atika Sari dan Wellia Shinta Sari yang berjudul "Kombinasi *Least Significant Bit* (LSB-1) dan Rivest Shamir Adleman (RSA) dalam Kriptografi Citra Warna", bahwa metode LSB-1 berhasil mendapatkan *stego image* yang tidak terlihat perubahan yang signifikan. Bukti lain dapat dibuktikan dengan menghitung nilai PSNR dan MSE untuk mengetahui perubahan dengan membandingkan piksel pada citra cover (Sari & Sari, 2022).

2.3.9 *Mean Square Error* (MSE)

MSE merupakan nilai eror kuadrat rata-rata antara citra asli (*cover image*) dengan citra hasil steganografi (*stego image*). Perbedaan MSE dan PSNR terletak pada satuannya, dimana PSNR memiliki satuan yaitu db (*decibel*), sedangkan MSE tidak memiliki satuan. Berikut merupakan persamaan untuk menghitung nilai MSE:

$$MSE = \frac{1}{M.N} \sum_{X=1}^{M} \sum_{y=1}^{N} (S_{xy} - C_{xy})^2 \text{..... (3)}$$

Keterangan:

C_{max} = nilai piksel terbesar citra

x dan *y* = koordinat titik citra

M dan *N* = dimensi citra

S = citra tersisipi (*stego image*)

C = citra asli (*cover image*)

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

PSNR merupakan istilah teknik untuk rasio antara kekuatan maksimum yang mungkin dari sebuah sinyal dan kekuatan *noise* yang merusak yang mempengaruhi ketepatan representasinya. PSNR biasanya dinyatakan sebagai besaran logaritmik menggunakan skala desibel dan digunakan untuk mengukur kualitas rekonstruksi untuk gambar dan video yang mengalami *lossy*. Apabila nilai PSNR (Peak Signal Noise Ratio) berada dibawah 30 dB maka itu berarti perbandingan terlihat jelas berbeda dengan gambar aslinya (Prayudi & Prihanto, 2020).

Berikut adalah persamaan yang digunakan pada parameter ini:

$$PSNR = 10\, log_{10} \left(\frac{C_{max}^2}{MSE}\right) \text{..... (4)}$$

BAB III
METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Penelitian ini menggabungkaan teknik kriptografi dan steganografi untuk menyembunyikan *secret message* dari pihak yang tidak bertanggungjawab. Tahapan penelitian dapat dilihat pada Gambar III.1.

Gambar III.1 Diagram Alur Penelitian

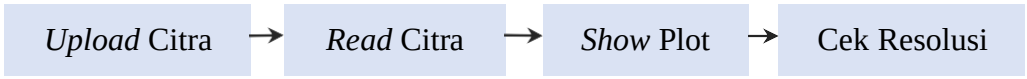
3.2 Analisis *Cover Image*

Teknik *embedding secret message* membutuhkan media sebagai penampungnya. Media yang digunakan pada penelitian ini adalah citra digital (*cover image*) dengan format TIFF. Alasan memilih citra digital sebagai media untuk menyisipkan *secret message* adalah sebagai berikut:

1. Citra digital memiliki kapasitas penyimpanan yang tinggi karena terdiri dari banyak piksel dan setiap piksel menyimpan informasi dalam bentuk nilai warna atau kecerahan. Jika melihat dari segi ini, memungkinkan untuk menyisipkan *secret message* tanpa mengubah tampilan visual citra secara signifikan.
2. Citra digital sangat umum digunakan dalam berbagai konteks. Hal ini dapat mengurangi kecurigaan pihak ketiga mengenai keberadaan *secret message* didalamnya.
3. Citra digital memiliki struktur yang terorganisir dengan baik. Piksel-piksel pada citra tersusun dalam pola tertentu dengan rapi. Jadi, proses *embedding* menjadi lebih terstruktur dan terkendali.

Adapun citra yang digunakan adalah format TIFF dengan alasan sebagai berikut:

1. Format ini sering digunakan untuk menyimpan citra tanpa kompresi, yang berarti tidak ada kehilangan kualitas akibat kompresi data.
2. TIFF mendukung kedalaman bit yang tinggi dan mampu menyimpan informasi lebih banyak pada setiap piksel. Hal ini meningkatkan kapasitas penyimpanan dan memungkinkan penyisipan pesan rahasia dalam jumlah yang lebih besar tanpa mengorbankan kualitas citra secara signifikan.
3. Citra TIFF memiliki toleransi yang tinggi terhadap perubahan pada nilai piksel. Hal ini memungkinkan penyisipan pesan pada bit yang kurang signifikan tanpa menghasilkan perubahan visual yang terlalu mencolok.



Gambar III.2 Diagram Blok Analisis Citra

Tahapan ini menganalisis format citra yang di-*upload* oleh *user* dan menghitung jumlah maksimal karakter yang dapat di-*embed* ke dalam citra. Apabila *user* meng-*upload* selain dari format .TIFF, maka program tidak akan jalan, karena penelitian ini hanya fokus pada citra dengan format tersebut. Resolusi citra kemudian dikalkulasikan untuk mengetahui jumlah maksimal karakter yang dapat di-*embed* ke dalam citra tersebut. Diagram alur analisis citra dapat dilihat Gambar III.2 berikut. Adapun Tabel III.1 merupakan *source code* untuk analisis *cover image*.*User* bebas meng-*upload* gambar apa saja selama citra tersebut berformat TIFF.

1. Membaca citra yang telah di-*upload* dalam mode warna, karena citra yang digunakan adalah citra RGB.
2. Menampilkan plot citra untuk mengetahui resolusi citra. Semakin tinggi resolusi citra, maka semakin besar kapasitas untuk menyembunyikan *encrypted message*.
3. Menganalisis jumlah maksimal karakter.

Tabel III.1 *Source Code* Untuk Analisis *Cover Image*

No	***Source Code***

1	***Upload* Citra**
	`def upload_tiff_image():`
	`  try:`
	`  uploaded_file = files.upload()`
	`  image_key = next(iter(uploaded_file))`
	`  image_content = uploaded_file[image_key]`
	`  img = Image.open(io.BytesIO(image_content))`
	`  img.verify()`
	……

No	*Source Code*

2	***Read* Citra**
	`def show_image_original(img_path):`
	`  def show():`
	`  img = cv2.imread(img_path, cv2.IMREAD_COLOR)`
	`  cv2_imshow(img)  #`
	`  debugPrint('Menampilkan Original Image', show)`

3	***Show Plot***
	`def show_image_plot(img_path):`
	`  def show():`
	`  img = cv2.imread(img_path, cv2.IMREAD_COLOR)`
	`  plt.imshow(img)`
	`  plt.show()`
	`  debugPrint('Menampilkan Plot Image', show)`

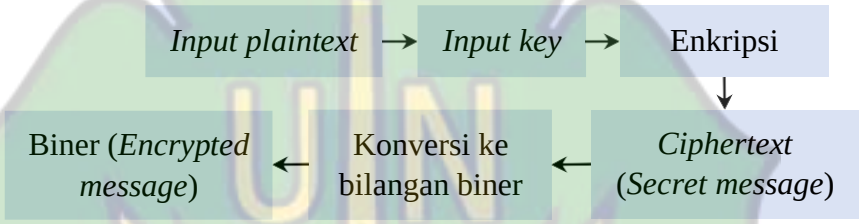
4	**Cek Resolusi**
	`def calculate_image_pixel(img_path):`
	`  img = cv2.imread(img_path, cv2.IMREAD_COLOR)`
	`  height, width, channels = img.shape`
	`  debugPrint('Resolusi Citra', f"{width} x {height} pixel ({width * height} pixel)")`
	……

3.3 Integrasi *Vigènere Cipher* dan DWT-LSB (*Embedding*)

Metode steganografi ini menggabungkan transformasi gelombang diskrit dengan konsep penyisipan pesan pada bit paling akhir dari piksel citra. Diagram alur proses integrasi DWT-LSB dan *vigènere cipher* dapat dilihat pada Gambar III.3 berikut:

Gambar III.3 Diagram Blok Integrasi *Vigènere Cipher* dan DWT-LSB

1. Siapkan *secret message*, yang akan disisipkan ke dalam citra. *Secret message* yang belum dienkripsi dalam kriptografi disebut sebagai *plaintext*, adapun pesan yang telah dienkripsi disebut *ciphertext*. Proses enkripsi membutuhkan *key* untuk mengenkripsi dan dekripsi pesan. Kemudian *input plaintext* dan *key*, lalu *secret message* akan dienkripsi menggunakan *vigènere cipher*.
2. *Encrypted message* kemudian dikonversi menjadi bilangan biner. Berikut penjelasan tahapan *secret message* menjadi *encrypted message* menggunakan *vigènere cipher* (lihat Gambar III.4):



Gambar III.4 Diagram Blok Enkripsi *Vigènere Cipher*

3. Memisahkan *layer* RGB sebelum melakukan proses DWT.
4. Transformasi *wavelet* dimulai dari memilih salah satu *layer* dari *red*, *green* dan *blue*. Selanjutnya menentukan level atau tingkat dekomposisi dan terakhir memilih salah satu *sub-band*, yaitu: hh, hl, lh atau ll untuk menyisipkan *secret message*.
5. Mengambil nilai intensitas *cover image* berdasarkan hasil DWT. Nilai intensitas tersebut kemudian dikonversi menjadi bilangan biner.
6. Bit ke-7 pada hasil biner intensitas *cover image* kemudian di-*replace* dengan hasil biner *encrypted message*. Satu karakter pada *encrypted message* membutuhkan 8 piksel dari citra.
7. Proses *replace* ini merupakan tahap akhir dari *embedding*. Maka jadilah *stego-image* yang berisi *secret message* terenkripsi didalamnya.

Lihat Tabel III.2 yang merupakan *source code* untuk tahapan *embedding* yaitu integrasi *Vigènere Cipher* dan DWT-LSB.

Tabel III.2 *Source Code* Tahap *Embedding*

No.	*Source Code*

1	**Enkripsi *Vigènere Cipher***
	`def vigenere_encrypt(plaintext, key):`
	`  plaintext = plaintext.strip()`
	`  encrypted_text = ''`
	`  key_repeated = (key * (len(plaintext) // len(key) + 1))[:len(plaintext)]`
	……

2	**Pisah *Channel* RGB**
	……
	`def show_splitted_channels(img_path):`
	`  img = cv2.imread(img_path, cv2.IMREAD_COLOR)`
	`  blue, green, red = cv2.split(img)`
	`  fig = plt.figure(figsize = (15, 7.2))`

3	**Proses DWT**
	`img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)`
	`if img.shape[-1] == 1:`
	`  img_rgb = cv2.cvtColor(img, cv2.COLOR_GRAY2RGB)`
	……

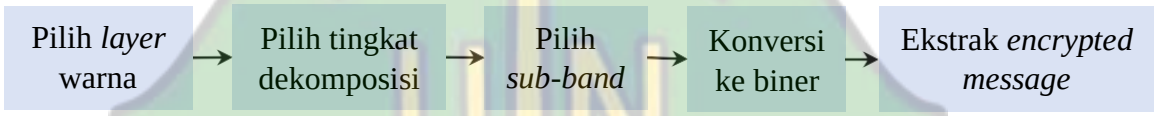
4	**Proses LSB**
	`for i in range(len(subband_data)):`
	`  for j in range(len(subband_data[i])):`
	`  pixel = subband_data[i][j]`
	`  pixel_bin = format(pixel, '08b')`
	……

3.4 Ekstraksi *Stego-image*

Proses ekstraksi *stego-image* adalah tahapan untuk mendapatkan kembali *encrypted message* yang telah disisipkan pada *cover image*. Melakukan transformasi

untuk mendapatkan *stego-image* dalam bentuk aslinya (lihat pada Gambar III.5). Adapun untuk *source code* tahap ekstraksi *stego-image* dapat dilihat pada Tabel III.3.

1. Pilih *layer* warna, tingkat dekomposisi dan *sub-band* tempat *encrypted message* disisipkan.
2. Konversi menjadi bilangan biner.
3. Ekstrak *encrypted message* yang telah disisipkan pada bit ke-7 dari hasil biner nilai intensitas *sub-band*.
4. Selanjutnya adalah tahapan dekripsi untuk membaca kembali *secret message* yang telah dienkripsi.



Gambar III.5 Diagram Blok Ekstraksi *Stego-image*

Tabel III.3 *Source Code* Ekstraksi *Stego-image*

No	*Source Code*

1	***Layer* warna**
	`if channel.lower() == 'r':`
	`  channel_data = r`
	……

2	**Tingkat Dekomposisi**
	`# Lakukan transformasi wavelet Haar`
	`coeffs = pywt.wavedec2(channel_data, 'haar', level=level)`

3	***Sub-band***
	`if subband.lower() == 'll':`
	`  message = extract_message(coeffs[0])`
	`elif subband.lower() == 'lh':`
	……

No	*Source Code*

4	**Konversi ke biner**
	`pixel = subband_data[i][j]`
	`pixel_bin = format(pixel, '08b')`
	……

3.5 Dekripsi *Encrypted Message* (*Ciphertext*)

Selanjutnya adalah tahapan dekripsi yaitu mengubah *ciphertext* menjadi *plaintext*. *Source code* untuk tahap dekripsi dapat dilihat pada Tabel III.4. Berikut penjelasan tentang tahapan proses dekripsi (lihat Gambar III.6):

1. Memisahkan setiap 8-digit biner yang telah diekstrak, lalu mengonversikannya menjadi bilangan ASCII.
2. Saat ini *secret message* masih terenkripsi, maka didekripsi menggunakan *key* untuk mendapatkan bentuk asli dari *secret message* sehingga dapat dibaca oleh penerima.

Gambar III.6 Diagram Blok Dekripsi *Encrypted Message*

Tabel III.4 Source Code Dekripsi Encrypted Message

No	*Source Code*

1	**Konversi biner ke ASCII**
	`# Convert to the appropriate data type (uint8)`
	`subband_data = subband_data.astype(np.uint8)`
	……

2	**Dekripsi *Encrypted Message***
	`if char.isupper():`
	`decrypted_text += chr((ord(char) - shift - ord('A')) % 26 + ord('A'))`
	……

3	***Secret Message* (*Plaintext*)** `return decrypted_text`

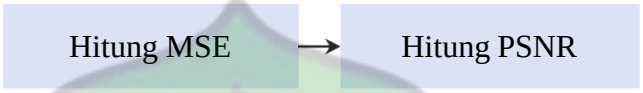
3.6 Pengujian MSE dan PSNR

Mean Squared Error (MSE) dan *Peak Signal-to-Noise Ratio* (PSNR) merupakan dua metrik evaluasi kualitas yang umum digunakan dalam pengujian citra, termasuk saat mengevaluasi perubahan yang terjadi pada *stego-image* dan *cover image* dalam konteks steganografi. Lihat Tabel III.5 untuk *source code* tahap pengujian MSE dan PSNR.

1. Pengujian terhadap *stego image* menggunakan MSE (*Mean Squared Error*) adalah dengan menjumlahkan kuadrat perbedaan antara setiap piksel dalam *cover image* dan *stego image*. Rumus MSE dapat dihitung dengan mengambil selisih nilai piksel pada setiap posisi dan mengkuadratkannya, kemudian menghitung rata-rata dari seluruh nilai tersebut.
2. PSNR memberikan ukuran yang lebih intuitif tentang kualitas citra dengan menghitung perbandingan antara energi sinyal maksimum dan MSE. Maksimal sinyal energi dapat dihitung dengan mengkuadratkan nilai maksimum

representasi piksel pada citra. PSNR diukur dalam satuan desibel (dB). (lihat Gambar III.7).

3. Hasil pengujian MSE dan PSNR memberikan informasi tentang kualitas citra yang diuji. Semakin tinggi nilai PSNR atau semakin rendah nilai MSE, maka semakin baik kualitas dari *stego image* tersebut.



Gambar III.7 Diagram Blok Pengujian MSE dan PSNR

Tabel III.5 *Source Code* Pengujian MSE dan PSNR

No	*Source Code*

1	**MSE**
	`# Hitung Mean Squared Error (MSE) menggunakan scikit-image`
	`mse_skimage = mean_squared_error(gray1, gray2)`

2	**PSNR**
	`# Hitung Peak Signal-to-Noise Ratio (PSNR) menggunakan scikit-image`
	`psnr_skimage = peak_signal_noise_ratio(gray1, gray2)`

3.7 Alat Bantu Penelitian

Alat bantu pada penelitian ini adalah *hardware* dan *software* komputer, Spesifikasi *hardware* dan *software* yang digunakan untuk mengimplementasikan sistem ini dapat dilihat pada Tabel III.6 berikut.

Tabel III.6 Spesifikasi *Hardware* dan *Software*

No	Spesifikasi	Detail
1	*Hardware*	
	Model Sistem	HP Laptop Model 14s-CF0069TX
	Processor	Intel ® Core™ i3-7020U CPU @ 2.30GHz
	Memori	8.00 GB
	Hard disk	111 GB SSD + 200 GB HDD
2	*Software*	
	Operating System	Windows 10 Pro 64-bit *operating system*, 64-bit *based processor*
	Programming Tool	Google Colaboration
3	Jenis Citra	TIFF
4	Teknik Enkripsi	Algoritma *Vigènere Cipher*
5	Teknik *Embedding*	*Discrete Wavelet Transform* dan *Least Significant Bit* (DWT-LSB)

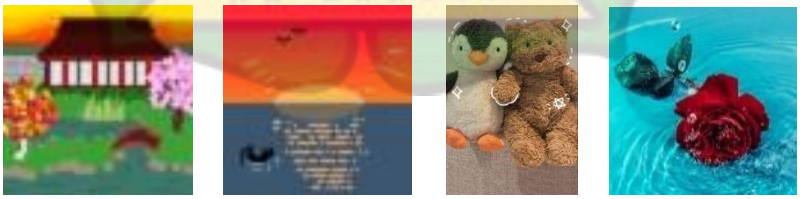
BAB IV
HASIL PENELITIAN DAN PEMBAHASAN

Bab ini menjelaskan proses *embedding encrypted message* pada *cover image* dengan format .tiff menggunakan metode DWT-LSB. Pada penelitian ini penulis melakukan uji coba terhadap beberapa citra. Hasil uji coba kemudian dilakukan analisa untuk mendapatkan sebuah kesimpulan sebagai akhir dari proses penelitian.

Eksperimen pada penelitian ini dilakukan menggunakan objek citra RGB sebanyak 4 citra format TIFF dengan ukuran yang bervariasi. Setiap citra juga di-*embed* dengan *secret message* yang berbeda. Implementasi ini dilakukan menggunakan Google Collaboration dengan metode yang telah diusulkan dan dijelaskan pada bab sebelumnya.

4.1 Hasil Uji Coba Kombinasi Steganografi dan Kriptografi

Citra yang digunakan adalah citra dengan format TIFF, *cover image* yang diuji coba pada penelitian ini ditujukan pada Gambar IV.1. Terdiri dari 4 citra yang berbeda dengan resolusi yang berbeda juga. Secara kasat mata, terlihat dengan jelas perbedaan kualitas antara citra 1, 2, 3 dan 4. Citra 1 dan 2 memiliki resolusi yang lebih rendah dibandingkan dengan citra 3 dan 4.



(Citra 1) (Citra 2) (Citra 3) (Citra 4)

Gambar IV.1 *Cover Image*

(sumber: fiverr.com)

Berikut adalah hasil perolehan dari uji coba proses enkripsi pada *cover image* menggunakan algoritma *vigènere cipher* yang dijelaskan secara berurutan dari Tabel IV.1 sampai dengan Tabel IV.4.

Tabel IV.1 Hasil Enkripsi dan *Stego-image* pada Citra 1

Proses	**Hasil**
Dimensi / Resolusi	64 x 64 / 4096
Maksimal Karakter	512 karakter
Secret Message	P : sintabermainditaman K : sorehari
Ciphertext (*Encrypted Message*)	kwexhbvzeozrkikieoe
Hasil *Stego-image*	
Status Dekripsi	BERHASIL DI DEKRIPSI
Nilai MSE	0.45361328125
Nilai PSNR	51.563945991780614 dB

Pada Tabel IV.1, perolehan nilai MSE dan PSNR menunjukkan kualitas yang sangat baik. Nilai MSE yang rendah menunjukkan seberapa kecil kesalahan antara *cover image* dan *stego-image*. Sedangkan nilai PSNR yang tinggi (>30 dB) dianggap baik.

Tabel IV.2 Hasil Enkripsi dan *Stego-image* pada Citra 2

Proses	Hasil
Dimensi / Resolusi	64 x 64 / 4096
Maksimal Karakter	512 karakter
Secret Message	P: perempuanyangberdiriitu K: hijau
Ciphertext (*Encrypted Message*)	wmaegwcjnshvpbyylrrcpbd
Hasil *Stego-image*	
Status Dekripsi	BERHASIL DI DEKRIPSI
Nilai MSE	30.263427734375
Nilai PSNR	33.32162244787588 dB

Pada Tabel IV.2 yang memiliki nilai MSE hampir mendekati nilai PSNR. Ini menunjukkan kemungkinan adanya kesalahan selama proses *embedding*. Nilai MSE yang bagus adalah yang mendekati angka 0, sedangkan pada Tabel IV.2 nilai MSE menunjukkan angka >30.

Perhatikan Tabel IV.3 dan Tabel IV.4, nilai PSNR pada kedua tabel ini sangat tinggi. Standar nilai PSNR berkisar antara 30 dB hingga 40 dB, namun jika melebihi 50 dB, pada umumnya dianggap sebagai kualitas citra yang sangat tinggi (Aziza & Imah, 2019). Seluruh *secret message* berhasil di enkripsi dan di-*embed* ke dalam *cover image*. Pesan tersebut juga dapat di ekstrak dari citra dan di kembalikan ke bentuk aslinya, bentuk yang bisa di baca.

Tabel IV.3 Hasil Enkripsi dan *Stego-image* pada Citra 3

Proses	Hasil
Dimensi / Resolusi	236 x 339 / 80004
Maksimal Karakter	10000 karakter
Secret Message	P: kuncinyaadadisaku K: kucing
Ciphertext (*Encrypted Message*)	uopkvtiuclnjsmcsh
Hasil *Stego-image*	
Status Dekripsi	BERHASIL DI DEKRIPSI
Nilai MSE	0.004374781260936953
Nilai PSNR	71.72124017690822 dB

Tabel IV.4 Hasil Enkripsi dan *Stego-image* pada Citra 4

Proses	Hasil
Dimensi / Resolusi	669 x 750 / 501750
Maksimal Karakter	62718 karakter
Secret Message	P: meskiSeringbertEngKAR,tapikitawajib salingbantusaAtadayaNGkesuLitandiAntaRakIta K: merah

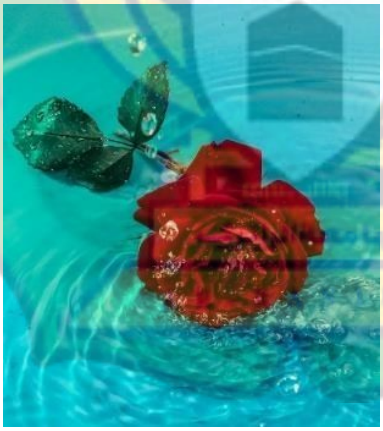
Proses	Hasil
***Ciphertext* (*Encrypted Message*)**	yijkpEiiiusfvraQrxKHD,kawuozthieaiieec iusfrnagwrAamhryhZKbezgPzthzhzAufeIarUxr
Hasil *Stego-image*	
Status Dekripsi	BERHASIL DI DEKRIPSI
Nilai MSE	0.022539113104135526
Nilai PSNR	64.60143537958089 dB

Kedua metrik ini digunakan untuk mengevaluasi kualitas citra, namun nilai MSE dan PSNR tidak selalu mencerminkan persepsi visual manudia. Oleh karena itu, dalam beberapa kasus, metrik evaluasi subjektif juga dapat diperlukan untuk mendapatkan pemahaman yang lebih baik tentang kualitas citra. Metrik evaluasi subjektif melibatkan peran manusia untuk mengukur kualitas suatu sistem atau proses. Dalam konteks ini, metrik evaluasi subjektif digunakan untuk mendapatkan penilaian langsung terkait perbandingan *cover image* dan *stego-image* dari sudut pandang visual manusia.

Pada Tabel IV.5, peneliti membandingkan *cover image* dan *stego-image* dengan evaluasi subjektif. *Stego-image* yang baik adalah yang terlihat hampir sama atau hampir tidak ada perbedaan dengan *cover image*.

Tabel IV.5 Perbandingan Cover Image dan Stego-image

Nama Citra	***Cover Image***	***Stego-image***
Citra 1	 Dimensi : 64 x 64 *Size* : 8,27 kb	 Dimensi : 64 x 64 *Size* : 6,39 kb
Citra 2	 Dimensi : 64 x 64 *Size* : 3,46 kb	 Dimensi : 64 x 64 *Size* : 3,54 kb

Nama Citra	*Cover Image*	*Stego-image*
Citra 3	 Dimensi : 236 x 339 *Size* : 285 kb	 Dimensi : 236 x 339 *Size* :159 kb
Citra 4	 Dimensi : 669 x 750 *Size* : 1,31 mb	 Dimensi : 669 x 750 *Size* : 705 kb

Evaluasi subjektif pada Tabel IV.5, mengindikasikan adanya perubahan yang terjadi pada *stego-image*. Pada *cover image* Citra 1 dan 3, tidak terlihat adanya perbedaan dengan *stego-image*. Namun perbedaan yang signifikan terlihat jelas pada Citra 2, hasil *stego-image* menjadi lebih cerah dari *cover image*. Berbeda dengan Citra 4, apabila hanya melirik sekilas hampir tidak ada perubahan yang terjadi.

Selain pada tampilan (warna), perubahan juga terjadi pada *size* citra. *Stego-image* cenderung memiliki *size* yang lebih kecil dari *cover image*. Hal ini terjadi pada Citra 1, Citra 3 dan Citra 4. Berbeda dengan citra 2 yang memiliki *size stego-image* lebih besar dari *cover image.*

4.2 Analisis Hasil Evaluasi

Dalam sub-bab analisis hasil evaluasi, penulis mengeksplorasi dan menginterpretasikan data yang diperoleh dari metrik evaluasi yang telah diterapkan. Melibatkan metrik objektif dan subjektif, analisis ini bertujuan untuk memberikan wawasan tentang kualitas dan kinerja sistem. Setelah melakukan uji coba pada 4 citra, berikut merupakan analisis hasil evaluasi penulis.

1. Interpretasi Metrik Objektif

   Pada metrik objektif, penulis menggunakan pengujian MSE dan PSNR. Nilai MSE yang rendah dan PSNR yang tinggi pada umumnya mengindikasikan kualitas citra yang baik. Dari 4 citra yang di uji coba, terindikasi 3 citra yang memiliki kualitas baik. Sedangkan citra yang satunya, mendapat nilai MSE yang hampir dekat dengan nilai PSNR, yaitu 30 dB. Hal ini menunjukkan bahwa tingkat kesalahan atau distorsi pada citra atau data yang relatif rendah dibandingkan dengan energi sinyal.

Tabel IV.6 Analisis MSE dan PSNR

Nama Citra	MSE	PSNR
Citra 1	Hasil: 0.45361328125 Kualitas: Baik	Hasil: 51.563945991780614 dB Kualitas: Baik
Citra 2	Hasil: 30.263427734375 Kualitas: Tidak baik	Hasil: 33.32162244787588 dB Kualitas: Baik
Citra 3	Hasil: 0.004374781260936953 Kualitas: Baik	Hasil: 71.72124017690822 dB Kualitas: Baik
Citra 4	Hasil: 0.022539113104135526 Kualitas: Baik	Hasil: 64.60143537958089 dB Kualitas: Baik

Pada Tabel IV.6, rata-rata seluruh citra mendapatkan hasil yang baik. Nilai MSE yang baik adalah yang mendekati angka nol dan nilai PSNR diatas 30 dB merupakan nilai yang baik sehubungan dengan kualitas citra. Meskipun nilai MSE dan PSNR memberikan indikasi kualitas, namun evaluasi subjektif manusia dapat dijadikan sebagai perspektif tambahan.

2. Evaluasi Metrik Subjektif

Dalam analisis evaluasi metrik subjektif pada *stego-image* dan *cover image*, penulis berfokus pada pemahaman pandangan manusia terhadap kualitas dan keaslian *stego-image* dibandingkan dengan *cover image*. Hasil dari analisis ini memberikan wawasan yang berharga terkait dengan sejauh mana *stego-image* berhasil menyamarkan data tersembunyi tanpa mengorbankan kualitas visual yang signifikan, serta sejauh mana keaslian citra dapat dipertahankan. Tabel IV.7 berikut adalah analisis metrik subjektif terhadap keempat citra yang di uji coba.

Table IV.7 Analisis Evaluasi Subjektif Manusia Terhadap *Stego-image* dan *Cover Image*

Citra	Hasil Analisis
Citra 1	Perbedaan hampir tidak terlihat. Hampir tidak ada perubahan pada *stego-image*. Kedua citra kelihatan mirip.
Citra 2	Perbedaan terlihat jelas. Warna langit yang awalnya kemerah-merahan, menjadi merah jambu, warna laut juga berubah menjadi biru cerah pada *stego-image*.
Citra 3	Perbedaan hampir tidak terlihat. Hampir tidak ada perubahan pada *stego-image*. Kedua citra kelihatan mirip.
Citra 4	Perbedaan tidak terlihat jelas. *Stego-image* kelihatan sedikit lebih cerah daripada *cover image*. Namun, sangat tipis perbedaan antara kedua citra.

BAB V
KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil pembahasan dan penelitian yang telah diuraikan pada bab sebelumnya, maka dapat disimpulkan sebagai berikut:

1. Berdasarkan hasil percobaan, program dapat mengamankan *secret message* dengan baik menggunakan algoritma *Vigènere Cipher* dan DWT-LSB. Kombinasi DWT-LSB membuat penyembunyian *secret message* menjadi lebih aman. Hasil analisis metrik secara objektif dengan menggunakan MSE dan PSNR menunjukkan kualitas citra yang baik. Dari percobaan terhadap keempat citra, diperoleh nilai PSNR terbaik yaitu 71.72124017690822 dB dengan nilai MSE sebesar 0.004374781260936953. Adapun hasil analisis metrik secara subjektif terhadap *stego-image* dan *cover image* mendukung dan sesuai dengan hasil metrik objektif.
2. Jumlah maksimal karakter yang dapat di-*embed* ke dalam citra, tergantung pada resolusi citra tersebut. Semakin tinggi resolusinya, maka semakin banyak karakter yang dapat di-*embed*.
3. *Secret Message* pada *stego-image* dapat diekstrak kembali dan didekripsi ke bentuk aslinya. Apabila *secret message* berhasil terdeteksi oleh pihak yang tidak bertanggung jawab, pesan akan sulit di baca karena telah dienkripsi.

5.2 Saran

Berdasarkan hasil percobaan pada penelitian ini, penulis merasa dalam penulisan ini masih memiliki kekurangan antara lain sebagai berikut serta sarannya:

1. Enkripsi menggunakan *vigènere cipher* hanya bisa mengenkripsi huruf dan tidak dapat mengenkripsi karakter lainnya, sehingga yang dienkripsi hanya berupa teks.

2. Diharapkan ketika penelitian ini dilanjutkan, dapat menciptakan program yang lebih baik lagi.
3. Program masih dalam tahap pengembangan, sehingga mungkin akan ada kendala apabila digunakan dalam jangka panjang. Peneliti berharap, agar peneliti selanjutnya dapat menyempurnakan program ini.

DAFTAR PUSTAKA

Afsari, M., Mulyana, D. I., Damaiyanti, A., & Sa'adah, N. (2022). Implementasi Mode Operasi Kombinasi Cipher Block Chaining dan Metode LSB-1 Pada Pengamanan Data Text. *Jurnal Pendidikan Sains Dan Komputer*, *2*(01), 70–82. https://doi.org/10.47709/jpsk.v2i01.1381

Alawiyah, T., Ardianto, R., & Purnia, D. S. (2020). Implementasi Vigenere Cipher Sebagai Pengaman Pada Proses Deskripsi Steganografi Least Significant Bit. *Jurnal Informatika*, *7*(1), 37–45. https://doi.org/10.31311/ji.v7i1.6431

Ibrahim, A. (2021). *Menggabungkan Teknik Steganografi Discrete Wavelet Transform Dua Dimensi (2-D) dan Algoritma Kriptografi RSA pada Perancangan dan Analisis Keamanan Pesan*.

Jevandika., Prasetio, Barlian H., & Syauqy, Dahnial. (2023). Rancang Bangun Alat Pengenal *Finger Vein* Menggunakan Raspberry PI Dengan Metode *Convolutional Neural Network* (CNN).

Marleny, Finki D. (2021). Pengolahan Citra Digital Menggunakan Python. p.6-11, 49-51

Marpaung, A., Ramadhan, P. S., & Pranata, A. (2023). *Implementasi RSA Untuk Enkripsi Dan Dekripsi File Dokumen*. *2*, 39–48.

Marpaung, Arnold., Ramadhan, Puji S., & Pranata, Adianto. (2023). Implementasi RSA Untuk Enkripsi dan Dekripsi File Dokumen. p.41

Marpaung, Faridawaty., Aulia, Fitrahuda., Suryani, Nita., & Nabila, Rinjani C. (2022). *Computer Vision* dan Pengolahan Citra Digital. p.13-16

Prayudi, Fino Ardiansyah. & Prihanto Agus. (2020). Penerapan Algoritma *Least Significant Bit* Untuk Menyembunyikan *Vigenere Cipher Text* pada Citra Digital, 144-148. https://doi.org/10.26740/jinacs.v1n03.p144-149.

Richards, J. A. (2022). *Remote Sensing Digital Image Analysis*. Vol. 5, Issue 3. p.256-258.

Rudhistiar, D. (2022). Implementasi Pengamanan Citra Digital Berbasis Enkripsi 2D Logistic Map Dan Dna Encoding Dengan Penyisipan Lsb Dan Dwt. *Jurnal Mnemonic*, *5*(1), 45–50. https://doi.org/10.36040/mnemonic.v5i1.4436

Ruing, M. O. I. (2020). *Penerapan Kombinasi Algoritma Kriptografi (Caesar, Vigenere, Zig-Zag) dan Metode Steganografi LSB untuk Mengamankan Pesan ke Dalam Citra Digital*. http://eprints.uty.ac.id/4888/

Sari, C. A., & Sari, W. S. (2022). Kombinasi Least Significant Bit (LSB-1) Dan Rivest Shamir Adleman (RSA) Dalam Kriptografi Citra Warna. *Jurnal Masyarakat Informatika*, *13*(1), 45–58. https://doi.org/10.14710/jmasif.13.1.43314

Umam, C., Muslih, M., & Fadillah, D. (2022). Kombinasi Steganografi LSB dan Kriptografi AES dalam Sekuriti Teks Rahasia Pada Citra Berwarna. *Seminar Nasional Teknologi Dan Multidisiplin Ilmu (SEMNASTEKMU)*, *2*(1), 109–118. https://doi.org/10.51903/semnastekmu.v2i1.160

Yanu, M., Yuwono, B., & Boedi, D. (2022). Dasar Pengolahan Citra Digital Edisi 2022. p.1-13.

LAMPIRAN

Lampiran berikut adalah *source code* dari penelitian ini:

1. Source code analisis *cover image*

Source code untuk *import library* pada *python* dan *debug*.

```
import io
import cv2
import pywt
import numpy as np
import pandas as pd
import imageio
from PIL import Image
from google.colab import files
from google.colab.patches import cv2_imshow
from matplotlib import pyplot as plt
from itertools import cycle
from skimage.metrics import mean_squared_error,
     peak_signal_noise_ratio
from skimage import color

# Debug
def debugPrint(header, value):
  print(header)
  if callable(value):
    value()
  else:
    print(value)
  print()
```

Source code untuk *upload* citra (*cover image*).

```
def upload_tiff_image():
  try:
    uploaded_file =  files.upload()
    image_key = next(iter(uploaded_file))
    image_content = uploaded_file[image_key]
    img = Image.open(io.BytesIO(image_content))
    img.verify()
```

```
    if img.format.lower() != 'tiff':
      print("Format citra tidak valid. Harap unggah citra dengan
            format .tiff")
      return None
    return image_key

  except:
    return None

    return image_content
```

Source code untuk membaca citra yang telah di-*upload*.

```
def show_image_original(img_path):
  def show():
    img = cv2.imread(img_path, cv2.IMREAD_COLOR)
    cv2_imshow(img)
  debugPrint('Menampilkan Original Image', show)
```

Source code untuk menampilkan plot citra.

```
def show_image_plot(img_path):
  def show():
    img = cv2.imread(img_path, cv2.IMREAD_COLOR)
    plt.imshow(img)
    plt.show()
  debugPrint('Menampilkan Plot Image', show)
```

Source code untuk memeriksa resolusi citra dengan menghitung total piksel dan jumlah maksimal karakter yang dapat di-*embedd* ke dalam citra.

```
def calculate_image_pixel(img_path):
  img = cv2.imread(img_path, cv2.IMREAD_COLOR)
  height, width, channels = img.shape
  debugPrint('Resolusi Citra', f"{width} x {height} pixel ({width
             * height} pixel)")

  pixel = width * height
  max = pixel // 8
  print("Jumlah max karakter yang dapat disisipkan pada citra =",
        max)
  print()
```

```
  debugPrint('Tipe Data', img.dtype)
  debugPrint('Jumlah Channel', f"{channels}")

  return width * height
```

2. *Source code* integrasi *Vigènere Cipher* dan DWT-LSB (*Embedding*)

Source code untuk enkripsi algoritma *Vigènere Cipher*.

```
def vigenere_encrypt(plaintext, key):
    plaintext = plaintext.strip()
    encrypted_text = ''
    key_repeated = (key * (len(plaintext) // len(key) +
                   1))[:len(plaintext)]

    for i in range(len(plaintext)):
        char = plaintext[i]
        if char.isalpha():
            shift = ord(key_repeated[i].upper()) - ord('A')
            if char.isupper():
                encrypted_text += chr((ord(char) + shift -
                ord('A')) % 26 + ord('A'))
            else:
                encrypted_text += chr((ord(char) + shift -
                ord('a')) % 26 + ord('a'))
        else:
            encrypted_text += char

    print()
    debugPrint('Plaintext : ', plaintext)
    debugPrint('Key :', key)
    debugPrint('Chipertext (encrypted message) : ',
    encrypted_text)

    binary_result = ''
    for char in encrypted_text:
        binary_result += format(ord(char), '08b')

    debugPrint('Encrypted Message In Binary : ', binary_result)

    return binary_result
```

Source code untuk memisahkan *channel* RGB.

```
def show_splitted_channels(img_path):
  img = cv2.imread(img_path, cv2.IMREAD_COLOR)
  blue, green, red = cv2.split(img)
  fig = plt.figure(figsize = (15, 7.2))

  fig.add_subplot(133)
  plt.title("CHANNEL MERAH")
  plt.imshow(blue)

  fig.add_subplot(132)
  plt.title("CHANNEL HIJAU")
  plt.imshow(green)

  fig.add_subplot(131)
  plt.title("CHANNEL BIRU")
  plt.imshow(red)

  print('Menampilkan splitted channels')
  plt.show()
  print()
```

Source code untuk proses transformasi *wavelet* (DWT).

```
def embed_wavelet_subband(image_path, channel, level, subband,
message):
  img = cv2.imread(image_path, cv2.IMREAD_COLOR)

  img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
  if img.shape[-1] == 1:
    img_rgb = cv2.cvtColor(img, cv2.COLOR_GRAY2RGB)
  else:
    img_rgb = img

  r, g, b = cv2.split(img_rgb)

  if channel.lower() == 'r':
    channel_data = r
  elif channel.lower() == 'g':
    channel_data = g
  elif channel.lower() == 'b':
    channel_data = b
```

```
  else:
    print("Kanal warna tidak valid.")
    return None

  coeffs = pywt.wavedec2(channel_data, 'haar', level=level)

  if subband.lower() == 'll':
    coeffs[0] = embed_message(coeffs[0], message)
  elif subband.lower() == 'lh':
    coeffs[level][0] = embed_message(coeffs[level][0], message)
  elif subband.lower() == 'hl':
    coeffs[level][1] = embed_message(coeffs[level][1], message)
  elif subband.lower() == 'hh':
    coeffs[level][2] = embed_message(coeffs[level][2], message)
  else:
    print("Subband tidak valid.")
    return None

  embedded_img_channel = pywt.waverec2(coeffs, 'haar')
  embedded_img_channel = np.clip(embedded_img_channel, 0,
                                 255).astype(np.uint8)

  if channel.lower() == 'r':
    embedded_img_rgb = cv2.merge([embedded_img_channel, g, b])
  elif channel.lower() == 'g':
    embedded_img_rgb = cv2.merge([r, embedded_img_channel, b])
  elif channel.lower() == 'b':
    embedded_img_rgb = cv2.merge([r, g, embedded_img_channel])
  else:
    print("Invalid color channel.")
    return None

  return embedded_img_rgb
```

Source code untuk *replace* bit ke-7 pada hasil biner intensitas *cover image* dengan hasil biner *encrypted message*.

```
def embed_message(subband_data, message):
  def replace_each_7th_pixel(subband_data, pesan_bin):
    pesan_bin_cycled = cycle(pesan_bin)

    for i in range(len(subband_data)):
```

```
        for j in range(len(subband_data[i])):
            pixel = subband_data[i][j]
            pixel_bin = format(pixel, '08b')
            pixel_bin = pixel_bin[:-1] + next(pesan_bin_cycled)
            subband_data[i][j] = int(pixel_bin, 2)

    return subband_data

  min_val = subband_data.min()
  max_val = subband_data.max()

  if min_val != max_val:
      subband_data = (subband_data - min_val) * (255.0 / (max_val
                     - min_val))
      subband_data = subband_data.astype(np.float32) + min_val
  else:
      subband_data = subband_data * 255

  subband_data = subband_data.astype(np.uint8)
  steganographed_img = replace_each_7th_pixel(subband_data,
message)

  binary_string_with_sign = pd.DataFrame(data=steganographed_img)
  binary_string_with_sign = binary_string_with_sign.applymap(
                            lambda x: f'{x:08b}')

  decimal_subband_with_sign = binary_string_with_sign.applymap(
                              lambda x: int(x, 2))

  return decimal_subband_with_sign.values
```

3. *Source code* untuk ekstraksi *stego-image*

Source code untuk memilih *layer* warna, tingkat dekomposisi dan *sub-band* tempat *encrypted message* disisipkan.

```
def extract_wavelet_subband(img, channel, level, subband):
  img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
  if img.shape[-1] == 1:
    img_rgb = cv2.cvtColor(img, cv2.COLOR_GRAY2RGB)
  else:
    img_rgb = img
```

```
  r, g, b = cv2.split(img_rgb)

  if channel.lower() == 'r':
    channel_data = r
  elif channel.lower() == 'g':
    channel_data = g
  elif channel.lower() == 'b':
    channel_data = b
  else:
    print("Kanal warna tidak valid.")
    return None

  coeffs = pywt.wavedec2(channel_data, 'haar', level=level)

  if subband.lower() == 'll':
    message = extract_message(coeffs[0])
  elif subband.lower() == 'lh':
    message = extract_message(coeffs[level][0])
  elif subband.lower() == 'hl':
    message = extract_message(coeffs[level][1])
  elif subband.lower() == 'hh':
    message = extract_message(coeffs[level][2])
  else:
    print("Subband tidak valid.")
    message =  ''

  debugPrint('Message dalam binary', message)

  chunks = [message[i:i+8] for i in range(0, len(message), 8)]
  message = ''.join(chr(int(chunk, 2)) for chunk in chunks)

  debugPrint('Message dalam text', message)

  return message
```

4. *Source code* untuk dekripsi *encrypted message* (*Ciphertext*)

Source code untuk konversi bilangan biner menjadi bilangan ASCII.

```
def extract_message(subband_data):
  min_val = subband_data.min()
  max_val = subband_data.max()
  if min_val != max_val:
```

```
        subband_data = (subband_data - min_val) * (255.0 / (max_val
                       - min_val))
        subband_data = subband_data.astype(np.float32) + min_val
    else:
        subband_data = subband_data * 255

    subband_data = subband_data.astype(np.uint8)

    pesan_bin = ''
    for i in range(len(subband_data)):
        for j in range(len(subband_data[i])):
            pixel = subband_data[i][j]
            pixel_bin = format(pixel, '08b')
            pesan_bin += pixel_bin[-1]

    return pesan_bin.split('00100011')[0]
```

Source code untuk mengubah *encrypted message* ke bentuk asli (*secret message*).

```
def vigenere_decrypt(ciphertext, key):
    decrypted_text = ''
    key_repeated = (key * (len(ciphertext) // len(key) +
                   1))[:len(ciphertext)]

    for i in range(len(ciphertext)):
        char = ciphertext[i]
        if char.isalpha():
            shift = ord(key_repeated[i].upper()) - ord('A')
            if char.isupper():
                decrypted_text += chr((ord(char) - shift -
                ord('A')) % 26 + ord('A'))
            else:
                decrypted_text += chr((ord(char) - shift -
                ord('a')) % 26 + ord('a'))
        else:
            decrypted_text += char

    return decrypted_text
```

5. *Source code* untuk melakukan uji MSE dan PSNR

```
def compare_images(img1, img2):
    gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
    gray2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

    mse_numpy = np.sum((gray1 - gray2) ** 2) / float(
                gray1.shape[0] * gray1.shape[1])

    mse_skimage = mean_squared_error(gray1, gray2)

    psnr = cv2.PSNR(gray1, gray2)

    psnr_skimage = peak_signal_noise_ratio(gray1, gray2)

    return mse_numpy, mse_skimage, psnr, psnr_skimage
```

Source code untuk menjalankan seluruh perintah *coding* sebelumnya.

```
def main():
  uploaded_image = upload_tiff_image()

  if not uploaded_image:
    return

  show_image_original(uploaded_image)
  show_image_plot(uploaded_image)
  calculate_image_pixel(uploaded_image)
  show_splitted_channels(uploaded_image)

  print()
  print('INPUT SECRET MESSAGE YANG AKAN DI EMBED KE GAMBAR : ')
  msg_text = input('Input Plaintext : ')
  msg_key = input('Input Key       : ')

  print()
  print('TENTUKAN METODE EMBED')
  rgb = input('Pilih Layer (R/G/B) : ')
  level = int(input('Pilih Tingkat Dekomposisi (0-7) : '))
  subband = input('Pilih Subband (LL/LH/HL/HH) : ')

  encrypted_msg = vigenere_encrypt(msg_text, msg_key)
```

```
    image_with_message = embed_wavelet_subband(uploaded_image, rgb,
                          level, subband, encrypted_msg)

    print('Menampilkan Stego Image')
    cv2_imshow(image_with_message)
    print()

    extracted_message = extract_wavelet_subband(image_with_message,
                         rgb, level, subband)

    decrypted_text = vigenere_decrypt(extracted_message, msg_key)

    debugPrint('Decrypted Message', decrypted_text)

    original_image = cv2.imread(uploaded_image)
    mse_numpy, mse_skimage, psnr, psnr_skimage = compare_images(
                                                  original_image,
                                                  image_with_message
                                                  )

    print(f"MSE (NumPy): {mse_numpy}")
    print(f"MSE (scikit-image): {mse_skimage}")
    print(f"PSNR (OpenCV): {psnr} dB")
    print(f"PSNR (scikit-image): {psnr_skimage} dB")

main()
```