BAB III
METODE PENELITIAN

3.1 Identifikasi Masalah

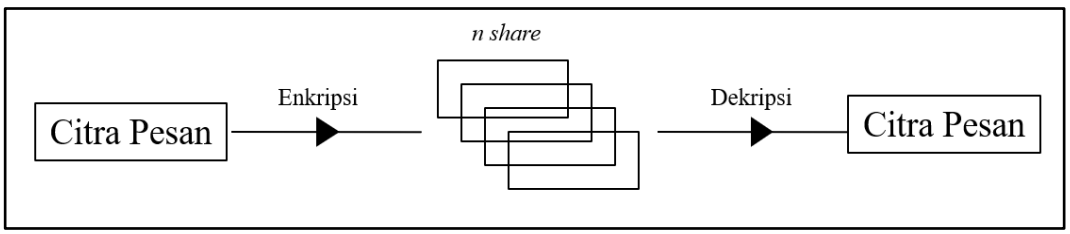
Informasi menjadi suatu hal yang penting dalam era digital modern. Citra digital merupakan salah satu informasi yang paling sering digunakan untuk melakukan komunikasi. Pengiriman citra digital yang bersifat penting perlu ditambahkan keamanan untuk menjaga kerahasiannya. Salah satu cara untuk mengamankan citra adalah dengan menggunakan kriptografi visual. Kriptografi visual dapat menyamarkan citra sehingga citra menjadi abstrak dan sulit untuk dikenali. Namun, karena citra berbentuk abstrak, hal tersebut dapat menimbulkan kecurigaan. Oleh karena itu, diperlukan adanya keamanan tambahan dengan cara menyisipkan citra abstrak tersebut ke citra lain menggunakan steganografi. Penyamaran dan penyisipan pesan dilakukan dengan menggabungkan kriptografi *Visual Secret Sharing* dan steganografi *Enhanced Least Significant Bit*.

3.2 Model Dasar

Model dasar yang digunakan dalam penelitian ini adalah kriptografi *Visual Secret Sharing* dan steganografi *Enhanced Least Significant Bit*.

3.2.1 *Visual Secret Sharing*

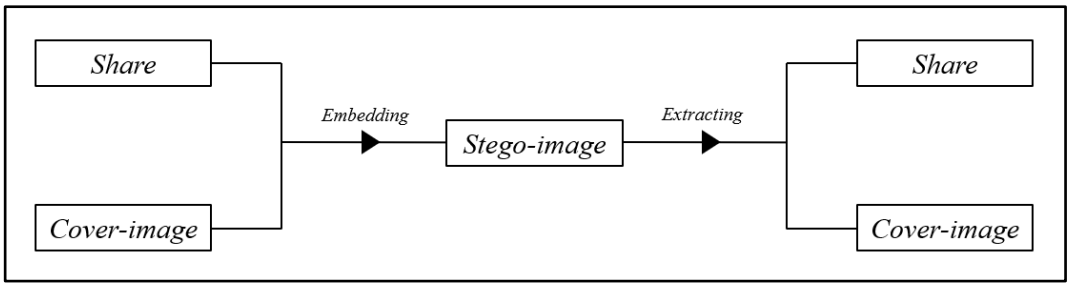
Visual Secret Sharing adalah salah satu metode kriptografi untuk merahasiakan citra dengan cara membagi citra sebanyak *n shares*. *Shares* yang diperoleh merupakan gambar abstrak yang apabila seluruh *shares* digabungkan, citra pesan dapat diperoleh kembali.



Gambar 3.1 Skema Kriptografi Visual *Secret Sharing*

3.2.2 *Enhanced Least Significant Bit*

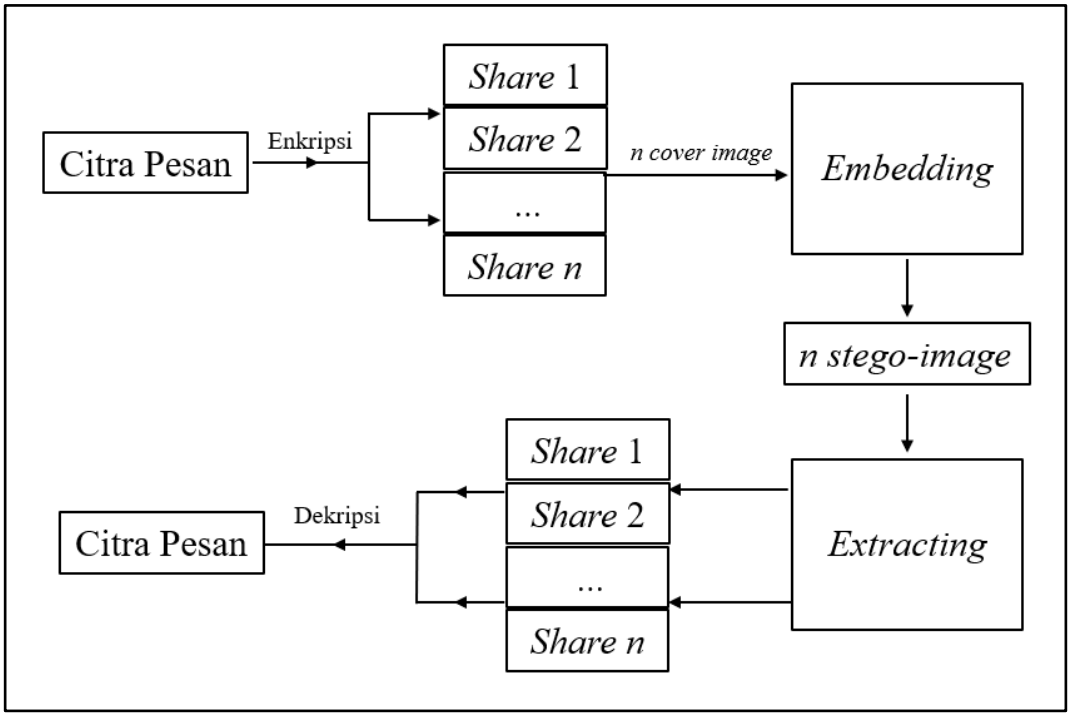
Least Significant Bit adalah metode steganografi untuk menyembunyikan pesan dengan mengubah bit terakhir dari *cover image* menjadi sebuah pesan. *Enhanced Least Significant Bit* merupakan metode LSB yang ditingkatkan, di mana bit yang diubah menjadi pesan adalah dua bit terakhir dari piksel atau sama dengan bit ke-7 dan bit ke-8.



Gambar 3.2 Skema Steganografi *Enhanced Least Significant Bit*

3.3 Pengembangan Model Dasar

Pada penelitian ini, pengembangan model dilakukan dengan menggabungkan kedua metode pada model dasar, yaitu kriptografi *Visual Secret Sharing* dan steganografi *Enhanced Least Significant Bit*. Langkah pertama yang dilakukan adalah pesan citra *grayscale* dienkripsi menggunakan skema *Visual Secret Sharing* sehingga menghasilkan *n buah share*. Hasil enkripsi kemudian di-*embed* ke dalam *n cover image* berwarna berformat *.png menggunakan metode *Enhanced Least Significant Bit* sehingga diperoleh *n stego-image*. Untuk mendapatkan kembali pesan citra *grayscale* yang ada di dalam *stego-image*, dilakukan proses *extracting* dengan mengambil dua bit terakhir atau bit ke-7 dan bit ke-8 dari setiap piksel pada *stego-image* kemudian digeser ke kiri untuk menjadi bit paling penting atau bit ke-1 dan ke-2 sehingga diperoleh *share*. *Share* tersebut didekripsi dengan menggabungkannya dengan *share* lain yang diperoleh dari *stego-image* lain sehingga didapat kembali pesan citra *grayscale*.



Gambar 3.3 Skema Penggabungan Kriptografi Visual *Secret Sharing* dan Steganografi *Enhanced Least Significant Bit*

3.4 Konstruksi Program Aplikasi

Program aplikasi penggabungan kriptografi *Visual Secret Sharing* dan steganografi *Enhanced Least Significant Bit* akan dibuat menggunakan bahasa pemrograman *Python*.

3.4.1 *Input dan Output*

Input untuk proses enkripsi dari program aplikasi ini adalah citra *grayscale* sebagai pesan dan bilangan n dengan $2 \leq n \leq 4$ dan *output* berupa n buah *share*. *Input* untuk proses *embedding* adalah n *share*, citra berwarna sebagai *cover image* dengan *output* adalah *stego-image*. *Input* untuk proses dekripsi dan *extracting* adalah *stego-image* dan n buah *share*.

3.4.2 Algoritma Deskriptif

Algoritma yang digunakan dalam program aplikasi di penelitian ini diuraikan sebagai berikut:

a. Enkripsi dan *Embedding*

Langkah-langkah enkripsi dan *embedding* adalah:

1. Pengirim menentukan citra *grayscale* dan *n* banyak *share*.
2. Pengirim melakukan enkripsi dengan memilih citra *grayscale* dan memasukkan *n* banyak *share* yang sudah ditentukan kemudian diperoleh *n* buah *share*.
3. *Share* yang diperoleh akan disisipkan pada *cover image* yang sebelumnya sudah ditentukan untuk proses *embedding*.
4. Proses *embedding* dilakukan sebanyak *n* kali.
5. Hasil yang didapat adalah *n* buah *stego-image* kemudian dikirim kepada penerima.

b. Dekripsi dan *Extracting*

Langkah-langkah dekripsi dan *extracting* adalah:

1. Penerima melakukan proses *extracting* dengan memilih *stego-image* yang diterima dari pengirim untuk mendapat *share*.
2. Proses *extracting* dilakukan sebanyak *n* kali, di mana *n* adalah banyak *stego-image* yang diterima.
3. *Share-share* hasil *extracting* kemudian didekripsi sehingga didapat pesan awal (citra *grayscale*).

3.4.3 *Library Python*

Berikut beberapa *library python* yang digunakan dalam pembuatan program aplikasi ini:

a. *Tkinter*

Tkinter adalah *library* yang digunakan untuk membuat *Graphic User Interface* (GUI) atau antarmuka grafis, yang mempunyai komponen-komponen pendukung seperti *Button, Textbox, Label, Frame* dan lain-lain.

b. *PIL (Pillow)*

Pillow adalah *library* yang digunakan untuk memanipulasi gambar. Pillow menyediakan fungsi untuk membuka dan menampilkan gambar, merotasi gambar, mengubah ukuran gambar dan lain-lain.

c. *OS*

OS adalah *library* yang menyediakan fungsi untuk melakukan operasi terhadap direktori (folder), file, proses dan informasi lingkungan sistem.

d. *NumPy*

NumPy adalah *library* yang digunakan untuk memudahkan perhitungan *scientific* seperti statistik, matriks dan lain-lain.

e. *Math*

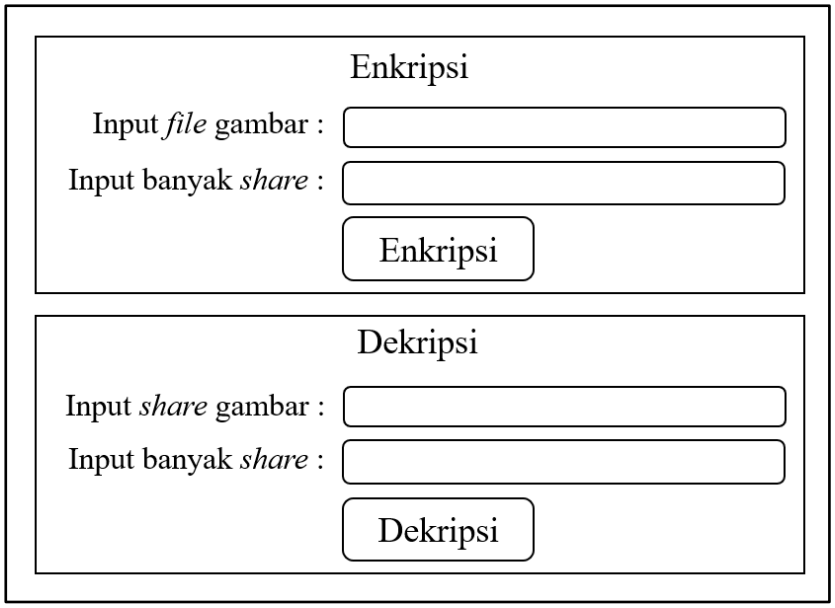
Math adalah *library* yang digunakan untuk perhitungan ilmiah dan matematika yang kompleks seperti operasi modulo, trigonometri, logaritma dan lain-lain.

f. *Open CV*

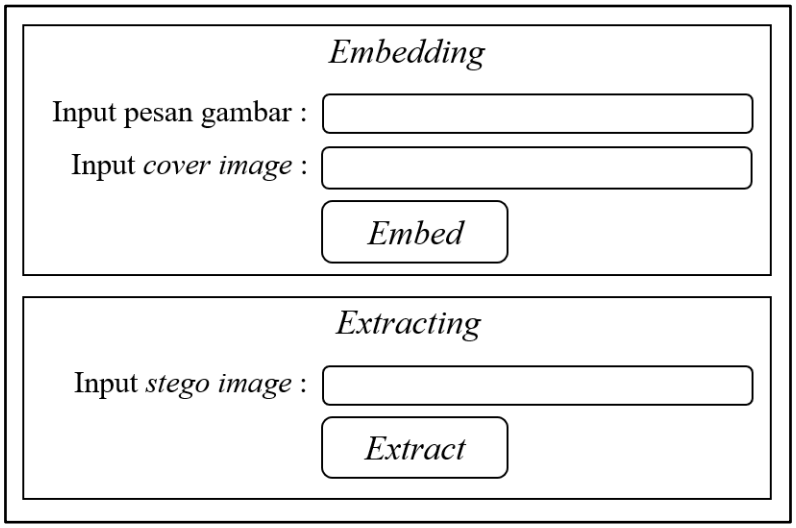
OpenCV (*Open-Source Computer Vision Library*) adalah *library* yang digunakan untuk mengolah gambar.

3.4.4 Rancangan Tampilan Program Aplikasi

Berikut merupakan rancangan tampilan program aplikasi:



Gambar 3.4 Rancangan Tampilan Program Enkripsi dan Dekripsi



Gambar 3.5 Rancangan Tampilan Program *Embedding* dan *Extracting*

3.5 Proses Validasi

Proses validasi dilakukan untuk mengetahui apakah model yang dirancang sudah sesuai atau belum. Program aplikasi tervalidasi jika citra pesan dapat diperoleh kembali setelah melakukan proses *extracting* dan dekripsi. *Stego-image* hasil proses *embedding* akan dilakukan uji *Peak Signal to Noise Ratio* (PSNR) untuk mengetahui kualitas *stego-image* dan juga citra hasil proses dekripsi akan dilakukan uji *Normalized Cross-Correlation* (NCC) untuk mengetahui tingkat kemiripannya dengan citra pesan.

3.6 Pengambilan Kesimpulan

Pengambilan kesimpulan dilakukan sebagai tahapan terakhir berdasarkan hasil yang diperoleh dari penelitian ini, serta pemberian saran untuk penelitian selanjutnya agar memperoleh hasil yang lebih baik.