BAB III
METODOLOGI PENELITIAN

3.1 Identifikasi Masalah

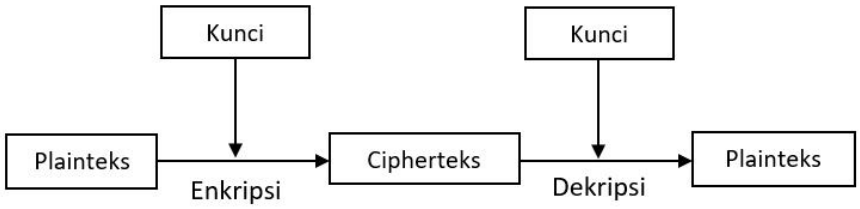
Informasi menjadi suatu hal yang penting dalam era digital modern. Teks digital merupakan salah satu bentuk informasi yang paling sering digunakan dalam proses komunikasi. Pengiriman teks digital yang berisi informasi penting perlu dilengkapi dengan sistem keamanan untuk menjaga kerahasiaannya. Salah satu metode yang dapat digunakan untuk mengamankan teks adalah dengan menerapkan kriptografi. Kriptografi dapat mengubah teks menjadi bentuk terenkripsi sehingga isinya tidak dapat langsung dibaca atau dipahami. Namun, karena teks yang terenkripsi biasanya tampak acak dan tidak bermakna, hal tersebut dapat menimbulkan kecurigaan dari pihak yang menerimanya. Oleh karena itu, diperlukan adanya keamanan tambahan dengan cara menyisipkan teks yang telah terenkripsi tersebut ke gambar lain menggunakan steganografi. Penyandian dan penyisipan pesan dilakukan dengan menggabungkan kriptografi AES-128 dan steganografi PVD.

3.2 Model Dasar

Penelitian ini menggunakan model dasar berupa penerapan algoritma kriptografi AES-128 serta algoritma steganografi PVD.

3.2.1 Model *Advanced Encryption Standard*

Proses enkripsi dan dekripsi dengan algoritma AES-128 yang telah dijelaskan pada BAB II bagian 2.4 akan digambarkan melalui skema pada Gambar 3.1.

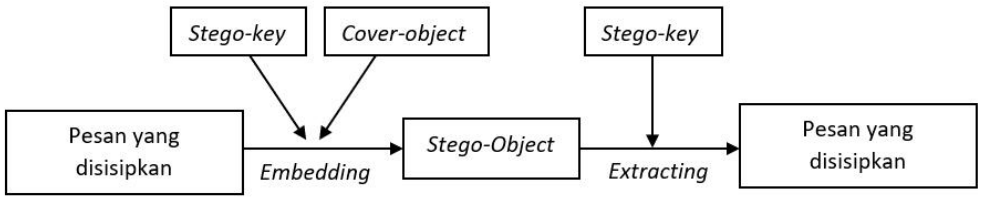


Gambar 3.1 Model Kriptografi *Advanced Encryption Standard*

Pada algoritma kriptografi AES-128, pengirim melakukan proses enkripsi AES-128 dengan kunci dan plainteks yang diinginkan, kemudian cipherteks dan kunci rahasia tersebut dikirimkan kepada penerima untuk didekripsi. Seperti yang telah dijelaskan sebelumnya, proses enkripsi dalam AES-128 melibatkan transformasi data secara berulang menggunakan serangkaian putaran (*rounds*) yang melibatkan substitusi, pergeseran, dan pencampuran bit., yang membuatnya sangat sulit untuk dipecahkan tanpa pengetahuan kunci yang tepat. Selain itu, AES-128 juga menerapkan operasi *bitwise*, yaitu operasi pada level bit (1 dan 0) dalam representasi biner dari data, memungkinkan manipulasi langsung terhadap bit-bit tersebut. Operasi *bitwise* memungkinkan implementasi perangkat keras khusus yang dioptimalkan untuk AES-128, karena perangkat keras dapat di-desain untuk melakukan operasi *bitwise* secara paralel dengan kecepatan tinggi. Namun, salah satu kekhawatiran yang muncul adalah ketika kunci rahasia AES-128 harus ditransmisikan dari satu pihak ke pihak lain secara aman. Jika kunci tersebut jatuh ke tangan yang salah atau disadap oleh pihak yang tidak berwenang, maka keamanan data yang dienkripsi menggunakan kunci tersebut dapat terancam.

3.2.2 Model *Pixel Value Differencing*

Proses penyisipan dan ekstaksi dengan algoritma PVD yang telah dijelaskan pada BAB 2 bagian 2.5 akan digambarkan melalui skema pada Gambar 3.2.



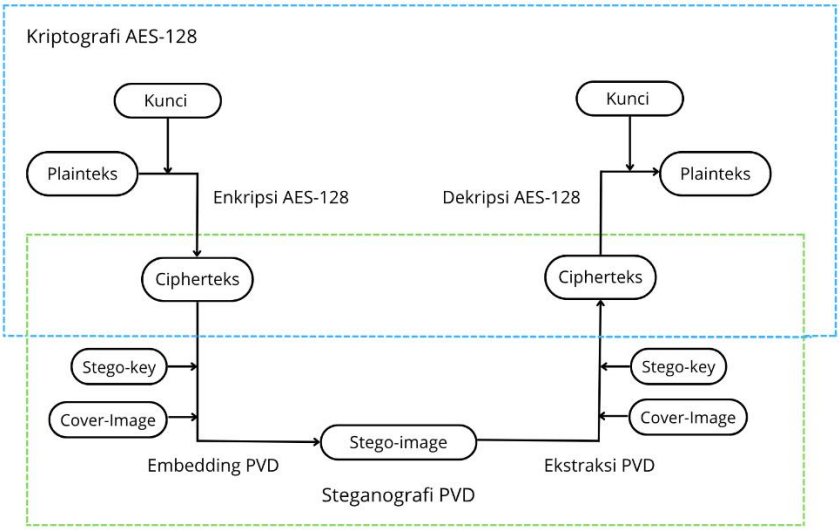
Gambar 3.2 Model Steganografi *Pixel Value Differencing*

Metode PVD merupakan salah satu metode steganografi pada gambar digital yang beroperasi pada ranah spasial. Konsep dari metode ini yaitu dengan menyisipkan pesan kedalam dua pixel yang bertetangga, dengan memanfaatkan perbedaan intensitas warna dari kedua pixel yang bertetangga tersebut. Seperti

halnya metode steganografi lainnya, pada metode ini terdapat dua proses yaitu proses penyisipan dan ekstraksi. Proses penyisipan yaitu proses menyembunyikan informasi kedalam media *cover*, dalam hal ini media *cover* berupa gambar digital. Proses ini akan menghasilkan gambar yang telah disisipkan pesan (*stego-object*) yang menyerupai dengan gambar sebelum disisipkan pesan. Proses ekstraksi yaitu proses pengambilan informasi yang tersembunyi pada gambar digital. Proses ini akan menghasilkan file informasi yang disembunyikan, dengan masukan berupa gambar *stego-object*.

3.3 Pengembangan Model Dasar

Dalam penelitian ini, pengembangan model dilakukan dengan memadukan algoritma dasar yaitu algoritma kriptografi AES-128 dan algoritma steganografi PVD, sehingga keamanan dan kerahasiaan data yang dikirimkan dapat lebih terjamin. Proses diawali ketika pengirim pesan mengenkripsi data berbentuk teks menggunakan algoritma AES-128 dengan kunci rahasia yang dihasilkan penerima pesan, sehingga menghasilkan cipherteks yang selanjutnya akan disembunyikan di dalam media gambar. Proses penyembunyian cipherteks ke dalam gambar dilakukan menggunakan algoritma PVD, sehingga menghasilkan *stego-image*. Ketika *stego-image* diterima, penerima pesan melakukan proses ekstraksi dengan algoritma PVD menggunakan kunci rahasia steganografi yang diberikan pengirim pesan. Setelah memperoleh cipherteks yang tersisipkan, penerima kemudian mendekripsinya menggunakan algoritma AES-128 dengan kunci rahasia yang telah sebelumnya ditentukan, sehingga pesan asli dapat diakses kembali.



Gambar 3.3 Skema pengembangan model

3.4 Konstruksi Program Aplikasi

Program aplikasi yang dikembangkan dalam penelitian ini dibangun menggunakan bahasa pemrograman *Python* dengan detail konstruksi sebagai berikut.

3.4.1 *Input* dan *Output*

1. *Input* dan *output* enkripsi dan dekripsi AES-128

Tabel 3.1 *Input* dan *Output* Enkripsi dan Dekripsi

Keterangan	*Input*	*Output*
Enkripsi	• Plainteks • Kunci	• Cipherteks
Dekripsi	• Cipherteks • Kunci	• Plainteks

2. *Input* dan *output embedding* dan ekstraksi PVD

Tabel 3.2 *Input* dan *Output Embedding* dan Ekstraksi

Keterangan	*Input*	*Output*
Embedding	• Cipherteks • *Cover-image*	• *Stego-image*
Ekstraksi	• *Stego-image*	• Cipherteks

3. *Input* dan *output* pengujian gambar digital

Tabel 3.3 *Input* dan *Output* Uji PSNR

Keterangan	*Input*	*Output*
Uji PSNR	• *Cover-image* • *Stego-image*	• Nilai PSNR

3.4.2 Algoritma Deskriptif

Program aplikasi yang akan dikembangkan menggunakan kombinasi antara algoritma kriptografi AES-128 dan algoritma steganografi PVD, serta pengujian gambar menggunakan pengujian PSNR. Algoritma deskriptif program aplikasi berdasarkan masing-masing algoritma yang digunakan pada penelitian ini adalah sebagai berikut.

1. Algoritma Kriptografi *Advanced Encryption Standard*

a. Algoritma enkripsi

1. *Input* plainteks yang akan dienkripsi.
2. *Input* kunci.
3. Tekan tombol *start* pada program untuk memulai proses enkripsi.
4. Setelah plainteks berhasil dienkripsi, akan terdapat hasil *output* berupa cipherteks.

b. Algoritma dekripsi

1. *Input* cipherteks yang akan didekripsi.
2. *Input* kunci.
3. Tekan tombol *start* pada program untuk memulai proses dekripsi.
4. Setelah plainteks berhasil didekripsi, akan terdapat hasil *output* berupa plainteks.

2. Algoritma Steganografi *Pixel Value Differencing*

a. Algoritma *embedding*

1. Pilih *cover-image* yang akan digunakan untuk menyembunyikan *embedded message*.
2. Masukkan *embedded message*.
3. Tekan tombol *start* yang terdapat pada program aplikasi.
4. Diperoleh *stego-image*.

b. Algoritma Ekstraksi

1. Pilih *stego-image* yang akan digunakan untuk mengeluarkan *embedded message*.
2. Tekan tombol *start* yang terdapat pada program aplikasi.
3. Diperoleh *embedded message*.

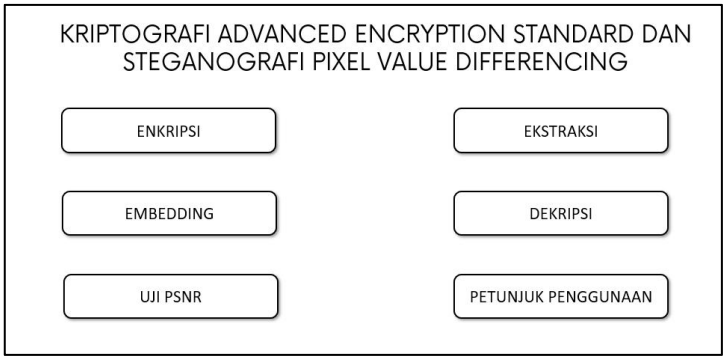
3. Algoritma Uji PSNR

a. Algoritma Uji PSNR

1. Pilih *stego-image* yang sebelumnya digunakan untuk menyembunyikan *embedded message*.
2. Pilih cover-image yang belum digunakan untuk menyembunyikan *embedded message*.
3. Tekan tombol *start* yang terdapat pada program aplikasi.
4. Diperoleh nilai PSNR.

3.4.3 Rancangan Tampilan Program Aplikasi

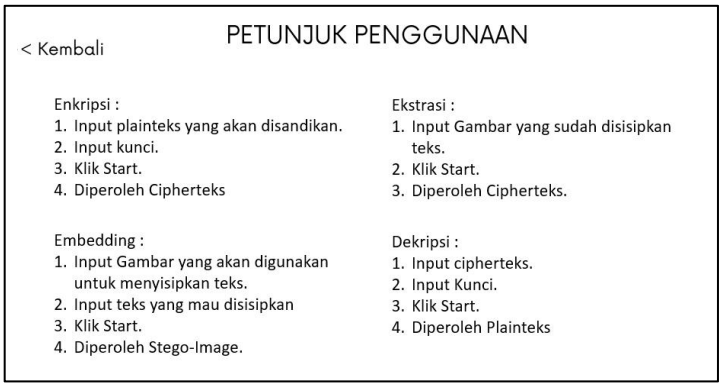
Tampilan program aplikasi pada penelitian ini menggunakan GUI (*Graphical User Interface*) dalam bahasa pemrograman *Python*. Rancangan tampilan pada program aplikasi ini memiliki bagian tampilan awal yang akan ditampilkan pertama kali, yaitu bagian halaman (*page*) yang digunakan untuk menampilkan tombol-tombol dari enkripsi, *embedding*, ekstraksi, dekripsi, uji PSNR atau tata cara, yang apabila dipilih akan berpindah ke halaman tampilan sesuai dengan program yang ditentukan. Rancangan tampilan pada program aplikasi yang akan dikonstruksi dapat dilihat pada Gambar 3.4.



Gambar 3.4 Rancangan Tampilan Awal Program

Terdapat enam halaman pada program aplikasi ini, rancangan tampilan pada setiap halaman tersebut adalah sebagai berikut:

1. Halaman Petunjuk Penggunaan



Gambar 3.5 Rancangan Tampilan Menu Petunjuk Penggunaan

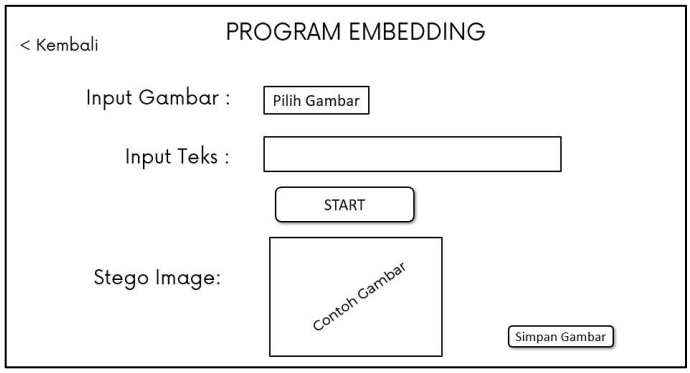
2. Halaman enkripsi

Gambar 3.6 Rancangan Tampilan Menu Program Enkripsi

3. Halaman dekripsi

Gambar 3.7 Rancangan Tampilan Menu Program Dekripsi

4. Halaman *embedding*

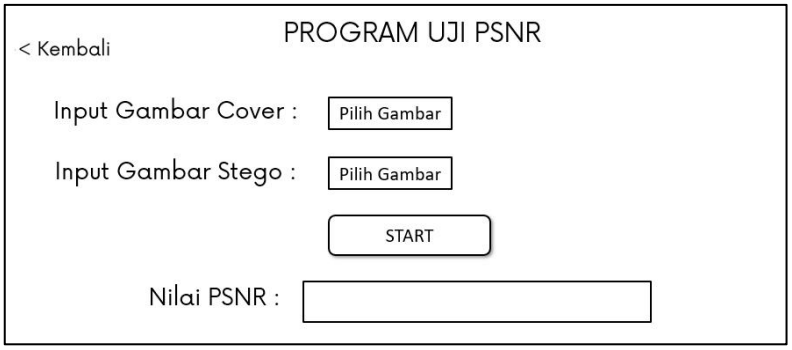


Gambar 3.8 Rancangan Tampilan Menu Program *Embedding*

5. Halaman ekstraksi

Gambar 3.9 Rancangan Tampilan Menu Program Ekstraksi

6. Halaman uji PSNR



Gambar 3.10 Rancangan Tampilan Menu Program Pengujian dengan PSNR

3.4.4 *Library* dan *Module* Program Aplikasi

Program aplikasi pada penelitian ini menggunakan beberapa *library* dan *module* yang terdapat pada bahasa pemrograman *Python*. Library yang digunakan program aplikasi pada penelitian ini adalah sebagai berikut:

1. *Tkinter*, merupakan library pada *Python* yang digunakan untuk menampilkan GUI pada program aplikasi.
2. *Pillow* (PIL), merupakan *library* pada *Python* yang digunakan untuk mengolah gambar digital, seperti menampilkan atau memanipulasi gambar digital.

Module yang digunakan program aplikasi pada penelitian ini adalah sebagai berikut:

1. *Messagebox*, merupakan module pada *library Tkinter* yang digunakan untuk menampilkan kotak dialog yang berisikan informasi
2. *Filedialog*, merupakan module pada *library Tkinter* yang digunakan untuk memilih atau menyimpan suatu file seperti gambar digital.
3. *Image*, merupakan module pada *library* PIL yang digunakan untuk mengolah gambar digital secara sederhana.
4. *ImageTk*, merupakan module pada *library* PIL yang digunakan untuk mengolah gambar digital yang lebih rumit.
5. *Math*, merupakan module yang sudah terinstal bersama dengan *Python* yang digunakan untuk perhitungan ilmiah dan matematika.

3.5 Proses Validasi

Pada tahap ini dilakukan validasi terhadap program aplikasi yang dirancang. Validasi dilakukan untuk mengetahui jika program yang dirancang dapat menghasilkan cipherteks dan menyisipkannya pada gambar digital serta mengembalikan cipherteks yang diperoleh dari ekstraksi gambar digital tersebut menjadi plainteks berdasarkan algoritma AES-128 dan algoritma PVD. *Stego-image* yang dihasilkan dari proses *embedding* akan dilakukan pengujian kualitas gambar menggunakan PSNR dengan membandingkan *stego-image* dengan *cover-image*. Validasi tersebut dilakukan dengan memperlihatkan kesamaan antara

plainteks, *stego-image*, serta cipherteks yang dihasilkan oleh program aplikasi sesuai dengan proses perhitungan manual.

3.6 Penarikan Kesimpulan

Pada bagian ini, kesimpulan akan diambil berdasarkan pengembangan model serta temuan penelitian yang telah dilakukan. Selain itu, akan diberikan sejumlah rekomendasi bagi peneliti berikutnya agar dapat memperoleh hasil penelitian yang lebih optimal. Program aplikasi dianggap valid jika algoritma yang diterapkan mampu melindungi, menyamarkan, dan mengembalikan pesan rahasia yang hendak dikirimkan.