# Implementasi Aplikasi Steganografi Berbasis Desktop Pada Berkas Citra, Audio, dan Video Menggunakan Algoritma LSB Dan BPCS

Yusup Roni S.L\*1, Rihadatul Aisy2, Fredy Enggar Yana3

1,2Teknik Informatika, Universitas Dipa Makassar, Indonesia

E-mail: \*1[Yusuproni346@gmail.com,](mailto:Yusuproni346@gmail.com,) [2Rihadatulaisy@gmail.com](mailto:2Rihadatulaisy@gmail.com), 3Fredyenggar@gmail.com

Abstrak

Steganografi merupakan teknik keamanan informasi yang bertujuan menyembunyikan data rahasia ke dalam media digital sehingga sulit terdeteksi oleh pihak yang tidak berwenang. Penelitian ini membahas implementasi aplikasi steganografi berbasis desktop untuk berkas citra, audio, dan video menggunakan algoritma Least Significant Bit (LSB) dan Bit-Plane Complexity Segmentation (BPCS). Algoritma LSB menyisipkan data rahasia dengan memanfaatkan bit paling tidak signifikan dari media penampung (cover media), sedangkan algoritma BPCS meningkatkan kapasitas penyisipan data dengan memanfaatkan analisis kompleksitas bit-plane.

Aplikasi ini dirancang untuk mendukung berbagai jenis media digital, seperti citra, audio, dan video, dengan memadukan keunggulan kedua algoritma. Hasil pengujian menunjukkan bahwa algoritma LSB memiliki kecepatan proses yang lebih tinggi dengan pengaruh minimal pada kualitas media penampung. Sebaliknya, algoritma BPCS menawarkan kapasitas penyisipan yang lebih besar, terutama untuk media dengan resolusi atau ukuran yang tinggi. Pengujian pada berbagai format berkas menunjukkan bahwa aplikasi ini berhasil menyisipkan dan mengekstrak data dengan tingkat akurasi yang tinggi, tanpa kerusakan signifikan pada media penampung.

Aplikasi ini diharapkan dapat menjadi solusi untuk kebutuhan komunikasi data yang aman di era digital, khususnya untuk menjaga kerahasiaan informasi. Pengembangan lebih lanjut dapat mencakup integrasi metode enkripsi untuk meningkatkan keamanan dan fleksibilitas aplikasi.

**Kata Kunci**: steganografi, algoritma LSB, algoritma BPCS, citra, audio, video, aplikasi desktop.

Abstract

*Steganography is an information security technique that aims to hide confidential data in digital media so that it is difficult for unauthorized parties to detect it. This research discusses the implementation of desktop-based steganography applications for image, audio and video files using the Least Significant Bit (LSB) and Bit-Plane Complexity Segmentation (BPCS) algorithms. The LSB algorithm inserts secret data by utilizing the least significant bit of the cover media, while the BPCS algorithm increases data insertion capacity by utilizing bit-plane complexity analysis.*

*This application is designed to support various types of digital media, such as images, audio and video, by combining the advantages of both algorithms. The test results show that the LSB algorithm has a higher processing speed with minimal impact on the quality of the storage media. In contrast, the BPCS algorithm offers greater embedding capacity, especially for media with high resolution or size. Tests on various file formats show that this application successfully inserts and extracts data with a high level of accuracy, without significant damage to the storage media.*

*It is hoped that this application can be a solution for the need for secure data communications in the digital era, especially for maintaining the confidentiality of information. Further development may include integration of encryption methods to increase application security and flexibility.*

***Keywords:*** *steganography, LSB algorithm, BPCS algorithm, image, audio, video, desktop application*

## **Pendahuluan**

Steganografi adalah salah satu metode dalam bidang keamanan informasi yang memungkinkan penyembunyian data rahasia ke dalam media digital sehingga keberadaannya sulit terdeteksi oleh pihak yang tidak berwenang. Dalam konteks perlindungan data, steganografi menawarkan pendekatan yang berbeda dibandingkan dengan kriptografi, yang hanya menyamarkan isi pesan. Dengan steganografi, keberadaan pesan itu sendiri tidak mudah dikenali, sehingga meningkatkan lapisan keamanan dalam komunikasi digital [1][2].

Latar belakang dari penelitian ini adalah meningkatnya kebutuhan akan teknologi perlindungan data yang dapat diimplementasikan pada berbagai jenis media digital, seperti citra, audio, dan video. Dengan berkembangnya teknologi digital, media-media tersebut menjadi sarana yang umum digunakan untuk menyimpan dan mentransmisikan informasi. Namun, kerentanan terhadap pencurian atau penyadapan data juga semakin tinggi, sehingga diperlukan solusi yang efektif untuk menjaga kerahasiaan informasi [3].

Motivasi utama dalam pemilihan topik ini adalah untuk menjawab tantangan dalam menyediakan aplikasi steganografi yang fleksibel, efisien, dan mampu menangani berbagai format media digital. Dengan menggunakan algoritma Least Significant Bit (LSB) dan Bit-Plane Complexity Segmentation (BPCS), aplikasi ini diharapkan mampu memberikan kapasitas penyisipan data yang tinggi tanpa mengurangi kualitas media penampung secara signifikan [4]. Pengembangan aplikasi berbasis desktop dipilih untuk memastikan kemudahan akses dan penggunaan oleh pengguna dari berbagai kalangan.

## **LANDASAN TEORI**

Steganografi berasal dari bahasa Yunani, yaitu “steganos” yang berarti tersembunyi dan “graphia” yang berarti tulisan. Dalam konteks teknologi informasi, steganografi adalah teknik menyembunyikan data ke dalam media penampung seperti citra, audio, dan video tanpa mengubah struktur visual atau audial secara signifikan [5].

Steganografi berbeda dengan kriptografi yang hanya mengamankan isi data; steganografi menyembunyikan keberadaan data itu sendiri. Data rahasia disisipkan ke dalam media penampung dengan cara yang sedemikian rupa sehingga tidak terlihat atau terdengar oleh orang lain. Salah satu aspek penting dalam steganografi adalah kapasitas penyisipan (payload), ketahanan terhadap deteksi (robustness), dan kualitas media penampung setelah data disisipkan [6].

Metode yang umum digunakan dalam steganografi adalah Least Significant Bit (LSB) dan Bit-Plane Complexity Segmentation (BPCS).**Least Significant Bit (LSB)** Metode ini bekerja dengan menggantikan bit-bit paling tidak signifikan pada media penampung dengan bit-bit data rahasia. Teknik ini memiliki keunggulan berupa kemudahan implementasi dan tidak memengaruhi kualitas media secara signifikan. Namun, metode ini kurang tahan terhadap serangan steganalisis seperti kompresi atau pengubahan format [7].**Bit-Plane Complexity Segmentation (BPCS)** BPCS adalah teknik steganografi yang memanfaatkan kompleksitas bit-plane dalam media digital. Area dengan kompleksitas tinggi digunakan untuk menyisipkan data karena perubahan di area tersebut sulit dideteksi oleh mata manusia. Metode ini memiliki kapasitas penyisipan yang lebih besar dibandingkan dengan LSB [8].

## **METODE IMPLEMENTASI**

**3.1 Alat dan Teknologi Pengembangan**

1. Visual Studio Code sebagai Integrated Development Environment (IDE).

2. Python versi 3.x sebagai bahasa pemrograman utama.

3. Library Python yang digunakan:

a. PIL (Python Imaging Library) untuk pemrosesan citra.

b. Wave untuk pemrosesan berkas audio.

c. OpenCV untuk pemrosesan berkas video.

d. Tkinter untuk pengembangan antarmuka grafis.

e. NumPy untuk operasi matematika dan manipulasi array.

**3.2 Metode Penelitian**

1. Metode Pengumpulan Data

a. Studi Literatur: Mengkaji jurnal dan publikasi terkait steganografi.

b. Dokumentasi: Mencatat hasil pengujian dan analisis.

2. Metode Pengembangan Sistem

a. Analisis Kebutuhan: Identifikasi kebutuhan fungsional dan non-fungsional.

b. Perancangan: Desain arsitektur dan antarmuka aplikasi.

c. Implementasi: Pengkodean dan integrasi sistem.

**3.3 Implementasi Algoritma**

1. Algoritma LSB (Least Significant Bit)

a. Pengembangan fungsi untuk membaca bit-bit dari berkas sumber.

b. Implementasi proses penyisipan bit data ke dalam bit LSB media penampung.

c. Pengembangan fungsi untuk mengekstrak data tersembunyi.

d. Optimasi performa menggunakan NumPy untuk operasi array.

2. Algoritma BPCS (Bit-Plane Complexity Segmentation)

a. Implementasi pemisahan bit-plane menggunakan operasi bitwise

b. Pengembangan fungsi analisis kompleksitas untuk setiap segmen

c. Implementasi penyisipan data pada area dengan kompleksitas tinggi

d. Pengembangan mekanisme penandaan area yang digunakan untuk penyisipan

**3.4 Dataset Pengujian**

1. Berkas citra: PNG dan JPG.

2. Berkas audio: WAV.

3. Berkas video: MP4.

4. Data yang akan disisipkan: file dokume.

## **Hasil dan pembahasan**

Penelitian ini menggunakan file gambar sebagai media utama (citra cover) untuk menyembunyikan teks pesan dengan menerapkan teknik enkripsi dan steganografi. Pesan teks akan dienkripsi terlebih dahulu, kemudian digabungkan ke dalam file gambar menggunakan algoritma Least Significant Bit (LSB) dan Bit-Plane Complexity Segmentation (BPCS).

Metode steganografi yang digunakan dalam penelitian ini melibatkan algoritma LSB untuk menyembunyikan data. Proses awal dimulai dengan menyisipkan data tersembunyi ke dalam stegomedium (media penyimpanan), menggunakan kunci tertentu. Hasil akhirnya adalah stegoimage, yakni media gambar yang telah mengandung data tersembunyi [9]. Berikut adalah langkah-langkah yang dilakukan:

1. Mengonversi citra gambar menjadi representasi biner.
2. Menentukan pesan teks (plaintext) yang akan disisipkan.
3. Menentukan dan memilih kunci (key) untuk proses enkripsi.
4. Mengenkripsi pesan dan kunci dengan algoritma Vigenère Cipher.
5. Mengubah hasil enkripsi (ciphertext) menjadi format biner.
6. Menyisipkan biner dari ciphertext ke dalam biner gambar menggunakan metode LSB, di mana setiap bit dari ciphertext ditempatkan pada bit terakhir dari setiap piksel gambar.
7. Mengonversi kembali hasil penyisipan biner tersebut menjadi format gambar.

Proses penyandian atau embedding dalam kedua metode ini bertujuan untuk menyisipkan bit-bit data pesan ke dalam gambar. Dalam algoritma LSB, data disisipkan secara langsung pada bit terakhir setiap piksel gambar. Sebaliknya, algoritma BPCS menyisipkan data ke dalam blok-blok gambar yang memiliki tingkat kompleksitas bit-plane yang tinggi.

Pada akhir proses penyandian (*encode*), file citra akan menjadi citra terenkripsi yang berisi teks pesan tersembunyi. Proses ini harus dilakukan dengan cermat untuk memastikan kecepatan, keamanan, dan integritas data tetap terjaga. Setelah itu, proses *decode* atau ekstraksi dilakukan untuk mengambil kembali data yang disisipkan ke dalam file citra.

Proses ekstraksi data dari *stegoimage* dilakukan menggunakan metode dan kunci yang sama dengan proses penyisipan sebelumnya. Hal ini memungkinkan data tersembunyi untuk diambil kembali [10]. Langkah-langkah dalam proses ekstraksi adalah sebagai berikut:

1. Memilih dan menentukan file citra yang mengandung data tersembunyi (*stegoimage*).
2. Mengonversi file *stegoimage* ke dalam bentuk biner.
3. Memisahkan biner *ciphertext* dari biner citra.
4. Melakukan dekripsi untuk memisahkan pesan dan kunci, dengan cara memasukkan kunci yang digunakan saat penyisipan, kemudian menjalankan proses dekripsi.
5. Mengonversi biner citra menjadi desimal untuk dipetakan kembali ke dalam bentuk citra asli.
6. Mengubah biner dari teks dan kunci ke dalam format desimal, lalu mengonversinya menjadi karakter atau teks asli.

Algoritma Bit-Plane Complexity Segmentation (BPCS) menggunakan pendekatan Canonical Gray Code (CGC) untuk proses penyisipan pesan. Dalam metode ini, representasi bit-plane dari sistem PBCM (Pure Binary Code Map) diubah menjadi CGC sebelum proses penyisipan dimulai. Data pesan disisipkan pada segmen-segmen dengan tingkat kompleksitas tinggi, yang dikenal sebagai wilayah noise-like. Penyisipan dilakukan pada seluruh bit-plane yang memenuhi kriteria kompleksitas tinggi. Dengan teknik BPCS, kapasitas penyisipan data dapat mencapai hingga 50% dari ukuran objek penutup (cover object) [11].

Langkah-langkah Penyisipan Data dengan Algoritma BPCS:

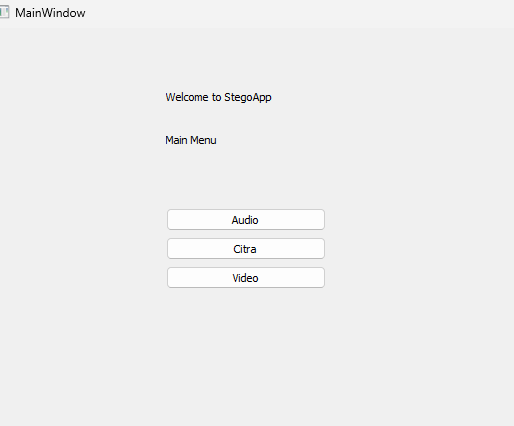
1. Konversi ke CGC: Objek penutup yang menggunakan sistem PBC dikonversi ke CGC. Gambar kemudian dipecah menjadi bit-plane berbentuk gambar biner, di mana setiap piksel gambar diwakili oleh bit-plane.
2. Pembagian Wilayah Berdasarkan Kompleksitas: Dengan menggunakan nilai ambang batas (α0), setiap bit-plane dari objek penutup dibagi menjadi dua wilayah: informatif dan noise-like. Nilai ambang batas standar yang digunakan adalah 0,3.
3. Pengelompokan Pesan Rahasia: Pesan rahasia disusun menjadi rangkaian blok data yang terdiri dari beberapa byte.
4. Penyesuaian Kompleksitas Blok: Jika sebuah blok pesan rahasia (S) kurang kompleks dibandingkan nilai ambang batas, maka dilakukan konjugasi terhadap blok tersebut sehingga menghasilkan blok konjugasi (S\*) dengan kompleksitas yang lebih tinggi. Blok pesan rahasia harus memiliki kompleksitas yang memenuhi atau melebihi nilai ambang batas.
5. Penyisipan Pesan ke Wilayah Noise-like: Blok pesan rahasia disisipkan ke dalam bit-plane yang memenuhi kriteria wilayah noise-like, atau menggantikan bit-bit pada wilayah tersebut.

Proses Decode:

Pada algoritma LSB, proses decode melibatkan pengambilan bit terakhir dari setiap piksel untuk merekonstruksi urutan bit pesan yang tersembunyi. Dalam algoritma BPCS, data yang disisipkan diekstraksi dengan memanfaatkan informasi kompleksitas wilayah dan nilai ambang batas yang digunakan pada saat penyisipan.Hasil implementasi algoritma ini menunjukkan bahwa teknik BPCS dan LSB efektif untuk menyembunyikan pesan dalam gambar atau citra digital menggunakan metode steganografi. Kedua algoritma tersebut juga dapat digunakan untuk mengamankan data berbasis file melalui platform berbasis desktop.Namun, implementasi ini juga memiliki beberapa keterbatasan. Aplikasi tidak dapat mengembalikan pesan ke bentuk semula pada media video, menyoroti perlunya pengembangan lebih lanjut.

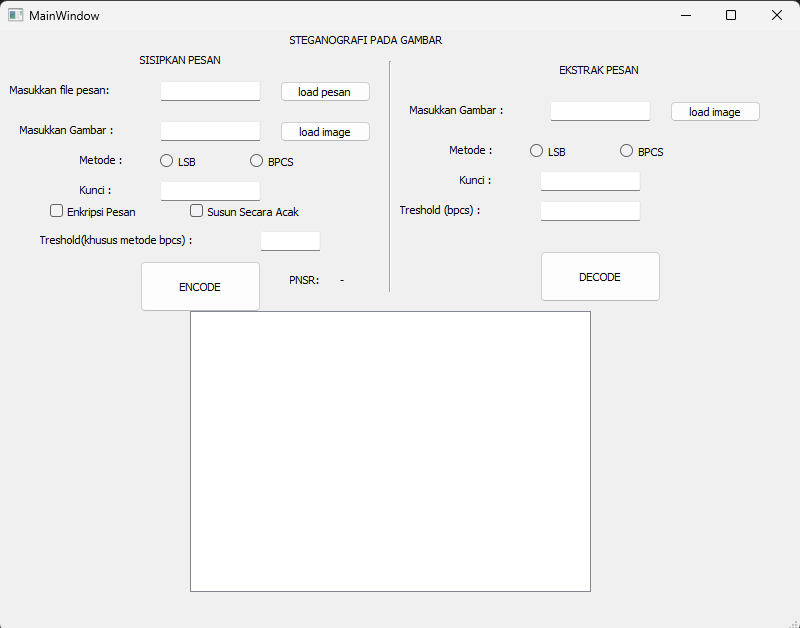
**4.1 Desain Antarmuka**

1. Halaman utama



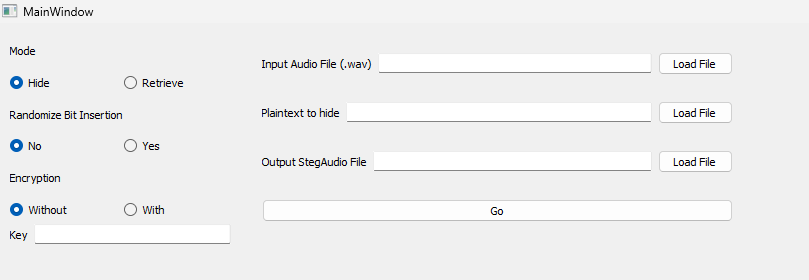
**Gambar 1.** Halaman Utama

2. Halaman Penyisipan Pesan Pada Gambar

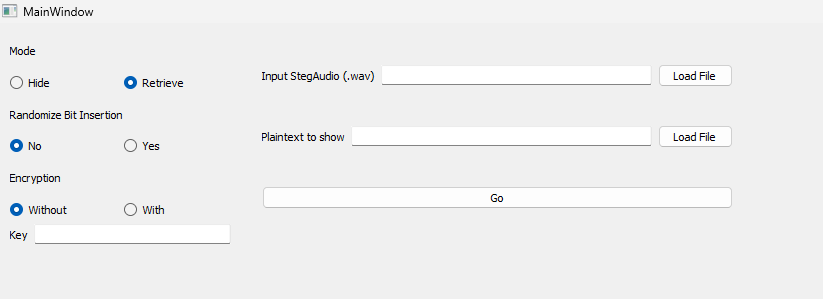


**Gambar 2.** Halaman Encode dan Decode

3. Halaman Penyisipan Pesan Pada Audio

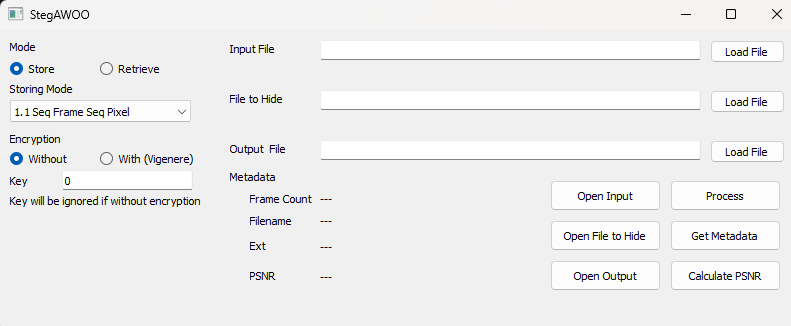


**Gambar 3.** Halaman Hide



**Gambar 4.** Halaman Retrieve

3. Halaman Penyisipan Pesan Pada Video



**Gambar 5.** Halaman Stego Video

## **Kesimpulan**

Penelitian ini telah berhasil mengimplementasikan algoritma LSB dan BPCS pada aplikasi steganografi berbasis desktop untuk media gambar, audio, dan video. Algoritma LSB digunakan pada semua jenis media, termasuk video, sementara algoritma BPCS hanya diterapkan pada media gambar dan audio. Proses penyisipan pesan dilakukan dengan efisiensi tinggi, terutama untuk media gambar dan audio, tanpa memengaruhi kualitas media secara signifikan. Aplikasi ini juga telah mendukung keamanan tambahan melalui enkripsi pesan menggunakan algoritma Vigenere Cipher, yang memastikan pesan tersembunyi tetap terjaga dari akses tidak sah.

Namun, terdapat beberapa keterbatasan yang perlu diperhatikan. Salah satu kelemahan utama adalah ketidakmampuan aplikasi untuk mengembalikan pesan ke bentuk semula pada media video. Selain itu, kapasitas penyisipan pesan sangat bergantung pada ukuran dan karakteristik media, serta performa aplikasi yang cenderung menurun pada file berukuran besar. Dukungan terhadap format media tertentu juga masih memerlukan pengujian lebih lanjut untuk memastikan kompatibilitas yang lebih luas.

Meskipun memiliki beberapa keterbatasan, aplikasi ini menunjukkan potensi besar untuk diterapkan dalam kebutuhan penyembunyian pesan secara digital. Pengembangan lebih lanjut, seperti optimalisasi algoritma LSB untuk media video dan penerapan teknik kompresi pesan, dapat menjadi arah penelitian di masa depan guna mengatasi kekurangan dan meningkatkan efektivitas aplikasi.

## **Daftar Pustaka**

1. Hartati, S., & Darmawan, D. (2018). Implementasi Steganografi pada Citra Digital Menggunakan Metode Least Significant Bit. Jurnal Informatika, 14(1), 15-24.
2. Putra, A. K., & Yuwono, T. (2019). Studi Implementasi Algoritma BPCS untuk Keamanan Informasi. Jurnal Teknologi Informasi dan Komunikasi, 11(2), 35-42.
3. Lestari, I., & Hidayat, R. (2020). Penggunaan Metode Steganografi LSB untuk Penyisipan Pesan pada Media Digital. Jurnal Ilmu Komputer, 16(3), 45-52.
4. Ramadhan, R., & Sari, M. (2021). Analisis Kapasitas Penyisipan Data pada Video Menggunakan Algoritma BPCS. Jurnal Sistem Informasi, 18(2), 67-74.
5. Suryadi, A. (2020). Konsep dan Implementasi Steganografi dalam Keamanan Data Digital. Jurnal Teknologi Informasi, 12(2), 23-30.
6. Wirawan, F., & Nugraha, B. (2019). Studi Literatur: Perbandingan Algoritma LSB dan BPCS dalam Steganografi. Jurnal Keamanan Informasi, 8(1), 34-41.
7. Pratama, R., & Dewi, T. (2021). Optimasi Kapasitas Penyisipan Data pada Media Digital dengan Metode LSB. Jurnal Informatika dan Rekayasa Perangkat Lunak, 15(4), 12-18.
8. Nugroho, P. T., & Widodo, S. (2022). Optimasi Algoritma Steganografi pada Media Audio untuk Keamanan Data. Jurnal Teknik Informatika, 10(1), 12-20.
9. N.F. Hasan, C. N. Dengen, dan D. Ariyus, “Analisis Histogram Steganografi Least Significant Bit Pada Citra Grayscale,” *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 11, pp. 20–29, 2020, doi: 10.31849/digitalzone.v11i1.3413.
10. B. W. Rauf, “Kombinasi Steganografi Bit Matching dan Kriptografi Playfair Cipher, Hill Cipher dan Blowfish,” *J. Teknol. Inf*, vol. 4, pp. 282–233, 2020, doi: 10.36294/jurti.v4i2.1346.
11. Nugroho, Adi, “Rekayasa Perangkat Lunak Berbasis Objek Dengan Metode USDP,” *Penerbit Andi*.