

**APLIKASI STEGANOGRAFI CITRA DIGITAL
MENGUNAKAN METODE LEAST SIGNIFICANT BIT (LSB)**



Disusun oleh:

MUHAMMAD ARI SAPUTRA

011200042

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
ISTITUT TEKNOLOGI DAN BISNIS PALCOMTECH
PALEMBANG**

A. Pendahuluan

Aplikasi ini bertujuan untuk mengembangkan suatu sistem pengolahan citra digital yang mampu menyembunyikan data atau informasi rahasia ke dalam gambar tanpa mengurangi kualitas visual dari gambar tersebut. Teknik yang akan digunakan dalam proyek ini adalah steganografi, yaitu seni atau teknik menyembunyikan data atau pesan rahasia di dalam suatu medium, seperti gambar, audio, atau video, tanpa menarik perhatian pihak ketiga yang melihat medium tersebut. Dalam proyek ini, kita akan menggunakan metode steganografi berbasis Least Significant Bit (LSB) untuk menyembunyikan data ke dalam gambar.

B. Teori Dasar

a) Pengolahan Citra Digital

Pengolahan citra digital adalah proses memanipulasi citra digital menggunakan algoritma komputer untuk meningkatkan kualitas visual, menganalisis citra, dan menghasilkan informasi yang relevan dari citra tersebut.

b) Steganografi

Steganografi adalah seni atau teknik menyembunyikan data atau pesan rahasia di dalam suatu medium, seperti gambar, audio, atau video, tanpa menarik perhatian pihak ketiga yang melihat medium tersebut.

c) Teknik LSB Steganografi

Teknik LSB steganografi adalah salah satu metode sederhana untuk menyembunyikan data dalam gambar. Pada dasarnya, setiap piksel dalam gambar diwakili oleh beberapa bit. Dalam gambar RGB, misalnya, setiap piksel diwakili oleh 24 bit (8 bit untuk setiap channel warna). Teknik ini menggunakan bit paling tidak signifikan (LSB) dari setiap channel warna (red, green, dan blue) untuk menyimpan bit-data yang ingin disembunyikan. LSB dipilih karena perubahan nilai bit-nya cenderung tidak terlihat secara visual.

C. Rancangan Aplikasi

a) Baca Gambar

Pada tahap ini, aplikasi akan membaca gambar yang akan digunakan sebagai media untuk menyembunyikan data.

b) Konversi Data Menjadi Bit

Data atau informasi rahasia yang ingin disembunyikan akan diubah menjadi bentuk bit. Misalnya, jika kita ingin menyembunyikan sebuah teks, teks tersebut akan diubah menjadi representasi bit menggunakan ASCII atau UTF-8.

c) Penyisipan Data

Dalam langkah ini, data yang sudah diubah menjadi bit akan disisipkan ke dalam gambar menggunakan teknik LSB steganografi. Bit terakhir dari setiap channel warna (LSB) akan diganti dengan bit-data yang ingin disimpan.

d) Simpan Gambar Hasil

Gambar yang telah dimodifikasi dengan data tersembunyi akan disimpan sebagai citra hasil.

e) Ekstaksi Data

Pada tahap ini, aplikasi akan mengambil gambar hasil yang berisi data tersembunyi dan mengekstrak kembali data tersebut.

D. Implementasi dan Penjelasan

Berikut adalah implementasi kode untuk aplikasi pengolahan citra digital yang menyembunyikan data menggunakan teknik LSB steganografi:

Impor pustaka yang diperlukan

```
import cv2
```

```
import numpy as np
```

Fungsi untuk mengubah data menjadi bit

```
def string_to_bits(message):
```

```
    binary_message = ''.join(format(ord(char), '08b') for char in message)
```

```
    return binary_message
```

Fungsi untuk menyembunyikan bit-data ke dalam gambar

```
def hide_data(image, data):
```

```
    data_size = len(data)
```

Ukuran gambar (lebar x tinggi)

```
    image_height, image_width, _ = image.shape
```

```
    image_size = image_height * image_width
```

```

if data_size > image_size:

    raise ValueError("Ukuran data terlalu besar untuk gambar ini.")

    data_index = 0

    for row in range(image_height):

        for col in range(image_width):

            pixel = image[row, col]

            for color_channel in range(3): # 3 untuk citra berwarna (RGB)

                if data_index < data_size:

                    # Ambil bit dari data

                    data_bit = int(data[data_index])

                    # Ganti bit LSB dengan bit-data

                    pixel[color_channel] =

                        (pixel[color_channel] & 254) | data_bit

                    data_index += 1

                else:

                    break

    return image

```

Fungsi untuk mengekstrak data dari gambar hasil

```

def extract_data(image, data_size):

    extracted_data = ""

    data_index = 0

    image_height, image_width, _ = image.shape

    for row in range(image_height):

```

```

        for col in range(image_width):

            pixel = image[row, col]

            for color_channel in range(3): # 3 untuk citra berwarna (RGB)

                # Ambil bit-data dari bit LSB

                extracted_data += str(pixel[color_channel] & 1)

                data_index += 1

            if data_index == data_size:

                return extracted_data

# Contoh Penggunaan

if __name__ == "__main__":

    # Baca gambar

    image_path = "gambar_input.jpg"

    image = cv2.imread(image_path)

    # Data yang akan disembunyikan

    data_to_hide = "Ini adalah pesan rahasia!"

    # Ubah data menjadi bit

    hidden_data = string_to_bits(data_to_hide)

    #Sembunyikan data ke dalam gambar

    image_with_hidden_data = hide_data(image.copy(), hidden_data)

    #Simpan gambar hasil

    output_image_path = "gambar_output.jpg"

    cv2.imwrite(output_image_path, image_with_hidden_data)

    # Ekstraksi data dari gambar hasil

```

```
extracted_data = extract_data(image_with_hidden_data, len(hidden_data))  
  
print("Data tersembunyi yang diekstrak:", extracted_data)
```

E. Kesimpulan

Dengan menggunakan teknik steganografi berbasis LSB, aplikasi ini mampu menyembunyikan data atau informasi rahasia dalam gambar tanpa mengurangi kualitas visual dari gambar tersebut. Aplikasi ini memiliki berbagai manfaat, seperti keamanan informasi, komunikasi rahasia, perlindungan privasi, serta dapat digunakan sebagai alat pembelajaran dan penelitian dalam bidang pengolahan citra digital dan steganografi. Meskipun teknik ini sederhana, teknik steganografi memiliki banyak variasi dan metode yang lebih canggih untuk meningkatkan keamanan dan daya guna aplikasi ini.