Concealing Intelligence in images using LSB

A PROJECT REPORT

Submitted by

Mohammad Hashir Khan[RA2111003030319]
Hanshu Agrahari[RA2111003030302]
Ayush Sharma[RA2111003030295]
Ankit Tiwari[RA2111003030290]

Under the guidance of

Ms. Swati Sheoran

(Professor, Department of Computer Science & Engineering)

In partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)
DELHI-NCR CAMPUS, GHAZIABAD (U.P)

SRM INSTITUTE OF SCIENCE & TECHNOLOGY, NCR CAMPUS

NOVEMBER 2024

SRM INSTITUTE OF SCIENCE &TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled "**Concealing Intelligence in Images using LSB**" is the bonafide work of "**Mohd Hashir Khan [Reg No:RA2111003030319], Hanshu Agrahari [Reg No:RA2111003030302], Ayush Sharma [Reg No:RA2111003030295], Ankit Tiwar[RA2111003030290]** "who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project re- port or dissertation on the basis of which a degree or award was conferredon an earlier occasion on this or any other candidate.

SIGNATURE

Ms.Swati Sheoran
GUIDE
Professor
Dept. of Computer Science & Engineering

SIGNATURE

Dr.Avneesh Vashishta
HEAD OF THE DEPARTMENT
Dept. of Computer Science & Engineering

Signature of the Internal Examiner

Signature of the External Examiner

ABSTRACT

Steganography is the art and science of hiding information within another medium, ensuring that the hidden data remains undetected. Derived from the Greek words "steganos" (meaning covered) and "graphy" (meaning writing), it is a technique that has been used for centuries to conceal messages from prying eyes. In modern digital contexts, steganography is widely applied to secure information, embed watermarks in digital media, and protect intellectual property. Unlike cryptography, which makes data unreadable without a decryption key, steganography aims to keep the existence of information undisclosed. The core purpose of this report is to delve into the methods, applications, and technical considerations involved in steganography. We also cover related system designs, coding techniques, and the testing processes critical to developing robust steganographic systems. In addition, this report examines the security and performance implications of steganography and explores potential future enhancements in this domain. By the end, readers should have a clear understanding of steganography’s practical implementations and its significance in modern data security.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my guide, **Ms. Swati Sheoran** for his valuable guidance, consistent encouragement, personal caring, timely help and providing me with an excellent atmosphere for doing research. All through the work, in spite of his busy schedule, he has extended cheerful and cordial support to me for completing this research work.

Mohammad Hashir Khan[RA2111003030319]
Hanshu Agrahari[RA2111003030302]
Ayush Sharma[RA2111003030295]
Ankit Tiwari[RA2111003030290]

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 General Introduction

Information security can be summed up to info, a group of steps, procedures, and strategies that are used to stop and observe illegal access, trouble- shooting, revelation, and adjustment of computer network sources. Enhancing the privacy, eligibility and reliability of the work requires a lot work to strengthen the current methods from constant trials to break them and to improve new ways that are resistant to most kinds of attacks if not all. As a result of the continuously growing number of internet users, complexities have surfaced about the mechanisms to securely store and/or transmit such enormous user data, facing more challenges in terms of data storage and transmission over the internet, for example
information like Account number, Bank Verification Number (BVN), password etc. Hence, in order to provide a better security mechanism, this project work proposed a data hiding technique called steganography along with the technique of encryption-decryption.

Steganography is the art and science of hiding data into different carrier files such as text, audio, images, video, etc. The need to send a message as securely and as safely as possible has been the point of discussion since long time. Information is the assets of any association. This makes security-issues main concern to an association dealing with secret information. Whatever is the process we select for the security point, the strong concern is the level of security. Steganography is the ability of covered or hidden writing.
The word steganography is derived from the Greek words **"stegos"** meaning **"cover"** and **"grafia"** meaning **"writing"**.

The art of hiding a message, image, or file within another message, image, file or video is known as Steganography. Steganography is used to reveal the information which is hidden in an audio or video file. To control the hiding process a stego-key is used so as to limit the detection or recovery of fixed data year, several suspected that Osama Bin Laden may have been posting images on eBay with hidden messages inside to send to different terrorist groups.

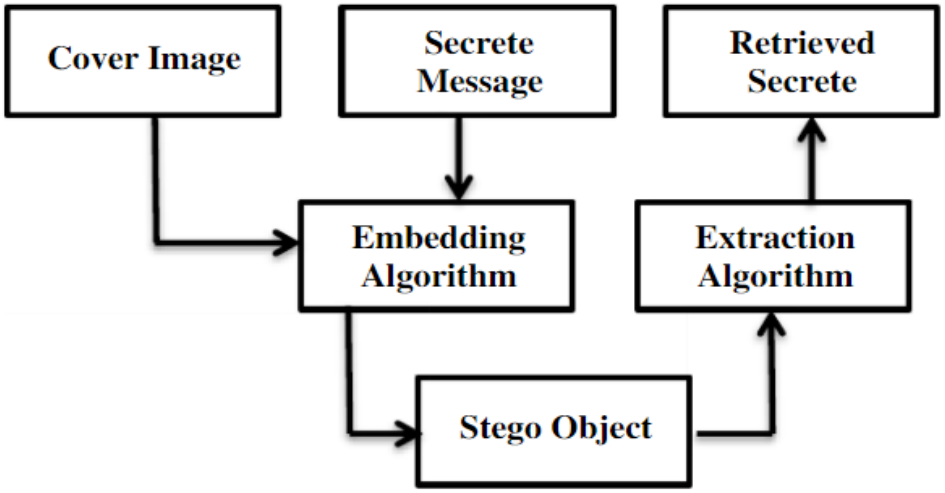


Fig. 1.1 Basic Steganography Model

The basic model of steganography consists of cover object, message, embedding algorithm and Stego key. The model for steganography is shown in Figure 1.

Cover object is also known as a carrier, which hides the message and serves to hide its presence. Digital images, videos, sound files, and other PC files that hold perceptually irrelevant or redundant information can be used as ―covers‖ or carriers to hide secret data.

After embedding a secret data into the cover-object, a so-called stego-object is obtained. At the receiver end, by applying an extraction algorithm we can retrieved the secrete message from stego object. In this project work using the technique of hiding the data with an image file, the visibility of the image, resolution or clarity is not being affected. The hidden data can be of length in size. To the hacker, only, the image is going to be visible when previewed and not a trace of the hidden data. If the image file is opened across a text editor, then also the data is not going to be visible as the information is stored in an encryption form, which is also binary, hence making it difficult for the enclosure to differentiate the data to the image file.

1.2 Problem Statement

In a world increasingly reliant on digital communication, the protection of sensitive information has become a critical challenge. Traditional encryption methods, while effective, often attract attention and may be vulnerable to sophisticated attacks. Steganography, the art of concealing information within other seemingly innocuous data, offers a complementary approach by hiding the existence of the communication itself. This project aims to explore, develop, and implement steganographic techniques to enhance the security and confidentiality of sensitive information.

The Least Significant Bit (LSB) method of steganography offers a unique approach to hiding

data by altering the least significant bits of pixel values in an image. This technique ensures that the changes made to the image are imperceptible to the human eye, thereby maintaining the visual integrity of the image while securely embedding the data.

1.2 AIM AND OBJECTIVES

The aim of this project work is to develop a system that will implement steganography using image which serve as security measure by hidden communication to cover a message from third party. The objectives of the study are highlighted as below:

i. To carry out a critical review of the concept of information security

ii. To carry out a thorough survey of the various steganography types we have

iii. To carry out an elaborate investigation into image steganography

SCOPE AND LIMITATION OF THE STUDY

This application hides data or information inside images (in a process called steganography) using Bitmap image (.bmp format) or JPG (.jpg) or gif format as its carrier file.

However;

- Only texts can be hid using this method; while other data forms may not work with it.
- Password have to be shared which can be hacked and used.
- Have to manually send the image to receiver.

1.3 SIGNIFICANCE OF THE PROJECT

Due to the incessant problem of security breach faced during information exchange (between a sender and a receiver), there is a need to devise a means of communicating secretly which will help in overcoming this menace of insecurity encountered during data transfer. With steganography, it will take an intruder a life time trying to reveal the message protected by the user. In other words, the security of the message is unarguably guaranteed.

1.4 DEFINITION OF TERMS/ACRONYMS USED

Audio/Video: is similar to the two techniques above, only that is usesaudio/video as a cover object to hide the existence of data.

Images Steganography: is the widely used as a cover object to hidedata due to high

redundancy number of bits.

LSB: Least Significant Bits

Protocol: this is the techniques of hiding the existence of data by embedding the data within a message and network protocols used in network transmission.

CHAPTER 2

LITERATURE SURVEY

Many researchers have work in relation to security protection in data transfer over internet. Below are few of work done:

Sahoo and Tiwari (2008), proposed a good method. In their method works on more than one image using the concept of file hybridization. This particular method implements the cryptographic technique to embed two information files using steganography. And due to this reason they have to use a stego key for the embedding process.

P. Marwaha et al., Proposed that Cryptography and steganography are the most extensively used techniques. Both these techniques provide some security of data neither of them individually is secure enough for sharing information over an unsecure communication channel and are unguarded to intruder attacks.

Gauravram P. in his thesis has suggested that the usage of cryptographic hash functions in several information processing applications to achieve various security goals is much more widespread than application of block ciphers and stream ciphers.

Rajeev & Geetha (2012), in addition to using Hash Functions for implementing MAC, Hash functions can be used to achieve message authentication and integrity goals without the use of symmetric encryption.

Rompay (2004), has also detailed the ways of ensuring authentication using hash functions alone as well as using hash functions with encryption.

2.1 AN OVERVIEW OF MODERN STEGANOGRAPHY

Due to the technology transformation steganography techniques has been transformed from ancient times till modern steganography where data to be passed across is hidden in the cover object such as images, audio/video, text, protocol by hidden the fact that communication is taking place rather than transforming the data to another form.

Many digital file formats can serve as cover object for hidden data in them, but the most preferably one is the one that has the high redundancy which is the bits of an object that provides accuracy far greater than the necessary for the object use and display (Morkel, Eloff,

& Olivier, 2005). The redundant bits can be defined as the number of bits that can be altered without the alteration being detected easily, an example is image and audio file formats.

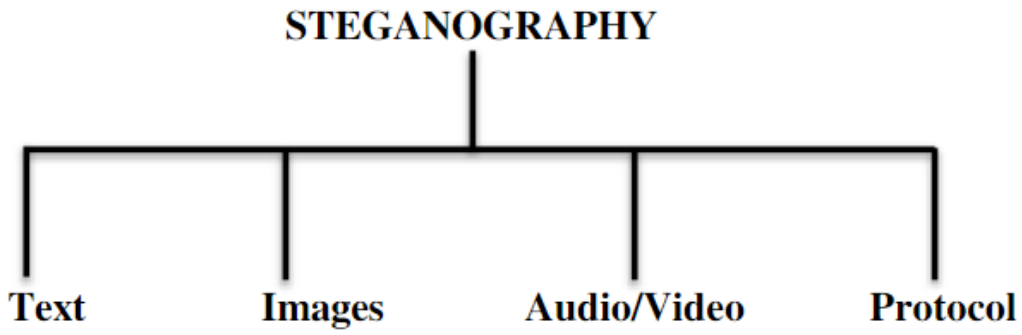


Fig. 2.1. cover object for steganography

As it can be seen above, there are four main file formats that can be used as cover object for hidden data.

a. Image Steganography

The Image Steganography is technique in which we hide the data in an image so that there will not be any change in the original image.

b. Audio Steganography

Audio Steganography can be used to hide the information in an audio file. The audio file should be undetectable.

c. Video Steganography

Video Steganography can be used to hide the information in video files. The video files should be undetectable by the attacker.

d. Text files Steganography

Text Steganography is used to hide the information in text files. The general process of steganography i.e., preparing a stego object that will contain no change with that of original object is prepared but using text as a source

2.2 STEGANOGRAPHY ALGORITHM

SUBSTITUTION ALGORITHM

Due to increase in the number of digital images over the internet and the high level of redundant bits of data is contained in digital image; there has been a rise in the use digital image for the purpose of steganography.

There are basically two broad categories of the substitution algorithms used in enhancing image in the area of image steganography and they are:

1.**Spatial Domain Techniques**

2.**Transform Domain Techniques**

- **SPATIAL DOMAIN TECHNIQUES (IMAGE DOMAIN)**

In this method, there is direct manipulation of the image pixel. This technique hides the secret data by replacing the chosen bits from the cover image with the bit value of the secret message.

This implies that the secret data has to be converted into its equivalent bit value before it can be substituted into the cover image. The most widely known steganography algorithm is the Least Significant Bit (LSB) algorithm (Mustafa, ElGamal, EIAImi, & Ahmed, 2011). It operates by substituting the Least Significant Bit of an image pixel with a representation of the secret message which is not detectable by the human eye as long as there is no clear degradation in the image quality.

There is other spatial domain algorithm that can change the LSB of the carrier image in a random manner and others do so by increasing or decreasing the pixel value (Wilson & Bryon, 1992). Examples of such algorithm are Pixel Value Differencing (PVD), Random Pixel Embedding Method, Texture Based Method, and Histogram Shifting Method and so on.

- **TRANSFORM DOMAIN TECHNIQUES**

This technique is complicated than the spatial domain technique in the sense that it hides the secret information in areas that are less susceptible to compression, cropping and other forms of manipulation. The data are hidden inside some mathematical function like sine/cosine function. One of the most commonly used transform domain technique is the discrete cosine transformation (DCT) which altars the discrete cosine transformation of the cover image. Basically, it obtains the respective DCT coefficient of the image and then looks for any value that is lower than a particular value known as the threshold. The returned value is then replaced with the secret bits to produce the stego image.

Other forms of transform domain techniques include: Discrete Fourier Transformation (DFT), Discrete Wavelet Transformation techniques and Lossless DCT Method.

2.3 LEAST SIGNIFICANT BIT (LSB) ALGORITHM

The Least Significant Bit (LSB) algorithm adjusts the least significant bit pixels of the image which in this case is the carrier file. In this algorithm, the bit insertion depends on the number of bits in the image. For instance, in an 8bit image, the LSB of each byte of the image will be changed to the bit of the secret message. In 24bit image, the RGB (Red, Green, Blue) color components are substituted accordingly with the MSB (Most Significant Bit) of the secret data When used with a JPEG image, LSB algorithm is very effective due to the lossless ability of jpeg image.

2.4 HIDING A TEXT WITHIN A PICTURE

Least Significant Bit (LSB)

- One of most common techniques
- Alters LSB of each pixel (1 bit out of 24 or 1 out of 8 for gray scale)
- Uses the concept of parity, i.e., even numbers in binary end in 0, odd one end in 1
- Easiest to implement: hiding bitmaps in a color picture
- Hiding ASCII code, one letter at a time

Example:

To hide letter C, which ASCII code is 67 and binary equivalent number is 1000011 in a gray scale file:

Original:

01001101 01001110 01001110 01001111 01010000 01010000

1 0 0 0 0 1

01001111

1

Encoded:

01001101 01001110 01001110 01001111 01010000 01010001

0100111

Encoding a message

The function encode() creates an image with message written in big black letters across it; it then invokes encode Picture() to hide the image with the message in the given picture.

.

CHAPTER 3

METHODOLOGY

INTRODUCTION

This chapter presents the analysis and design for the implementation of the proposed system. A system is a collection of inter-related elements and procedures that work together harmoniously to achieve certain goals. On the other hand, analysis is a careful examination of the composing varying elements of the system in order to understand it better. Therefore, system analysis is a process of collecting, organizing, and looking through fact in respect of existing operations, procedures, and systems so as to get full appreciation of the situation prevailing with the aim of designing and implementing an effective computerization program.

3.1 SYSTEM ANALYSIS

Systems analysis is a term that collectively describes the early phases of systems development. It gives a concise and well-grounded description of the intended system design. Making an insight in to the step by step procedures that will be used in building and developing the final system. It deals with the dissection of a system into its component pieces to study how those component pieces interact and work.

This section discusses the various parts of steganography technique.

I study the sets of algorithm, text, key and interacting entities that make up the system. There are a number of different approaches to system analysis (according to the waterfall model) would constitute the following steps.

- The development of a feasibility study, involving determining whether a project is economically, socially and organizationally feasible.
- Conducting fact-finding measures, designed to ascertain the

  requirement of the requirement of the system's end users.

- Gauging how the end user would operate the system (in terms of general experience in using computer hardware and software); what the system would be used for and so on.

An interaction and structural model of systems is used for the analysis of the proposed system.

3.2 ANALYSIS OF THE EXISTING SYSTEM

Security has always been a bedrock for transaction and protection of human lives and until the existence of artificial intelligence, humans have learned to trust themselves for each other safety or safety of information except for few times where security was threatening by breach of trust or betrayal by personal which often lead to enormous crisis. For every system and situation available there was a means of security available. For instance, indigenously, birds were sent on errands to neigh bouring villages or towns to announce either the birth or death of a royal blood, spies and allies where used by kings to convey very confidential messages within and around the village, Aroko (this is a material used mostly in the western part of Nigeria especially amongst the Yoruba's) it was a symbol of information which can only be interpreted by the intended receiver despite receipt by another person, these and many more secured ways were explored in the ancient times before the advancement of modern technology.

Increasingly, as more and more situations called for stricter and tighter means of security new and inventive ways were created. A real life example is seen in the Banking sector where forms containing security questions were used in the release of funds. The banks required customers to fill forms and provide correct answers tot he personal questions which apparently only the rightful owners knew, this method was used for a while till fraud became too difficult to control and banks started to invent passcodes which only rightful owners knew and gave them access to their funds. These and many more advancing security methods have been employed by banks in order to guarantee maximum security and reduce risk to its minimum.

Other means of ensuring security include sworn secrecy, oaths and many more before the advent of artificial intelligence. These measures and forms of security had several loop holes which flawed them and made security less efficient. Taking password code as an example, it was possible for an attacker to guess the random selection of numbers or letters

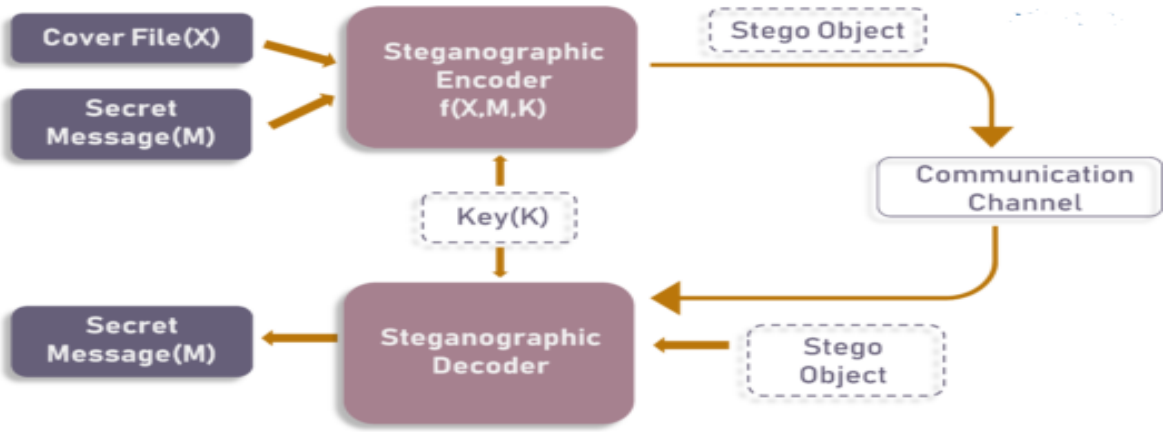
used in forming the password thereby defeating the security and carting away confidential information, People have always wanted to keep their words and thoughts secret, hidden from prying eyes and as such, the options available to keep secrets safe are either dismal, or draw unwanted attention to the presence of a locked secret; hence, they introduced an approach for securing and safely hiding things in an unsuspecting manner.Hence, the inception of steganography The most common methods to make alterations involve the use of the Least Significant bit LSB for masking, filtering and transformations on the cover image; the art and science of making communication unintelligent to all except the intended recipient.

CHAPTER 4

PROPOSED SYSTEM

THE PROPOSED SYSTEM

The proposed system deals with information and a means of securing it. It uses one of the methods of information security; Steganography. The rapid development in the transfer of data through internet made it easier to transfer data accurate and faster to the destination.



Security of information is one of the important factors of information technology and communication. Steganography is art and science of invisible communication. Steganography is the method through which existence of the message can be kept secret. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information.

4.1 PROPOSED ALGORITHM

The algorithm that is adopted for the course of this project work is

Fig. 3.1 Graphical representation of Steganography

4.1.1 LEAST SIGNIFICANT BIT (LSB)

The Least Significant Bit (LSB) algorithm adjusts the least significant bit pixels of the image which in this case is the carrier file. In this algorithm, the bit insertion depends on the number of bits in the image. For instance, in an 8bit image, the LSB of each byte of the image will be changed to the bit of the secret message. In 24bit image, the RGB (Red, Green, Blue) color

components are substituted accordingly with the MSB (Most Significant Bit) of the secret data When used with a JPEG image, LSB algorithm is very effective due to the lossless ability of jpeg image.

4.1.2 ALGORITHM TO EMBED TEXT MESSAGE: -

Step 1: Read the cover image and text message which is to be hidden in the cover image.

Step 2: Convert text message in binary format.

Step 3: Calculate LSB of each pixels of cover image.

Step 4: Replace LSB of cover image with each bit of secret message one by one.

Step 5: Displays stego image

4.1.3 ALGORITHM TO RETRIEVE TEXT MESSAGE: -

Step 1: Read the stego image.

Step 2: Calculate LSB of each pixel of stego image.

Step 3: Retrieve bits and convert each 8 bit into character.

Step 4: Displays the original message

CHAPTER 5

CODING & TESTING

5.1 Python Source code

```python
from PIL import Image
import numpy as np
```

The PIL (Python Imaging Library) and NumPy libraries are useful for image processing and manipulation.
PIL (Pillow): Used for opening, manipulating, and saving images. The module Image from PIL can be used to load and manipulate images.

NumPy: Used to handle numerical data, which can be very useful for processing the pixel data in images.

To hide a message in an image a function is needed that converts a string to a binary representation and one that embeds that data in the least significant bits of an image. To reveal the message we need to be able to reverse this, so first a function will extract the least significant bits and the final step is to convert them back into a string.

```python
# Convert the hidden message to bytes

def encode_text(text, encoding='utf-8', errors='surrogatepass'):
    bits = bin(int.from_bytes(text.encode(encoding, errors), 'big'))[2:]
    return bits.zfill(8 * ((len(bits) + 7) // 8))
```

Python

```python
def decode_text(bits, encoding='utf-8', errors='surrogatepass'):
    n = int(bits, 2)
    return n.to_bytes((n.bit_length() + 7) // 8, 'big').decode(encoding, errors) or '\0'
```

Python

To convert text to binary -

str.encode is used to turn the string into bytes int.from_bytes now creates an integer number from those bytes. This works because in Python integer numbers can be arbitrarily large. this number is converted into binary (e.g. 0b00101011010101010101010001...) using bin() an array slice is used to remove the first two characters (0b) zfill is used to make sure the output is a multiple of 8 To revert from a binary the padding and slicing can be omitted and the other steps need to be reversed.

int() is used to convert a binary representation back to an integer number, note the parameter 2 using the int.to_bytes this number is converted back to a list of bytes str.decode converts bytes back to text

```
hidden_message = "India is My Country"

encoded_text = encode_text(hidden_message)
decoded_text = decode_text(encoded_text)

print("encoded:", encoded_text)
print("decoded:", decoded_text)
```

Python

```
encoded: 0100100101101110011001000110100101100001001000000110100101110011001000000100110101111001001000000100001101101111011010101101110011101000111001001111001
decoded: India is My Country
```

Here the PIL library is used to load the image, which is than turned into a one-dimensional list of all Red, Green and Blue values of the pixel in the image.

```
def encode_in_image(filename, text_message):
    # Open the image, store the shape and convert to one-dimensional list
    input_im = Image.open(filename, 'r').convert("RGB")
    image_shape = np.asarray(input_im).shape
    flat_array = np.asarray(input_im).flatten()

    # Encode the message and add prefix
    encoded_text = encode_text(text_message + "<STOP>")

    # Enter message in the least significant bit where necessary
    encoded_array = [
        (0b11111110 & value) | int(encode_bit) if ix < len(encoded_text) else value
        for ix, (encode_bit, value) in enumerate(zip(encoded_text.ljust(len(flat_array), '0'), flat_array))]

    # Turn encoded array into image and return
    encoded_im = np.array(encoded_array).reshape(image_shape)
    return Image.fromarray(np.uint8(encoded_im)).convert('RGB')



encoded_im = encode_in_image('modi.jpg', "Attch on China")
encoded_im.save('hidden.png')
encoded_im
```

[14] Python

5.2 RESULTS

```
def extract_from_image(filename):
    # Open image
    encoded_im = np.asarray(Image.open(filename, 'r').convert("RGB"))

    # Extract least significant bits from flat (one-dimensional) image
    extracted_bits = [str(0b00000001 & value) for value in encoded_im.flatten()]

    # Join bits together, decode and split at <STOP>
    extracted_bits = ''.join(extracted_bits)
    return decode_text(extracted_bits, errors='replace').split('<STOP>')[0]

extract_from_image('hidden.png')
```

25] Python

```
'Indian army always for your Security'
```

ENCODED IMAGE

5.3 Discussion

The provided code implements a basic steganography technique using the Least Significant Bit (LSB) method to embed a text message within an image. Here's a concise overview of its components, strengths, limitations, and potential improvements.

Key Components

1. Encoding Functionality:
    - The encode_in_image function opens an image, flattens its pixel data, and encodes the text message into a binary string, appending a <STOP> delimiter.
    - It modifies the least significant bits of the pixel values to embed the message.
2. Decoding Functionality:
    - The extract_from_image function retrieves the least significant bits from the image, reconstructs the binary string, and decodes it back into text, stopping at the <STOP> delimiter.

Strengths

- Simplicity: The code is straightforward and effectively

demonstrates LSB steganography.

- Flexibility: It allows for embedding any text message, making it versatile for various applications.
- Clear Structure: The use of a stop sequence aids in accurate message extraction.

Limitations

- Data Capacity: Limited to the least significant bits, restricting the amount of data that can be hidden.
- Vulnerability: Sensitive to image modifications (e.g., compression), which can corrupt the hidden message.
- Lack of Security: The embedded message is not encrypted, making it susceptible to unauthorized access.

Potential Improvements

1. Advanced Techniques: Implement adaptive methods or use multiple channels to increase data capacity and robustness.
2. Error Correction: Incorporate error correction to recover messages even if some bits are altered.
3. Encryption: Add a layer of encryption to enhance security.
4. User Interface: Develop a simple GUI for easier use by non-technical users.
5. Validation: Implement tests to ensure the integrity of the hidden message after extraction.

CHAPTER 6

CONCLUSION

The steganography project successfully implemented a Least Significant Bit (LSB) method to embed text messages within image files, demonstrating its potential for secure communication. The code featured two main functions: one for encoding a text message into an image and another for extracting the hidden message from the modified image. The implementation accurately embedded the message "Attach on China" into the image modi.jpg, resulting in a new file hidden.png, from which the original message was successfully retrieved. While the project showcased simplicity, flexibility, and clear extraction processes, it also highlighted limitations such as restricted data capacity, vulnerability to image processing alterations, and lack of encryption. Future enhancements could include advanced encoding techniques, error correction, encryption for security, and the development of a user-friendly interface. Overall, the project provides a solid foundation for further exploration and application of steganography in data hiding and security.

CHAPTER 7

FUTURE ENHANCEMENT

1. Advanced Encoding Techniques

- **Adaptive Steganography**: Implement algorithms that adaptively choose the best pixels for embedding based on their color intensity or other characteristics, improving capacity and imperceptibility.
- **Multi-channel Embedding**: Utilize multiple color channels (Red, Green, Blue) or even different image formats (e.g., using alpha channels in PNGs) to increase data capacity.

2. Robustness Improvements

- **Error Correction Codes**: Integrate error correction techniques to recover the hidden message even if some bits are altered or lost due to image processing.
- **Resilience Against Compression**: Develop methods to protect embedded data from lossy compression techniques commonly used in image formats (like JPEG).

3. Security Enhancements

- **Encryption of Hidden Data**: Before embedding, encrypt the message to ensure that even if the hidden data is extracted, it cannot be easily read without the decryption key.
- **Digital Watermarking**: Consider adding a watermarking feature to protect copyright and ownership of the images.

4. User Interface Development

- **Graphical User Interface (GUI)**: Create a user-friendly GUI to allow non-technical users to easily embed and extract messages without needing to modify code.
- **Batch Processing**: Enable the application to handle multiple images and messages simultaneously for efficiency.

5. Cross-Platform Compatibility

- **Web Application Version**: Develop a web-based version of the tool that allows users to upload images and messages without needing to install software.
- **Mobile Application**: Create a mobile app for easy access to steganography features on smartphones.

6. Performance Optimization

- **Speed Enhancements**: Optimize the encoding and decoding processes to handle larger images and messages more efficiently.
- **Memory Management**: Improve memory usage, especially when dealing with high-resolution images.

7. Testing and Validation

- **Integrity Checks**: Implement mechanisms to verify the integrity of the hidden message after extraction, ensuring it matches the original input.
- **User Testing**: Conduct user testing to gather feedback on usability and effectiveness, leading to further refinements.

8. Integration with Other Technologies

- **Blockchain for Security**: Explore the use of blockchain technology to securely store and verify the integrity of the hidden messages.
- **Machine Learning**: Use machine learning techniques to improve detection and extraction processes, making them more robust against attacks.

REFERENCES

1. Ansari, A.Q.; Khan, M.E.; Pant, M., "Data Security by Steganography: A Review," ResearchGate, vol., no., pp., 2019.https://www.researchgate.net/publication/379806300_Data_security_by_steganography_A_review
2. Sharma, S.; Parashar, P., "Steganography in Images Using LSB Technique," ResearchGate, vol., no., pp., 2020. https://www.researchgate.net/publication/371671984_Steganography_in_Images_Using_LSB_Technique#:~:text=DWT)%20based%20steganography.-,The%20LSB%20algorithm%20is%20implemented%20in%20spatial%20domain%20in%20which,the%20frequency%20domain%20and%20the
3. Changder, S.; Chakraborty, M.; Sarkar, R., "Steganography Techniques and Their Applications in Security," IEEE Explore, vol., no., pp., 2001. https://ieeexplore.ieee.org/document/959097
4. Fridrich, J., Goljan, M., Du, R., 2001. Detecting LSB steganography in color and gray scale images. IEEE Multimed. 8, 22–28. https://doi.org/10.1109/93.959097.
5. Supriadi Rustad⇑, De Rosal Ignatius Moses Setiadi, Abdul Syukur, Pulung Nurtantio Andono, Inverted LSB image steganography using adaptive pattern to improve imperceptibility. https://doi.org/10.1016/j.jksuci.2020.12.017.
6. Thampi, S.M.; Mukhopadhyay, S.; Moschoyiannis, S., "A Survey on Image Steganography," IEEE Explore, vol., no., pp., 2015. https://ieeexplore.ieee.org/document/7226122