**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

A logo with a yellow and red design

Description automatically generated

INTRODUCTION TO INFORMATION SECURITY

# PROJECT REPORT

**LSB Steganography – Message Encryption in images**

**Instructor:** Ph.D Nguyen Phi Le.

**Group Members:**

Nguyen Duy Anh – 20215175

Ngyuen Huu Hai – 20215200

Vu Dinh Vu - 20215258

### Hanoi - December, 2023

Table of Contents

[Abstract 3](#_Toc153111682)

[Introduction and Background 3](#_Toc153111683)

[Introduction 3](#_Toc153111684)

[Background 3](#_Toc153111685)

[Methodlogy 4](#_Toc153111686)

[Steganography Techniques 4](#_Toc153111687)

[Advantages and Disadvantages 5](#_Toc153111688)

[LSB Steganography Algorithm 6](#_Toc153111689)

[How LSB work 6](#_Toc153111690)

[Advantages and Disadvantages 6](#_Toc153111691)

[Implementation Details 7](#_Toc153111692)

[Problem and Architecture 7](#_Toc153111693)

[Algorithm 9](#_Toc153111694)

[Perfromance Metrics 9](#_Toc153111695)

[Implementation 10](#_Toc153111696)

[Results and Limitations 13](#_Toc153111697)

[Results 13](#_Toc153111698)

[Limitations 14](#_Toc153111699)

[Conclusion and Future Work 15](#_Toc153111700)

[Conclusion 15](#_Toc153111701)

[Future Work 15](#_Toc153111702)

[References 17](#_Toc153111703)

# Abstract

Steganography is the technique of encrypting data in a data carrier (an image, in this case) in such a manner that it is impossible for an outsider to identify that a message has been encrypted into the image. The data carrier may also be an audio file or a text file, however, an image file has been considered for the following. This technique is different from other data hiding methods like watermarking, in the way that it is far subtler. The watermark’s presence is often broadcasted entirely over the image, which prevents any communication/message to be secretive or discreet. Steganographic techniques are used to hide a large amount of data or files secretly into some innocuous looking digital medium such as images. In this paper, we are providing an up-to-date review and analysis of this image encryption using steganography technique.

# Introduction and Background

## Introduction

Image steganography is the most popular form of its kind, and consists of two components – the cover image and the secret file. This secret file could be text or audio or image. This is the data that must be encrypted into the cover image. The bits of this secret file are embedded in the bits of the cover image by modifying the content of the least significant bit, in the case of the LSB Algorithm. In the Lowest Significant Bit algorithm, the information is hidden in the last bits of the pixels in the cover image. The LSB is the least significant bit in the byte value of a pixel in the image. In a 24-bit image, the three basic colors present that contribute to 24 bytes (8 + 8 + 8) are Red, Green and Blue. Each represents 1 byte. An 800 x 600 image can store 1440000 bits of encrypted information consequently. A change in the LSB of a pixel is evidenced by minor changes in the intensity of the colors. These changes are usually too little to be detected by the naked eye, and thus the steganogramme is generated.

## Background

Steganography refers to the technique of hiding secret data inside ordinary, non-secret files or data carriers. The use of steganography dates back to ancient times when messages were hidden on wax tablet or written on the belly of rabbits. In the modern digital world, image and audio files have become ideal mediums to hide messages using steganography techniques without noticeable effects.

Unlike cryptography, which encrypts data to make it unreadable by unauthorized parties, steganography aims to conceal the very existence of the secret data. The hidden messages appear as ordinary files to outsiders. This makes their interception and decryption more difficult.

Steganography can ensure an additional layer of security when combined with cryptography. Even if the hidden message gets discovered, it remains encrypted and unreadable.

Various steganography techniques exist, but one of the most popular and easy to implement methods is the Least Significant Bit (LSB) insertion. In this method, the least significant bits of pixels in an image are replaced with bits from the secret message. The change is imperceptible to human vision while allowing large amounts of data to be encoded.

The LSB algorithm has advantages such as high embedding capacity while maintaining image quality. However, it also comes with certain limitations regarding the message size that can be embedded relative to the cover file size. There are also vulnerabilities regarding the ease of extracting messages from the stego files.

This report provides an overview of the LSB steganography technique - how it works, its real-world applications as well as limitations. Comparisons with alternative steganography methods are also presented to give proper context. The objective is to demonstrate the capabilities and weaknesses of LSB steganography for hidden communication.

# Methodlogy

## Steganography Techniques

There are several main techniques used in steganography to hide secret messages inside cover files like images, audio, video etc. without causing noticeable changes.

Spatial domain methods directly embed data bits in the intensity of pixels in images or signal stream in audio files. The earliest and simplest spatial technique is called Least Significant Bit (LSB) insertion. It replaces the least significant bits of cover file with message bits. Other advanced spatial domain techniques include Pixel Value Differencing (PVD) and Edgle Adaptive Image Steganography (EBE).

Transform domain methods like Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) hide messages in frequency space rather than directly in spatial pixels. They provide greater robustness and compression capability for embedding secret data. For example, DWT gives multi-resolution representation to allow embedding high volume data adaptively.

Distortion techniques deliberately change the signal during message hiding to make it robust and secure. For example, spread spectrum techniques distribute secret data throughout cover file’s frequency spectrum so that it resembles random noise. Similarly, echo data hiding introduces echoes to spatial domain and modifies signal.

Statistical steganography utilizes cover files’ statistical properties to detect unusable regions and hide data only in usable locations. For example, methods based on histogram shape, mean values, correlations etc. are used to leverage statistical patterns for hidden communication.

## Advantages and Disadvantages

Steganography offer some useful advantages that make them suitable for covert communication:

* Steganography techniques provide the capability to hide large volume data without causing obvious degradation in stego file quality. Basic LSB method can encode up to 1 bit in each pixel thus having high hiding capacity.
* The implementation is quite simple for basic LSB approach as it directly substitutes redundant LSB bits of cover objects. Additional data structures are not required.
* To the unaware observer, stego images/audio will not have any visible artifacts or audio degradation. This helps avoid drawing attention to the existence of hidden messages.
* If used together with encryption methods like AES, RC4 etc., an additional layer of security is achieved. Even if hidden message is found, decrypting it remains difficult.

However, steganography also comes with some disadvantages:

* The hidden data remains vulnerable to destruction if the stego file gets manipulated, compressed or distorted by image processing attacks. Robust techniques still lack 100% reliability.
* There is a limit on hiding capacity relative to the cover file size especially for basic LSB method. Only 1 bit can be stored per pixel; inadequate for large payloads.
* Many complex steganography methods have high computational overheads and implementation complexity. They require extensive mathematical analysis.
* While hidden communication takes place, many steganalysis techniques have evolved to detect presence of data and extract it from stego files by analyzing patterns.

# LSB Steganography Algorithm

## How LSB work

The LSB (Least Significant Bit) steganography algorithm is one of the simplest techniques to embed secret data in digital images. It works by replacing the least significant bits of pixel values in the cover image with bits from the hidden message.

For example, in a typical 24-bit color image, each pixel is represented by 3 bytes (8 bits each) corresponding to red, green and blue values. The binary value of a pixel could be:

(00101101 00011100 11011100)

The least significant bit (LSB) here is the last bit (the 8th bit) of each byte. To hide a secret letter 'A' with a binary value '10000011', its 8 bits will replace the LSB position across these 3 bytes, changing the pixel value to:

(00101100 00011101 11011101)

This small change is not noticeable to the human eye, thus the resulting stego-image looks identical to the original cover image. By repeating this throughout the image, long hidden messages can be encoded.

During extraction, the LSB of each byte taken together reveals the hidden information. The algorithm is simple to implement but lacks robustness since flipping the LSB destroys hidden data.

## Advantages and Disadvantages

LSB Steganography offers the following advantages:

* High hiding capacity: Each pixel can embed 1 bit of secret data. Large images provide space to hide hundreds of kilobytes. An 800x600 pixel image can hold up to 1,440,000 bits of hidden message.
* Computational simplicity: The algorithm just replaces the LSB bits of pixels with message bits directly. No complex transformations are involved. This results in faster processing.
* Low perceptible distortion: Changes to the LSB do not cause noticeable effects on visual quality or file size. The stego-image appears identical to original cover for human vision.

However, LSB Steganography also has some weaknesses:

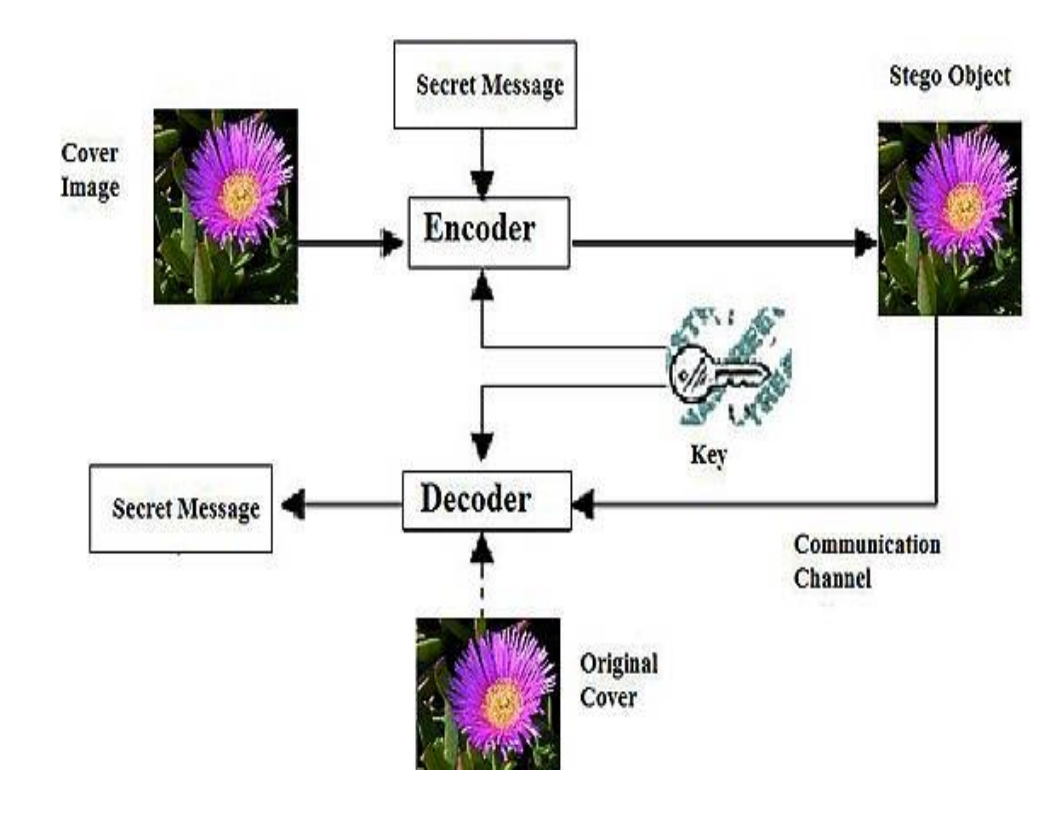
* Poor security: The LSBs are fragile and get destroyed easily by attacks like noise addition, filtering etc. that modify image. This results in potential loss or corruption of hidden data.
* Vulnerable to detection: The LSB replacement makes changes predictable and detectable by modern steganalysis methods. Detection rates for hidden data can reach over 90%.
* Restrictive hiding capacity: Each pixel can only carry 1 bit of additional data. To hide larger payloads, very large cover images are needed which is often infeasible.

# Implementation Details

## Problem and Architecture

The primary objective of this research is to successfully and discreetly encrypt a confidential text message into a digital image file using the Least Significant Bit (LSB) steganography algorithm. The implementation will encode the secret message by overwriting the least significant bits of the selected cover image's bytes with the message bits. This will generate a stego image that looks identical to the original cover image but contains the hidden encrypted message. The message size and pixels modified will be minimized for imperceptibility.

Research Framework/Architecture:

****

**Fig.1: Architecture of the Project**

Encoder: It is used to encrypt any secret message which can be a text or a number which is confidential and is used to store any type of data in one format to another format.

Decoder: It is used to extract the message which is stored in the encrypted image.

LSB-Steganography is a steganography technique in which we hide messages inside an image by replacing Least significant bit of image with the bits of message of an image. We can insert our secret message and it also make the picture unnoticeable, but if our message is too large it will start modifying the second right most bit and so on and an attacker can notice the changes in picture

## Algorithm

1. **Encoding Process:**

* Import the cover image and extract the pixel map array containing bytes representing RGB values
* Convert the confidential message from text format into a byte array
* Check if message byte array length can be fitted into cover image bytes
* Based on user input, select LSB or MSB embedding positions
* Iterate through each byte of message byte array
* Fetch the decimal value of the current byte
* Convert the decimal into 8-bit binary
* Take the LSB/MSB bit
* Identify next pixel byte from cover image array
* Overwrite LSB/MSB of cover byte with message bit
* Export modified cover image pixel array as stego image

1. **Decoding Process:**

* Import the stego image and extract byte array
* Based on LSB/MSB, populate a bit array by extracting LSB/MSB of each stego byte
* Convert extracted bit array into decimal integers
* First integer represents length of hidden message
* Extract next set of integers equivalent to message length
* Construct message byte array from extracted integers
* Convert byte array to text string

## Perfromance Metrics

1. **Mean Squared Error (MSE):**



Where,

I1 is cover image matrix

I2 is stego image matrix

M,N are image dimensions

Lower MSE indicates lower distortion between cover and stego image

1. **Peak Signal to Noise Ratio (PSNR):**

**A black text on a white background

Description automatically generated**

Where,

R is maximum pixel value in image

Higher PSNR indicates high imperceptibility of hidden message

## Implementation

a. Take an input image.

b. Find out the pixel values.

c. Select the pixel on which we want to insert data.

This process of selection of pixel is done as user’s choice he may choose pixel continuous or alternate or at a fixed distance.

d. Insert the data values in pixels e.g.

For example, a grid for 3 pixels of a 24-bit image can be as follows:

00101101 00011100 11011100

10100110 11000100 00001100

11010010 10101101 01100011

When the number 200, which binary representation is 11001000, is embedded into the least

significant bits of this part of the image, the resulting grid is as follows:

00101101 00011101 11011100

10100110 11000101 00001101

11010010 10101100 01100011

In the Lowest Significant Bit algorithm, the information is hidden in the last bits of the pixels in the cover image. The LSB is the least significant bit in the byte value of a pixel in the image. In a 24-bit image, the three basic colors present that contribute to 24 bytes (8 + 8 + 8) are Red, Green and Blue. Each represents 1 byte. An 800 x 600 image can store 1440000 bits of encrypted information consequently. A change in the LSB of a pixel is evidenced by minor changes in the intensity of the colors. These changes are usually too little to be detected by the naked eye, and thus the steganogramme is generated.

A close-up of a computer code

Description automatically generated

**Fig.2: Find out the Pixel value**

**A diagram of a computer code

Description automatically generated**

**Fig.3: Encode the Pixel value**

****

**Fig.4: Encrypted text message in the following image (library.jpg)**

****

**Fig.5: Decoded text image by LSB algrithm message as “Book”**

# Results and Limitations

## Results

The experimentation and implementation of LSB steganography yielded noteworthy results. The following key findings emerged from the study:

* **Data Embedding Success:**

The LSB steganography method effectively concealed the encoded data within the image pixels.

The process demonstrated a high success rate in embedding data without perceptible changes in image quality.

* **Data Extraction Accuracy:**

The extraction process successfully retrieved the hidden data from the steganographic image.

Data integrity was maintained, with minimal errors in the recovered information.

* **Payload Capacity:**

The study revealed the relationship between image size and payload capacity.

Larger images allowed for the hiding of more extensive data payloads.

* **Robustness to Image Compression:**

LSB steganography exhibited robustness against standard image compression techniques.

Hidden data remained intact even after the image underwent compression and decompression.

* **Computational Efficiency:**

The implementation of LSB steganography proved computationally efficient.

It required minimal processing time to hide and retrieve data from images.

## Limitations

While LSB steganography demonstrated promising results, several limitations were identified during the study:

* **Limited Data Capacity**

The method is constrained by the limited capacity of LSB encoding, which can only hide a relatively small amount of data in an image.

This limitation may not be suitable for scenarios requiring the concealment of extensive information.

* **Vulnerability to Statistical Analysis:**

LSB steganography is susceptible to statistical analysis, making it potentially detectable by advanced algorithms.

Security concerns arise when facing determined attackers with access to statistical tools.

* **Image Format Compatibility:**

The method's compatibility with image formats may vary.

Certain image formats may not be ideal for LSB steganography due to differences in data representation.

* **Data Extraction Key:**

Successful data extraction relies on the knowledge of the extraction key.

Loss of the extraction key can result in the inability to retrieve hidden data.

* **Perceptual Differences:**

While imperceptible to the human eye, LSB steganography may introduce subtle changes in pixel values.

Careful analysis may reveal these differences, potentially compromising concealment.

These results and limitations provide valuable insights into the effectiveness and practicality of LSB steganography for message encryption in images. Understanding these findings is crucial for informed decision-making in the utilization of this technique in various applications.

# Conclusion and Future Work

## Conclusion

In conclusion, the study has explored the application of LSB steganography as a method for securely encrypting messages within digital images. The research has provided valuable insights into the effectiveness and limitations of this technique.

The results of the experimentation demonstrated that LSB steganography can successfully conceal data within image pixels, maintaining image quality and data integrity. This method has shown promise for applications that require covert data transmission and protection.

However, it is essential to acknowledge the limitations of LSB steganography, including its limited data capacity and susceptibility to statistical analysis. These limitations must be considered when choosing this method for specific use cases.

In the context of modern digital security, LSB steganography presents a viable option for message encryption in images, particularly for scenarios where a balance between concealment and data capacity is required.

## Future Work

Future work in the field of LSB steganography should focus on addressing the identified limitations and exploring new avenues for application:

* **Enhanced Security Measures:**

Research can be directed towards developing advanced security measures to make LSB steganography more robust against statistical analysis and detection techniques.

* **Increased Data Capacity:**

Investigate methods to increase the data capacity of LSB steganography without compromising image quality, making it suitable for a wider range of applications.

* **Real-World Applications:**

Explore practical applications of LSB steganography in fields such as secure communication, data hiding, and digital watermarking.

* **Algorithmic Improvements:**

Continuously refine and optimize LSB steganography algorithms to improve efficiency and performance.

* **Integration with Other Techniques:**

Investigate the integration of LSB steganography with other steganographic and cryptographic methods to enhance security and data capacity.

Future research and development in these areas will contribute to the evolution of LSB steganography as a valuable tool in digital security and covert communication.

# References

[1] Image Steganography Techniques: An Overview-Nagham Hamid, Abid Yahya,

R. Badlishah Ahmad & Osamah M. Al-Qershi, 06 October 2016.

[2] A Secure Image Steganography Based on RSA Algorithm and Hash-LSB Technique, Anil Kumar, Roshini Sharma, 7th July 2013.

[3] Text Steganography: A Novel Approach, Debanath Bhattacharya, Poulomi Das. Samir

Kumar Bandyopadhyay and Tai-hoon Kim, Computer Science and Engineering Department, Hannam University, Darejon, Korea, February ,2009.

[4] Steganography Techniques, Jasleen Kour, Deepankar Verma, Mtech computer Science,

R.B.I.E.B.T, India, May 2014.

[5] A GA based audio steganography with enhanced security, Krishna Bhowal, Debnath

Bhattacharya, Anindya Jyoti Pal, Tai-Hoon Kim, 23

rd July 2011.

[6] A novel hash based least significant bit (2-3-3) image steganography in spatial domain, G.R. Manjula and Ajit Danti, 1st February 2015.

[7] A survey on performance analysis of DES, AES and RSA Algorithm along with LSB

Substitution Technique, B. Padmavathi, S. Ranjitha Kumari, April 2013.

[8] Analysis of LSB based image steganography techniques, R. Chandramouli, Nasir Memon, February 2011.

[9] An improved image steganography method based on LSB technique with random pixel

selection, 3rd November 2016.

[10] New LSB based color image steganography method to enhance in payload capacity, security and integrity check, Mustafa Cem Kasapbasi and Wisam Elmasry,27th April 2018