



**Daffodil**  
*International*  
**University**

GiFaRi: A Secured Data Hiding Technique in LSB based Image  
Steganography Using Blowfish and Edge-based Pixel Selection

Submitted By

**Sk Md Abuzar Gifari**  
**201-35-3024**  
**Dept. of Software Engineering**

Supervised By

**Md. Maruf Hassan**  
**Associate Professor**  
**Department of Software Engineering, FSIT**

A thesis submitted in partial fulfillment of the requirement for the degree of  
Bachelor of Science in Software Engineering

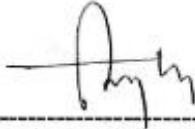
Fall 2023

©All right reserved by Daffodil International University

# APPROVAL

This thesis titled on “GiFaRi: A Secured Data Hiding Technique in LSB based Image Steganography Using Blowfish and Edge-based Pixel Selection”, submitted by **Sk Md Abuzar Gifari (ID: 201-35-3024)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

## BOARD OF EXAMINERS



-----  
**Dr. Engr. Abdul Kader Muhammad Masum**  
**Professor**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

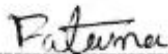
**Chairman**



-----  
**Md Khaled Sohel**  
**Assistant Professor**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Internal Examiner 1**



-----  
**Fatama Binta Rafiq**  
**Lecturer (Sr. Scale)**

Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University

**Internal Examiner 2**



-----  
**Dr. Md. Liakot Ali**  
**Professor**

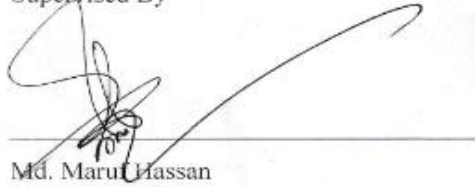
Institute of Information & Communication Technology (IICT)  
Bangladesh University of Engineering and Technology (BUET)

**External Examiner**

# DECLARATION

I, hereby declare that, this thesis report is done by me under the supervision of Mr. Md. Maruf Hassan, Associate Professor, Department of Software Engineering, Daffodil International University, in partial fulfillment my original work. I am also declaring that neither this thesis nor any part therefore has been submitted else here for the award of Bachelor or any degree.

Supervised By



Md. Maruf Hassan

Associate Professor,  
Department of Software Engineering,  
Daffodil International University.

Submitted By

Gifari

Sk Md Abuzar Gifari

ID: 201-35-3024

Department of Software Engineering,  
Daffodil International University.

## ACKNOWLEDGMENT

First of all, I am grateful to the Almighty Allah for making me eligible to complete this thesis. Then I would like to thank my supervisor **Md. Maruf Hassan, Associate Professor, Department of Software Engineering**. I am extremely grateful and indebted to her as she has given me her expert, sincere and valuable guidance and encouragement. I would like to thank everyone who helped me in my thesis by their important suggestion. Without their passionate participation and input, the thesis could not be successfully conducted. I take this occasion to convey my sincere thanks to all faculty members of the Department of Software Engineering for their help and encouragement.

## ABSTRACT

Steganography is a technique that uses cryptography to encrypt secret data, thereby passing confidential data through the transmission channel from the sender to the receiver, even though this secret data is very difficult for an intruder to capture because the data is not easily detectable. This paper, explores an innovative approach to image steganography, integrating the Blowfish encryption algorithm and an Edge-based Pixel Selection technique to enhance the security and imperceptibility of hidden data. The encryption process involves the Blowfish algorithm for robust protection of confidential information, while the Edge-based Pixel Selection method ensures discreet embedding within the image. Blowfish is a very strong algorithm because its key length is not fixed means variable, also it can range from 32 bits to 448 bits. This flexibility allows users to adapt the key size based on their specific security requirements. A longer key generally provides better resistance against brute-force attacks and provides data security. Furthermore, Sobel edge detection can help identify regions of an image that are less sensitive to human perception. Embedding information in regions with existing high gradients (edges) might make the modification less noticeable. The contribution of this work lies in its ability to significantly improve imperceptibility compared to existing methods, providing a strong and elegant method of data hiding in image steganography field.

**Keywords:** Image Steganography, XOR, Blowfish algorithm, Edge-based Pixel Selection technique, LSB (Least Significant Bit)

# Table of Contents

<b>APPROVAL</b> .....	<b>i</b>
<b>DECLARATION</b> .....	<b>ii</b>
<b>ACKNOWLEDGMENT</b> .....	<b>iii</b>
<b>ABSTRACT</b> .....	<b>iv</b>
Chapter-1: Introduction.....	1
1.1 Background .....	1
1.2 Fundamental requirement for steganography.....	2
1.3 Problem Statement .....	3
1.4 Research Objective.....	3
1.5 Scope of works .....	5
1.6 Contributions.....	5
1.7 Solution Requirement.....	5
1.8 Thesis Outline .....	6
Chapter-2: Literature Review.....	7
2.1 Commencement of this Study .....	7
2.2 The history of image Steganography .....	8
2.3 The evaluation of image Steganography over time.....	9
2.4 Application of image Steganography .....	10
2.5 Research Gap.....	13
2.6 Research Objective.....	14
2.7 Closure of this study.....	15
Chapter-3: Methodology.....	17
3.1 How Image Steganography Works.....	17
3.2 Proposed Method.....	18
Chapter-4: Result Analysis and Discussion .....	29
Chapter-5: Conclusion .....	36

## Table of Figures

Figure 3.1: Image Steganography Procedure.....	17
Figure 3.2: Embedding System of the Proposed Approach .....	18
Figure 3.3: Extracting System of the Proposed Approach.....	19
Figure 3.4: Working Procedure of The Proposed Model .....	20
Figure 3.5: Blowfish Encryption and Decryption Process.....	23
Figure 3.6: XOR Embedding Process.....	24
Figure 3.7: Detect Co-ordinates from the edges based on Sobel Edge-Detection Algorithm .....	26
Figure 4.1: Matlab Code Implementation.....	32
Figure 4.2: Proposed model code implementation .....	33

## Lists of Table

Table 2.1: Literature Review Table.....	16
Table 4.3: Comparison PSNR of our proposal with other researchers .....	34
Table 4.4: Comparison MSE of our proposal with other researchers.....	34



# Chapter-1: Introduction

## 1.1 Background

The process of sending messages from a sender to a recipient securely it's known as Steganography. It ought to establish that no one can draw firm conclusions about the sender and recipient's covert communication. To maintain this level of secrecy, the secret message is concealed within some cover media, making it unlikely that a third party could discover it. Cryptography is used to account for secure communication and data capacity across unreliable organizations. Different strategies are developed to deal with particular problems; one well-known key strategy is the use of information encryption to obtain secret data from organizations. Because of the increased demand for encryption techniques, Blowfish is not the only one that has become more well-known [1]

Modern fields like business, medicine, the military, and other fields have greatly benefited from the increased amount of information being sent in visual correspondences due to technological advancements. Thus, if sensitive information is to be transferred via correspondence channels, creating a protected association is now imperative. Data handling and access by unauthorized individuals can be prevented with the help of this significance, it was found [2]. Proposed model additionally set up an edge-based pixel selection mechanism to improve the imperceptibility of the hidden information. Proposed model aims to achieve a delicate balance between installing limit and perceptual forthrightness by deliberately selecting pixels along picture edges, where visual changes are less likely to be acknowledged. This lowers the possibility of uncertainty by ensuring that the host picture maintains its visual fidelity while also enhancing the steganographic cycle's security.

The need for safe communication techniques is growing as the digital world develops. Proposed model combines well-established cryptographic ideas with cutting-edge pixel selection techniques to advance the state of LSB-based image steganography. This thesis investigates the creation and application of the Proposed model, shedding light on its efficacy, security, and possible uses in protecting sensitive data in the constantly growing digital sphere. Proposed model is not only highly proficient in cryptography, but it also includes a pixel selection mechanism that is

strategically based on image edges. Due to embedded information, traditional LSB-based steganographic techniques frequently have to choose between hiding capacity and perceptual transparency, which could jeopardize the visual integrity of the host image. In order to get around this problem, this model selects pixels together on the edges of images, where the human eye is less likely to notice changes. This new approach aims to strike a suitable balance between minimizing any discernible effects on the host image's appearance and permitting information to remain hidden within it.

The Proposed model is important because of its potential influence on secure communication practices in addition to its technical complexity. The Proposed model offers a timely response to the growing demands of information security as the digital world develops and adversaries grow more cunning. The goal of this thesis is to shed light on the design, implementation, and performance evaluation of the Proposed model in order to advance knowledge of LSB-based image steganography and its uses for protecting confidential data in the ever-changing digital environment. [1]

## **1.2 Fundamental requirement for steganography**

A successful steganographic technique requires three essential elements: imperceptibility, capacity, and robustness. Being imperceptible is among them. Steganography's primary prerequisite, imperceptibility, emphasizes the hiding of secret data inside a carrier medium without causing appreciable changes. Imperceptibility in image steganography guarantees that the embedded data does not visually differ from the original content. In order to minimize the chance of detection, a high degree of imperceptibility requires carefully choosing embedding locations, such as in the least important bits or areas less sensitive to human vision. The quantity of data that a steganographic method can encode into a particular carrier is measured by its capacity. Capacity is concerned with efficiency, whereas imperceptibility is seeking subtlety. It's critical to strike a balance between imperceptibility and a significant hiding capacity. Capacity optimization techniques frequently take advantage of redundant or less important carrier components, enabling more data hiding without sacrificing the overall visual integrity. Increased capacity is particularly important for situations where a significant amount of hidden data needs to be transmitted. Robustness is the capacity of a steganographic technique to fend off detection and attacks while preserving the confidentiality of the data that is being concealed. Imperceptibility is improved by

integrating cryptographic algorithms, as demonstrated by the Proposed model's application of the Blowfish algorithm. This cryptographic layer strengthens the concealed data against attempts at tampering and unauthorized access, enhancing the steganographic technique's overall resilience. For applications where the hidden information needs to be shielded from adversarial scrutiny or inadvertent distortions during transmission, robust steganography is crucial.

### **1.3 Problem Statement**

The Problem Statements PS1 and PS2 underscores the critical issues within existing image steganography techniques, prompting the need for a comprehensive solution. PS1 highlights how vulnerable existing techniques are because they don't have strong security safeguards, leaving hidden data open to illegal extraction and discovery. The privacy of hidden data is seriously threatened by this. However, PS2 draws attention to the difficulties that commonly used LSB-based image steganography techniques face, particularly in achieving the best possible imperceptibility. Here, the possible introduction of observable artifacts that jeopardize the embedded information's secrecy is the cause for concern.

**PS1:** Existing image steganography techniques lack robust security measures, making them susceptible to unauthorized data extraction, and detection. This raises concerns about the confidentiality of hidden information. [15]

**PS2:** The widely used LSB-based image steganography methods face challenges in achieving optimal imperceptibility leading to noticeable artifacts. [16]

### **1.4 Research Objective**

The research goals (RO1 and RO2) are developed in response to these interrelated problems. In order to solve the security flaws found in PS1, RO1 suggests and puts into practice a novel model that greatly improves data security when it comes to image steganography. This entails creating cutting-edge strategies to strengthen the concealment procedure against detection and unwanted access.

Concurrently, RO2 is intended to address the imperceptibility problems mentioned in PS2. The goal is to create and refine a spatial steganographic algorithm, primarily concentrating on increasing imperceptibility. This suggests a methodical investigation of methods to incorporate

data into pictures with the least amount of obvious artifact introduction, so maintaining the secrecy and undetectable nature of the hidden data.

**RO1:** To prepare a new model that enhances data security for image steganography.

**RO2:** To design and develop a technique which mainly focus on improving imperceptibility with better image quality.

Mapping:

1. RO1 → PS1
2. RO2 → PS2

The problem statement (PS1), which emphasizes how vulnerable current image steganography methods are to unauthorized data extraction and detection, endangering the confidentiality of hidden information, is addressed by this mapping. In order to improve data security for image steganography, a new model that accomplishes this goal is being prepared (RO1). The main objective is to create a strong and novel steganographic technique that addresses the security flaws in the methods that are currently in use. To strengthen the steganographic process and guarantee that the hidden information is kept private and safe from unwanted access, this may entail incorporating sophisticated encryption or authentication techniques.

The second problem statement (PS2), which highlights the difficulties commonly encountered by LSB-based image steganography techniques in reaching optimal imperceptibility and frequently resulting in observable artifacts in the output, is addressed by this mapping. In the spatial domain, the corresponding research objective (RO2) seeks to design and develop a steganographic algorithm with an emphasis on enhancing imperceptibility. To reduce the likelihood of detection, creating an algorithm to blend hidden data into a picture so the changes are not obvious to the naked eye is the task at hand. The overall visual quality of the steganographic image can be improved by using strategies like edge-based pixel selection, which guarantees that the hidden information is well-concealed and does not jeopardize the integrity of the carrier image.

## 1.5 Scope of works

This paper will only address image steganography with the solution I will offer. It is not applicable to audio or video steganography. Right now, we are working on this paper on a small scale. I would advise future research on this topic to make use of audio or video.

## 1.6 Contributions

Contribution of this paper is given as follows:

- This approach is also used LSB embedding approach where it is embedded two bits into a single pixel that provides good image quality than other existing techniques.

## 1.7 Solution Requirement

Let us talk about the Data Embedding Process. Phase 1: Blowfish's Secret Data Encryption: The Blowfish algorithm is used to encrypt the secret data during this first stage. Symmetric key block cipher Blowfish is renowned for its effective and safe encryption. The algorithm generates a ciphertext that will be embedded in the cover image using the secret message and secret key. In Phase 2: Particular Picture: One cover image is selected specifically for the embedding procedure. The encrypted secret data is carried by this image. Phase 3: Edge-based Pixel Selection for Image Element Selection: Edge-based pixel selection is used to carefully select image elements in order to improve the hiddenness of the embedded data. This involves identifying and selecting pixels along edges within the image, as edges are less noticeable to the human eye. Phase 4: Data Embedding: In this stage, the chosen picture elements are embedded with the encrypted secret data. The method is to embed the encrypted data into the selected pixels, making sure that the changes are undetectable so as to preserve the cover image's aesthetic integrity. In Phase 5: Retrieving Metadata: When we talk about the data retrieval process, then we should extract metadata from the data which is embedded. This metadata gives important details about the content that is embedded. Phase 6: Selecting Retracted Image Elements: Certain image elements are chosen, just like in the embedding process; The object of this step is to decline the previously embedded data. The system uses the metadata that was obtained in the previous stage to inform its selection, which helps it find and identify the precise pixels that have the embedded data. Phase 7: Data Extraction: In this phase, the embedded data is extracted by processing the identified image elements. In this

stage, the hidden data must be meticulously extracted from the cover image without introducing any noticeable artifacts. Phase 8: Decrypt the Data That Was Retrieved The last step is to use the right key to decrypt the data that has been recovered using the Blowfish algorithm.

Image quality is the primary objective of the steganographic system. Most famous metrics to assess image quality are Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) [3]. One metric to assess how much the embedding image has degraded in comparison to the cover image is PSNR.

$$\mathbf{MSE} = \frac{\mathbf{1}}{\mathbf{a} * \mathbf{b}} \sum_{\mathbf{i}=0}^{\mathbf{a}-1} \sum_{\mathbf{j}=0}^{\mathbf{b}-1} [\mathbf{X}(\mathbf{i}, \mathbf{j}) - \mathbf{Y}(\mathbf{i}, \mathbf{j})]^2 \dots \dots \dots (\mathbf{1.1})$$

$$\mathbf{PSNR} = \mathbf{10} \log_{\mathbf{10}} \frac{\mathbf{MAX}_x^2}{\mathbf{MSE}} \dots \dots \dots (\mathbf{1.2})$$

The difference between two images is measured by MSE. Equations 1 and 2 define PSNR and MSE. The dynamic range of pixel values, or the maximum value that a pixel can take for 8-bit images, is I=255. Where are the row and column pixels in the original (cover) image, the row and column pixels in the reconstructed (Stego) image, and the height and width of the image.

## 1.8 Thesis Outline

The data being presented emphasizes how important secure communication is in the digital age and how sensitive data can be protected with LSB-based image steganography. It offers an incredible steganographic technique that boosts imperceptibility by combining cryptographic principles with random pixel selection. Three essential components are necessary for steganography to be successful: robustness, capacity, and imperceptibility. Proposed model integrates the Blowfish algorithm and makes deliberate pixel selections to address these issues. The problem statement draws attention to the drawbacks of current steganographic methods as well as the challenges faced by LSB-based strategies in maintaining optimal imperceptibility. Modern cryptographic techniques are the main tool used in the suggested solution to improve data security for image steganography, particularly in the spatial domain.

## Chapter-2: Literature Review

### 2.1 Commencement of this Study

Image steganography is the process of hiding sensitive information inside an image so that it is difficult to decipher. While cryptography focuses on making a message unreadable for unauthorized users, Steganography seeks to hide the embedded data—not the content—from prying eyes. Image steganography embeds secret data into an image's pixel structure. Changing the pixel values' least significant bits (LSBs) is the most popular method. Little variations in these bits are less likely to be detected by the human eye because the LSBs contribute less to the perception of color or intensity overall. Digital watermarking, copyright protection, and secure communication are just a few of the uses for image steganography. It's crucial to remember that steganography is not infallible, and detection techniques have been created to recognize photos that have been tampered with during the steganography process. In a never-ending game of cat and mouse, researchers strive to continually improve both steganographic techniques and detection methods. Two key components of steganography are the type of carrier and the embedding technique. Several carriers are used as the cover message in steganography. Image steganography is a commonly used carrier medium because images are the most frequently transferable messages over the Internet. The field of image steganography has seen a great deal of research. There are essentially two categories of image steganography. [1], [2]:

**1) Spatial domain:** With this method, information will directly embed in the pixel intensity values. It is specifically utilized for lossless compressed images. because the format of the image affects embedding. The replacement of least significant bits (LSB) is a widely used technique in the spatial domain.

**2) Transform domain:** With this technique, data is added to an image that has already undergone transformation in the frequency domain. To hide data in an image, various transformation techniques such as the Discrete Cosine Transform are employed. JPEG images that have undergone lossy compression can be used with this method.

## 2.2 The history of image Steganography

As the idea of concealing information has existed since antiquity, so too does the history of image steganography. Although the current state of image steganography has developed along with technological advancements, the fundamental ideas are rooted in history. This is a quick synopsis of image steganography's past: According to historical accounts, the Greeks employed a method known as "scytale" to conceal messages. Written on a parchment strip that was coiled around a rod with a specific diameter was the message. The message would be jumbled and appear meaningless when unwound. With a rod the same diameter, the recipient could decipher the message. Different types of invisible ink have been used to hide messages throughout history. These inks might come to light by means of heat, chemicals, or other techniques. Steganography was employed by both the Allied and Axis forces in World War II. To conceal information, for instance, tiny photos called microdots—the size of a punctuation mark—were employed. Throughout the Cold War, steganography was still used to encrypt letters, radio broadcasts, and other types of correspondence. As the use of computers increased, steganography moved into the digital realm. In the past, methods included data hiding in the least important parts of audio and digital image files. As digital steganography grew in prominence, scientists started creating increasingly complex techniques. During this time, the well-known image steganography algorithm F5 was released. Steganography became more widely available as digital communication and the internet grew in popularity. Different algorithms were created by researchers to conceal data in audio, video, and image files. As steganography tools and techniques developed, researchers looked into new ways to embed data while causing the least amount of damage to the carrier file.

- Applications: Digital watermarking, authentication, and secure communication are three areas where image steganography is used. It evolved into a tool for defending intellectual property and confirming the legitimacy of online content. Steganography gained popularity along with the development of steganalysis, or techniques for uncovering hidden data. As a result, there has been a continuous arms race between stenographers and people looking for hidden content. In conclusion, the history of image steganography is extensive and extends from prehistoric times to the digital era. The dynamic evolution of communication technologies and the continuous pursuit of secure and surreptitious information exchange have propelled its development. The effectiveness of steganographic techniques, the advancement of detection techniques



(steganalysis), and overall security are all evaluated throughout the course of image steganography evaluation.

### **2.3 The evaluation of image Steganography over time**

Here's an overview of how the evaluation of image steganography has evolved. Early research on digital steganography concentrated on fundamental methods such as LSB (Least Significant Bit) embedding. These techniques lacked resilience and security but were straightforward and somewhat covert. The ability to conceal data without raising red flags and the imperceptibility of the steganographic modifications—that is, how visually identical the Stego-image is to the original—were among the evaluation criteria. The introduction of sophisticated steganographic algorithms like OutGuess and F5. The goals of these algorithms were to increase security and robustness while addressing the shortcomings of the earlier techniques. Metrics like resistance to steganalysis, payload capacity (the amount of data that could be hidden), and the capacity to survive standard image processing operations without losing the hidden information were taken into consideration by researchers. The focus shifted to strengthening steganographic algorithm security as steganalysis techniques advanced. This required creating defenses against steganalysis based on statistics and machine learning. More sophisticated embedding strategies, such as spatial domain techniques, transform domain techniques (like frequency domain techniques), and adaptive methods that change based on the properties of the cover image, were investigated by researchers. Researchers looked into combining cryptography and steganography to improve security. The goal of this combination—known as steganographic encryption—was to offer secret communication in addition to secrecy. The resistance to attacks like chosen-plaintext attacks, known-plaintext attacks, and adaptive steganalysis was added to the evaluation criteria. Steganalysis and stenographers are still engaged in an arms race. The problem for researchers is to create steganographic techniques that work and can withstand ever-more-advanced detection techniques. Current investigations investigate novel frameworks, like deep learning-driven steganography and steganalysis, in order to overcome the drawbacks of conventional methods. Steganographic techniques that can function in various and difficult environments are required for real-world applications like digital forensics and secure communication.

## 2.4 Application of image Steganography

Overall, from simple imperceptibility to more intricate considerations of security, robustness, and integration with cryptographic techniques, the evaluation of image steganography has progressed. With an emphasis on offering safe and discrete means of information exchange, the field keeps up with technological advancements and adapts to new opportunities and challenges.

Secure communication is essential for military and defense applications. Your method can be used to secretly transmit classified data by embedding sensitive information into images. You can use your steganographic technique to watermark digital files, which enables digital media tracking and authentication. This is especially helpful in preventing illegal distribution and copying. Media and entertainment sectors can utilize your technique to incorporate ownership information or copyright information into digital photos, videos, or audio files to stop illegal use or distribution. It is critical to guarantee patient data confidentiality in medical imaging. Your steganographic method can be used to conceal private patient data from view while maintaining the accuracy of the diagnostic information. You can safely transmit private business or legal documents using your method. An additional degree of security is added to data transfer when data is embedded within images. Safe Information Transmission for Journalists: Without drawing notice, journalists and whistleblowers can use your technique to safely transfer sensitive data or evidence. Your method of embedding information within images for secure communication and collaboration can be used by researchers and academics working on sensitive projects or collaborative research. You can offer your private messaging system to people who are concerned about their privacy. An extra layer of concealment is added when text is added to images. As part of a DRM strategy, content providers can embed data into images using your steganographic technique to stop illegal access or distribution of digital content. You can discuss your method in relation to anti-forensic techniques, which are ways in which people try to conceal information from digital forensic examination.

The author put forth a novel approach in [4]. This method conceals the secret message by searching for identical bits between the image pixel value and the secret message. One randomly chosen pixel separates the image into three layers (Red, Green, and Blue). Next, by searching for similar bits, two bits of the secret message are embedded in each layer's two least significant bits.

A spatial domain method is proposed in [4]. The proposed method uses the third least significant bit (LSB-3) of the cover image to embed the message bits with the goal of minimizing the difference between the cover and the stego-cover. Then, depending on the message bits, LSB -1

and 2 can be changed. For extra security, the message bits have been permuted using a stego-key prior to embedding. However, the results of the method showed that the LSB -1 method had higher PSNR values than the recommended method, meaning that even though the capacity was the same, the LSB -1 image was of higher quality than the modified one.

An image steganography technique based on LSB substitution and random pixel selection within the required image area has been proposed by the author [5]. It chooses the area of interest where after generating random numbers, the required message is embedded along the random pixels. The goal of this technique is to increase security in cases where the password is added using LSB pixels.

Using a key for both data encryption and decryption, the Blowfish Encryption Algorithm is a straightforward, quick, and small encryption method. It takes sixteen rounds, with an XOR operation and a function (F) in each round. The Blowfish algorithm is used in applications such as file encryptors and communication links where a key is not required because it does not change the key. However, Blowfish is inefficient when it comes to using packet switching as a one-way hash function or supporting frequent changes to the key. The algorithm consists of a feistel network that enables encrypted data encryption and key expansion. [6]. Using a variable key size of 448 bits, the algorithm outlined by [7] divides the image data into blocks for encryption. The modified Blowfish algorithm is a remarkable standard encryption algorithm because, by increasing the number of rounds, it yields more efficient results than the previous algorithm. The algorithm works more securely and effectively than symmetric encryption algorithms like AES and DES because it can use a variable length key. The enhanced Blowfish algorithm is more secure than the original Blowfish algorithm because it jumbles data using an additional block switching technique. On picture pixels, the improved Blowfish algorithm produces random numbers.

With image steganography, words and images can be concealed throughout images. Discrete Cosine Transform, Transform Domain, Spread Spectrum, Filtering and noise, Concealment, MSB, and LSB are a few of the methods applied here (Kamble et al., 2013). Because the MSB approach gives the steganographic image a suspicious appearance to human beings, it is not an optimal alternative for steganography systems. Generally speaking, there are two types of steganographic techniques: spatial and frequency domain [8]. This method distributes the secret message across each color plane using a predetermined secret key. A modified PVD-based steganography method was proposed by Nagaraj et al. [9]. The pixel value difference technique

(PVD) is an embedding method that was proposed in [3]. With this technique, data is embedded onto each pixel after the image is divided among randomly chosen non-overlapping blocks of nearby pixels. The amount of data embedded, or the number of least significant bits used, has a direct connection to the variance in brightness of adjacent pixels. Unusual steps appear in the stego image's pixel difference histogram as a result of this uneven embedding in PVD. An enhanced method (IPVD), suggested in [10], has taken advantage of this weakness. By adding a readjusting phase, the adaptive edge LSB technique (AE-LSB) [11] has also eliminated this uneven pixel difference and improved capacity. All of these methods have one basic drawback, but they are edge adaptive in the sense that they embed so much data in areas with significant pixel differences. Rather than choosing based on greater differences, these methods take a random look at pixel pairs. As a result, they might wind up distorting the texture in the image's LSB plane and embedding data at random locations throughout. It is discovered that these techniques perform poorly [12].

The author of "Information Hiding Using Edge Boundaries of Objects" [13] focuses more on the difference across the border pixel of the stego image and the cover image. Several techniques are applied to the stego object to lessen this disparity, allowing the stego image to be utilized as the original image for additional processing at the receiving end. Using a clever edge detector, the edges are first found. Next, the absolute difference between the edge pixel and its upper edge pixel is calculated. The LSB approach is used in the embedding process if the difference is smaller than the threshold value. Once more, the edges of a stego picture are identified using an ingenious edge detector. If there is a difference between the edges, the threshold value is changed. This technique is continued until the margins of the stego picture and the cover image differ. This method is safer and has less computing overhead.

The author of "Steganography in images using Sobel Edge Detection with 2k Correction Method" [14] suggested an edge-based image steganography technique that makes use of the 2k correction method and the Sobel Edge Detector. Because the cover image and stego image differ, the 2k method is used to provide better imperceptibility. The Stego Pixel Value (SPV) is changed by  $SPV-2k$  or  $SPV+2k$  if the difference between the actual and SPV values is greater than  $2k-1$ . In an experiment, the Mean Square Error (MSE) and PSNR are computed. Compared to the LSB approach, this method offers higher embedding capacity and PSNR.

The proposed image steganography technique employs a novel approach to achieve high imperceptibility within the LSB-based domain. The technique minimizes visual changes in the

host image by carefully integrating information employing XOR-based embedding to translate pixel values into their least significant bits (LSBs). By introducing minute modifications that are hard for the human eye to notice, XOR operations improve imperceptibility while maintaining the cover image's visual integrity. Additionally, adding the Blowfish algorithm during the embedding process adds an additional layer of security. To safeguard the privacy of the embedded data and stop unauthorized access, Blowfish makes sure that the hidden data is encrypted. Because of this encryption, changes made to the LSBs become less noticeable from fluctuations in pixel values that occur naturally. Using the visual properties of edges—where minute changes are less likely to be noticed—the embedding of edge-based pixels improves imperceptibility. This deliberate pixel selection successfully conceals information within image edges while minimizing the effect on smooth regions. To summarise, the model achieves high imperceptibility by means of selective pixel embedding in edge regions, encryption using the Blowfish algorithm, and strategic LSB-based XOR embedding. These techniques work together to ensure that the steganographic alterations remain undetectable to observers.

## **2.5 Research Gap**

The problem statement highlights a critical issue in the field of image steganography. It implies that current steganographic strategies are vulnerable to unauthorized extraction and location of covered information due to the absence of robust safety measures affecting them. The ramifications include the potential for private information concealed in photos to be jeopardized as a result of flaws in existing protocols. In this case, the identification of the steganographic interaction itself raises potential concerns about data confidentiality in addition to the expected disclosure of hidden information by unauthorized parties. An alternative explanation centers on a particular method within image steganography, specifically approaches that utilize the Least Critical Piece (LSB). It claims that with commonly used LSB-based steganography, achieving optimal imperceptibility is challenging. Subtlety is crucial to steganography because it guarantees that any changes made to the cover image (or implant stowed-away data) are invisible from the exterior. That's what the claim implies: the steganographic images exhibit observable artifacts as a result of the difficulty in maintaining subtlety brought about by LSB-based techniques. These old changes might have been inadvertent, raising questions about the procedure and possibly disclosing private data, which would undermine the steganographic cycle's effectiveness.

## 2.6 Research Objective

The objective is to solve the weak security features of the currently used image steganography techniques, which make them susceptible to detection and unauthorized data extraction. This entails creating a new model for image steganography that enhances data security. This underscores how inadequate the security measures in the currently in use image steganography techniques are and raises concerns about the privacy of hidden data. Here, the objective is to develop a new model that significantly enhances the data security features of image steganography. This might entail incorporating cutting-edge cryptographic methods, like the Blowfish algorithm that you mentioned in the title of your thesis. Strengthening steganography and making it more resilient to unauthorized data extraction and detection should be the main objectives of the new model. By doing this, the confidentiality of the data concealed within the steganographic images will be improved, assuaging concerns about possible security holes in current techniques. Commonly used LSB-based image steganography techniques have trouble reaching optimal imperceptibility, which causes observable artifacts. This emphasizes the necessity of creating a steganographic algorithm in the spatial domain that is in line with the techniques and focuses on enhancing imperceptibility. The objective here is to design and develop a steganographic algorithm for the spatial domain. This algorithm should be specifically created to ensure that any modifications made during the data hiding process are practically invisible to the unaided eye in order to maximize imperceptibility. Using techniques like edge-based pixel selection, which lessens perceptible artifacts and helps guarantee that hidden information is seamlessly integrated into the carrier image, can improve the overall visual quality of steganographic images.

## 2.7 Closure of this study

For the purpose of developing the proposed thesis paper, "GiFaRi: A secured data hiding technique in LSB-based image steganography (XOR) using Blowfish algorithm and edge-based pixel selection," a thorough review of the body of literature on image steganography has been conducted. Critical insights into the state of steganographic techniques today have been uncovered by the literature review, which highlights the need for improved data security and imperceptibility in the spatial domain. The weaknesses in conventional LSB-based steganography techniques have been repeatedly brought to light by the reviewed studies, highlighting the significance of creating new strategies to overcome these drawbacks. Combining the XOR process with the Blowfish algorithm shows promise in strengthening the security of data that is hidden in image files. The suggested Proposed model's imperceptibility is further enhanced by the innovative integration of edge-based pixel selection, which ensures a careful balance between data hiding and visual fidelity. The literature review has emphasized how important it is to develop steganographic techniques in order to protect against contemporary threats and changing detection techniques. This review has positioned the Proposed model as an innovative solution that not only improves data security but also focuses on improving imperceptibility within the spatial domain by synthesizing insights from diverse sources. The literature review has prepared the groundwork for a thorough examination of the complexities of XOR-based LSB steganography, the stability of the Blowfish algorithm, and the effectiveness of edge-based pixel selection as we proceed with the Proposed model thesis. The integrated information extracted from the literature review acts as an invaluable roadmap, directing the investigation toward the creation of a state-of-the-art steganographic methodology. Building on this foundation, the following chapters of the thesis will attempt to make new contributions to the field of image steganography and lay the framework for future developments in safe and undetectable data hiding methods.

Researcher Name	Year	Technique	Capacity	Limitation	PSNR
Pratiksha Sethi	2018	LSB based image steganography	N/A	Improving algorithm efficiency for larger data while enhancing cipher strength against brute force attacks.	73%
Kamaldeep Joshi	2020	The mixture of the YCbCr Color Model, 2-bit XOR LSB substitution in Cr, and crypto-algorithm.	30.75 kb	Implement Bluefish algorithm can improve image PSNR value.	68%
Mukesh Dalal	2021	Spatial domain-based embedding, LSB, DCT	27.53%	Embed secret data in a specific image frame region for discreet concealment.	37%
Sanjay Misra	2022	Modified LSB Steganography	N/A	Implement Higher Order bits and Error-diffusion technique to improve more Capacity in image.	68%
Sami Ghoul	2023	randomization, encryption and region-based	6300 kb	applying an additional layer of protection by making use of the current encryption techniques.	52.74-58.10%
S. M. Ammar Alam	2023	XOR-based data hiding in 2-LSBs	N/A	A parity-based embedding approach could enhance the quality of the image.	N/A
Moon et al.	(2018)	4LSB	12.5%	More prone to attacks	N/A
Kaur et al	(2019)	Hash-LSB	100%	A single image was utilized for the testing, and it was not tamper-resistant.	74.18

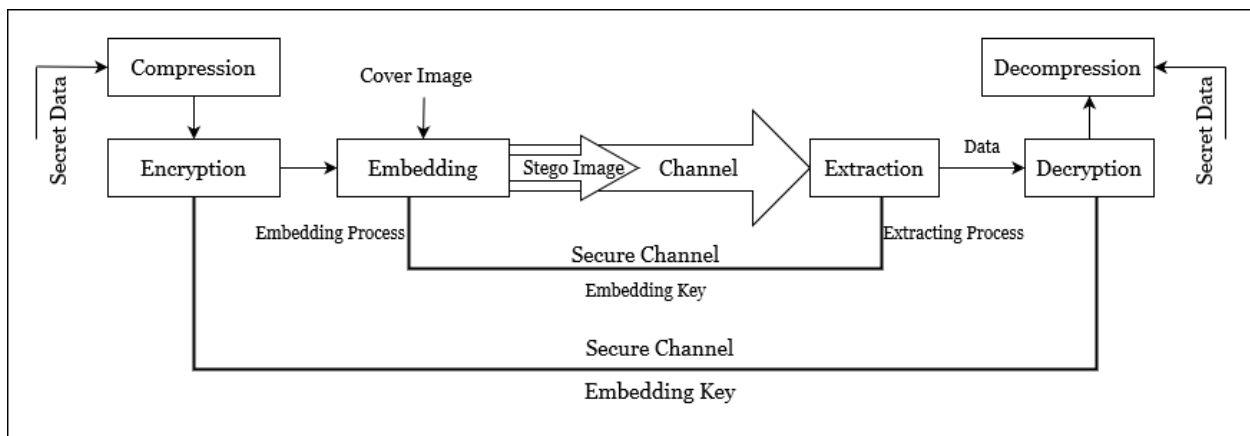
**Table 2.1: Literature Review Table**



## Chapter-3: Methodology

### 3.1 How Image Steganography Works

Actually Image, audio, video, text, and network steganography are the five categories into which steganography can be separated as a result of the cover media. Using an image as the cover object is how secret messages are hidden in image steganography. The secret data is concealed in this process by using the cover image's pixel intensities. A wide range of techniques, mostly classified as spatial domain and transform domain techniques, are available for embedding data into the cover image. The methods involved, which fall primarily into the spatial domain and transform domain categories, involve the following: LSB Substitution, Pseudorandom Method, Discrete Cosine Switching Technique (DCST), Discrete Wavelet Transformation Method (DWT), etc.

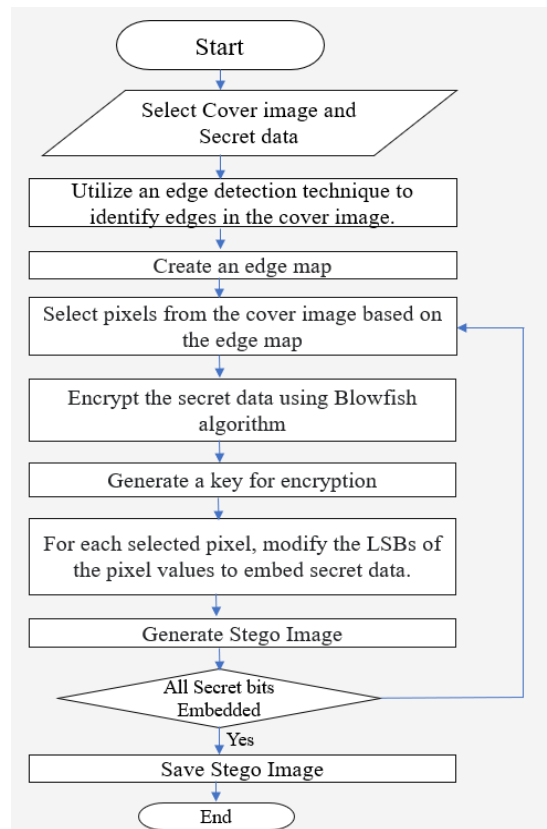


**Figure 3.1: Image Steganography Procedure**

Figure 2 depicts the general steganographic scheme. The concealed data is inserted into the cover image utilizing the appropriate technique and stego key in the embedding system. The recipient then obtains the stego image over the communication channel and extracts the message from its stego image using the stego key and the used procedure. There are two main prerequisites for image-based steganography that scientists believe are critical to the concealing process. First, parts of a secret message can be hidden in an image using steganography so that the original and the stego-image are visually identical to one another; Stated differently, the covert communication is undetectable. Second, the technique should be able to include a sizable quantity of hidden data in

the cover image without sacrificing imperceptibility. The link between these two criteria needs to be balanced, hence the digital steganography technique's parameters should be carefully selected.

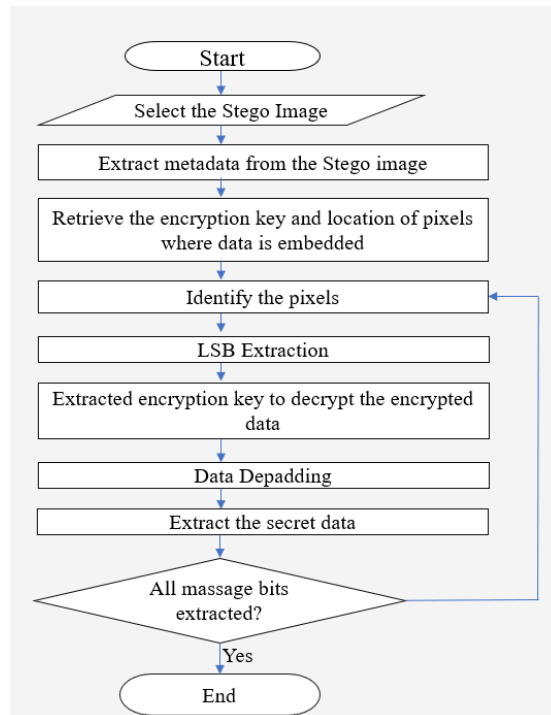
### 3.2 Proposed Method



**Figure 3.2: Embedding System of the Proposed Approach**

In the embedding process of the technique, the procedure initiates with the selection of a cover image and secret data for concealment. Subsequently, an edge detection technique is applied to the chosen cover image to identify and delineate edges, forming an edge map. This edge map serves as a guide for pixel selection in the cover image, ensuring that embedding occurs predominantly in areas characterized by edges. Following this, the secret data undergoes encryption using the Blowfish algorithm, and a unique key is generated for the encryption process. The selected pixels from the cover image, determined by the edge map, have their least significant bits (LSBs) modified to accommodate the encrypted secret data. This alteration in the LSBs ensures that the

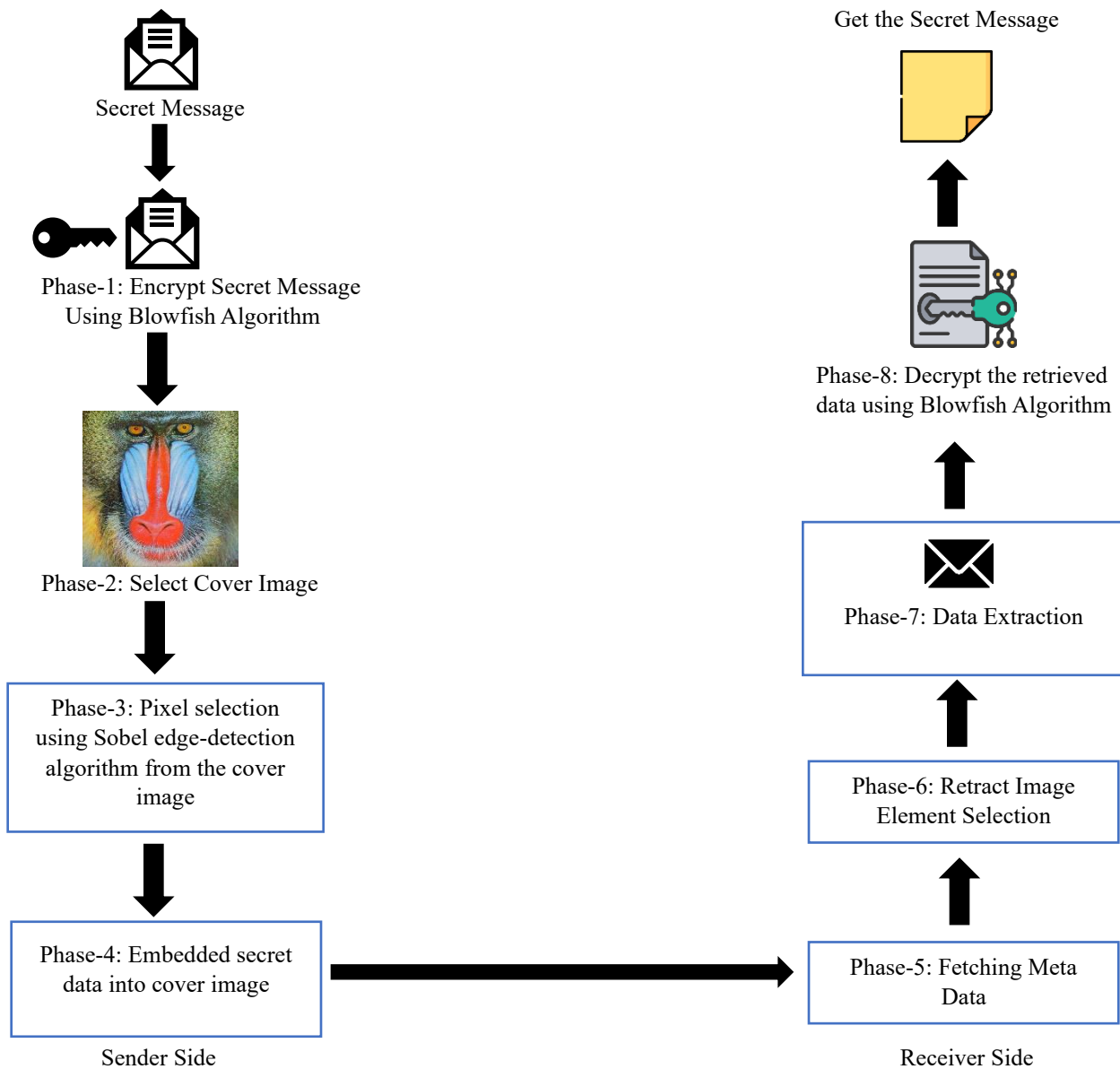
changes are imperceptible to the human eye, preserving the visual integrity of the cover image. The embedding process iterates through each selected pixel until all secret bits are successfully embedded. Upon completion, the stego image is generated. A check is performed to ascertain whether all secret bits have been embedded; if affirmative, the stego image is saved as the final output. In the event of negative confirmation, the process returns to pixel selection based on the edge map, continuing the embedding until completion. The entire embedding process concludes when all secret data is successfully hidden in the cover image, resulting in the creation of the stego image.



**Figure 3.3: Extracting System of the Proposed Approach**

In the extraction process of the technique, the procedure begins by selecting the stego image containing the hidden information. Metadata is then extracted from the stego image, providing essential details for subsequent steps. The encryption key and the pixel locations where data is embedded are retrieved from the metadata. Using this information, the specific pixels are identified in the stego image. The next step involves LSB extraction, wherein the least significant bits of the identified pixels are extracted. The extracted LSBs are then used in conjunction with the retrieved encryption key to decrypt the data which is embedded in the image which is stego. Following decryption, a depadding process is applied to remove any padding that may have been added during

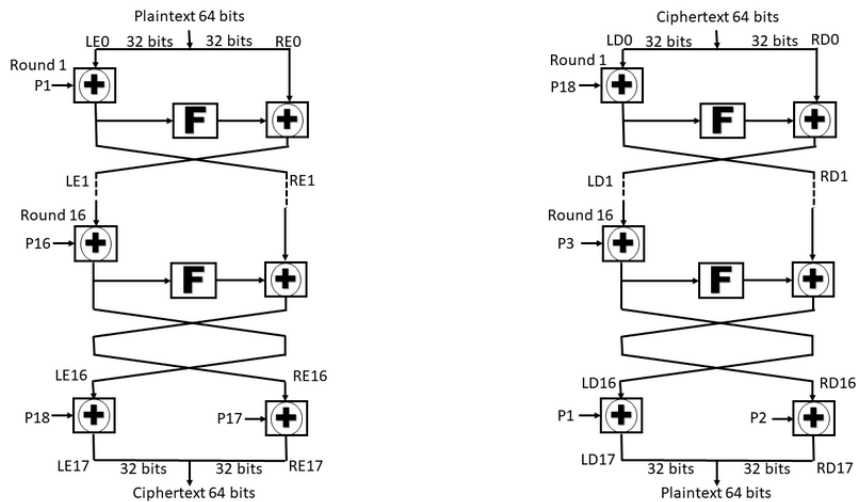
encryption. The final step involves the extraction of the secret data from the depadded information. A check is performed to determine whether all message bits have been successfully extracted, and if affirmative, the extraction process concludes. In the event of negative confirmation, the process returns to the identification of pixels, initiating a loop until all message bits are successfully extracted. The entire extraction process concludes once all secret data has been retrieved from the stego image, marking the end of the process.



**Figure 3.4: Working Procedure of The Proposed Model**

I will now talk about the overall process of my methodology. Certainly, I'll provide a brief overview of each phase in the process of hiding and extracting data using the GiFaRi technique. In the Phase 1, The secret message "Gifari" is encrypted using the Blowfish algorithm to enhance security. In Phase 2, The cover image "leena.png" is chosen as the carrier for the secret message. In the Phase 3: Image Element Selection by Edge-based Pixel Selection Technique. Employing the Edge-based Pixel Selection technique, specific pixels in the image are chosen to embed the encrypted data. Edges are preferred for data hiding to maintain imperceptibility. In the Phase 4, XOR (Least Significant Bit) is used to embed the encrypted data into the chosen image pixels. This involves replacing the least significant bits of the selected pixels with the corresponding bits from the encrypted message. In the Phase 5, The metadata related to the embedded data, such as the location and nature of the embedded pixels, is retrieved to facilitate the extraction process. In the Phase 6, The same Edge-based Pixel Selection technique is applied to identify and select the pixels that were originally chosen for embedding during the data hiding process. In the Phase 7, the data is extracted from the selected pixels using the XOR LSB method. To reconstruct the encrypted message, this entails removing the least important bits from the selected pixels. In the Phase 8, The extracted and encrypted data is decrypted using the Blowfish algorithm, reversing the encryption applied during the embedding phase. This yields the original secret message "Gifari." This technique combines the security of the Blowfish algorithm with the imperceptibility provided by the Edge-based Pixel Selection technique in LSB-based steganography. It ensures that the secret message is embedded into specific pixels of the chosen image, utilizing the least significant bits for minimal visual impact. During extraction, the process is reversed, retrieving metadata, selecting embedded pixels, extracting data, and decrypting it to obtain the original secret message. This comprehensive approach aims to enhance the security and imperceptibility of data hiding in image steganography, making it suitable for applications where confidentiality and stealth are crucial. The combination of Blowfish encryption and Edge-based Pixel Selection contributes to a robust and effective data hiding technique. The imperceptibility in this technique is enhanced through the combined use of XOR LSB, the Blowfish algorithm for encryption, and the Sobel edge detection algorithm for edge-based pixel selection. Let's examine how each of these components contributes to increased imperceptibility. XOR LSB is a common steganographic

method where the least significant bits of pixel values are modified to embed information. It is chosen for its stealthiness, as small changes in the least significant bits are less likely to be visually noticeable. The secret message is first encrypted using the powerful symmetric-key encryption method Blowfish before it is embedded. By doing this, it is made sure that the information is securely implanted and that only the right person can access it. The encryption adds a layer of security, making it challenging for unauthorized parties to decipher the hidden message. Sobel edge detection identifies regions with significant changes in pixel intensity, which often corresponds to edges in an image. By focusing on these areas during pixel selection, changes introduced during embedding are more likely to be visually masked by the existing structure of the image. This selective approach minimizes the impact on smooth regions and helps maintain the visual integrity of the cover image. The combination of XOR LSB and Sobel edge-based pixel selection ensures that modifications are concentrated in areas where changes are less likely to be visually disruptive. Blowfish encryption provides a high level of security for the embedded data, making it resistant to unauthorized decryption. Meanwhile, the XOR LSB method and edge-based pixel selection contribute to imperceptibility by strategically placing changes where they are less likely to be noticed. Sobel edge detection helps preserve important visual features like edges, which are crucial for image perception. Embedding data in these regions minimizes the risk of introducing noticeable artifacts. The combination of these techniques represents a balanced trade-off between security and imperceptibility. While the embedded information is strongly encrypted, the choice of XOR LSB and edge-based pixel selection aims to ensure that the modifications are subtle and do not significantly alter the visual appearance of the cover image. When selecting the Blowfish algorithm for data hiding in LSB-based image steganography, several factors need to be taken into account, including application type, security, and efficiency. The secure symmetric key block cipher Blowfish is well-known for. It is resistant to brute-force attacks since its key size can reach 448 bits. Our steganography method's security is essential to preventing unwanted extraction of the hidden data.



**Figure 3.5: Blowfish Encryption and Decryption Process**

The symmetric block cipher algorithm called Blowfish encrypts 64 bits of data at a time. It uses the Feistel network as its model, and its operation is split into two steps.

### A. Key-expansions

This section will decompose the key, which consists of a maximum of 448 bits, into many subkey arrays, resulting in a total byte count of 4168.

### B. Data-Encryption

We will iterate the network 16 times during the data encryption procedure. Additionally, there are two types of permutations in each round: key-dependent and data-dependent. The algorithms perform adds or XORs on 32-bit words. For every iteration of this operation, we still need to generate four indexed array data lookup tables.

### C. Key Generation

Blowfish makes extensive use of subkeys. Prior to any data encryption or decryption, these keys are generated.

The p-array consists of 18, 32-bit sub keys:

P1, P2, P3, P4, P5, ....., P18

Each of the four 32-bit S-Boxes has 256 entries:

S1,0, S1,1, S1,2, S1,3, ..... S1,255

S2,0, S2,1, S2,2, S2,3, ..... S2,255

S3,0, S3,1, S3,2, S3,3, ..... S3,255

S4,0, S4,1, S4,2, S4,3, ..... S4,255

Sub Key Generate Steps:

1) Initialize each of the four S-boxes with a fixed string after setting up the P-array initially. Additionally, this string also contains the pi hexadecimal digits (without the first three).

2) Perform an XOR with P1 using the first 32 bits of the key, P2 using the second 32 bits of the key, and so on until all bits of the key (potentially up to P14) are obtained. Whenever the complete P-array has been XORed with key bits, cycle over the key bits repeatedly. (For example, if A is a 64-bit key, then AA, \ AAA, etc., are equivalent keys.) There is at least one equivalent longer key for every short key. The only difference between encryption and decryption is the order in which P1, P2, P3, P4, ..., P18 are employed. [17].

### XOR Embedding process:

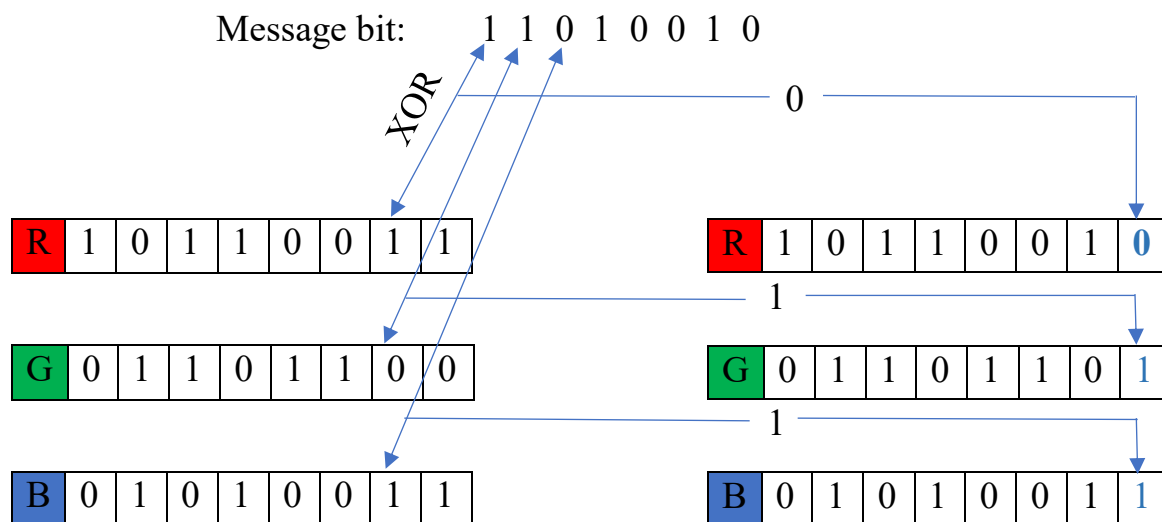
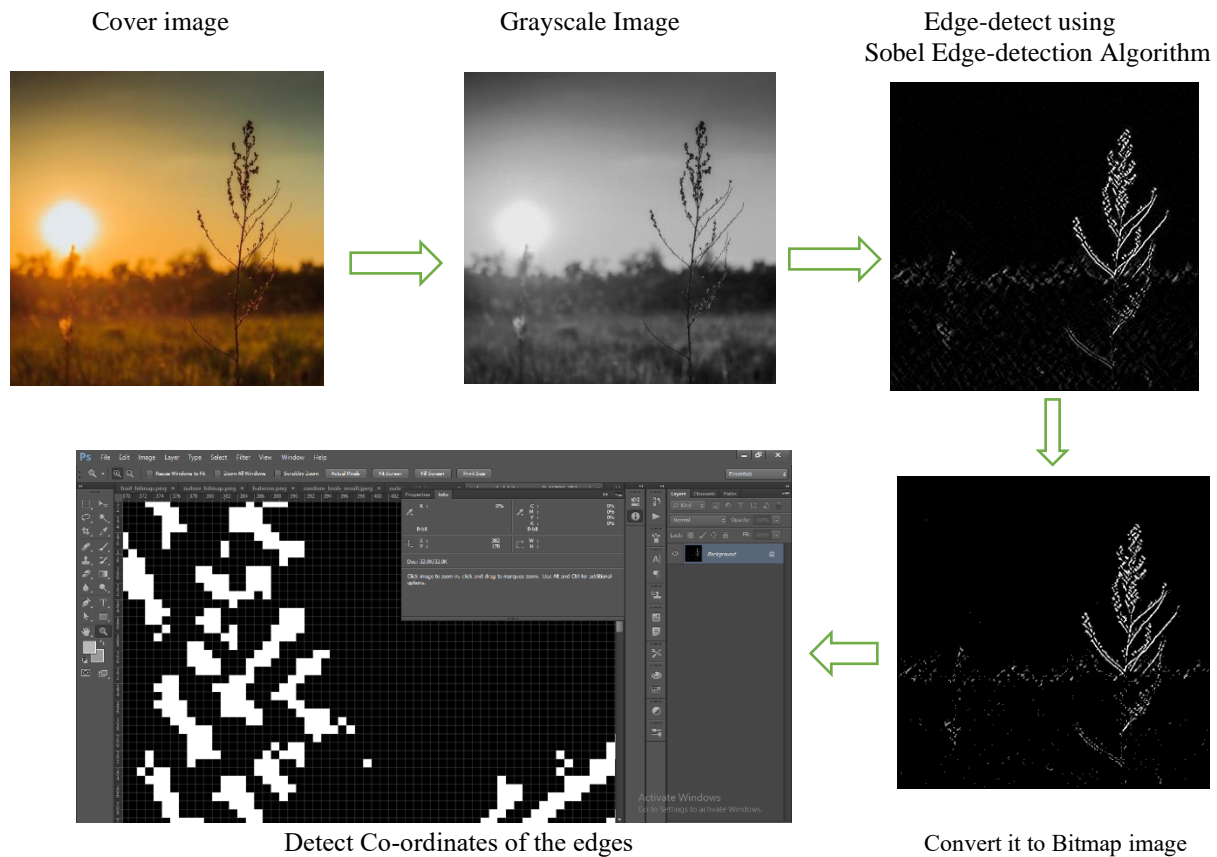


Figure 3.6: XOR Embedding Process



I will now show the process of XOR Embedding in image steganography. Suppose there is a secret message bit 11010010 which will be XORed in the RGB color model into one pixel. The Red binary value is 10110011 which in the decimal is 179. The Green binary value is 01101100 which in the decimal is 108. The Blue binary value is 01010011 which in the decimal is 83. The secret message bit number is 8. I will work here for only one pixel to hide confidential data. So, first of all, secret message's 1st bit is 1 which will be XORed with the Red binary value's 7th bit which is 1. We know,  $1 \oplus 1 = 0$ . Now that XORed bit "0" will be replaced in the Red binary's 8th bit value. Previously the value was 1 in the 8th bit but after XORed that bit value suddenly changed that is 0. In the next step, the secret message's second bit is 1 which will be XORed with the Green binary value's 7th bit. After XORed  $1 \oplus 0 = 1$  value will be replaced with the Green binary value's 8th bit. The value will change which will be 1. The same process will be repeated in the 3rd step. The secret message's third bit is 0 which will be XORed with the Blue binary value's 7th bit. After XORed  $0 \oplus 1 = 1$  value will be replaced with the Blue binary value's 8th bit. The value will overwrite which will be 1. The XOR embedding working process is described here.

In the Sobel edge detection process, we need to create a grayscale image by converting the input color image. This preserves the intensity information while streamlining the procedure. To compute the gradients in both the horizontal and vertical dimensions, use Sobel kernels. Every pixel undergoes the convolution process, producing two gradient images—one for horizontal changes and another for vertical changes. To get the gradient's magnitude at each pixel, combine the images of the horizontal and vertical gradients. The strength of the edge is indicated by the gradient's magnitude. To binarize the gradient magnitude image, set a threshold value. Pixels that are part of an edge are those whose magnitude is greater than the threshold; all other pixels are set to zero. As you go through the binarized image iteratively, note the coordinates of any non-zero pixels.



**Figure 3.7: Detect Co-ordinates from the edges based on Sobel Edge-Detection Algorithm**

In the figure, we add a simple cover image for its edge detection. We just show how to detect edges from this image by using Adobe Photoshop tool. An interruption in the intensity of the image or the initial derivative of the intensity of the image is typically linked to a notable local shift in the image's intensity, which is called an edge. In this tool, we can convert cover image to grayscale image. Then we apply Sobel edge detection algorithm for generating the edge. Now we convert it into bitmap image. From this bitmap image, we can get co-ordinate of that pixel edges. Now we randomly select some pixels for hiding the secret data, basically in the XOR LSB method. We can hide this data by randomly pick some pixel which intensity is very high. Two 3 x 3 convolution masks are applied in the Sobel edge detection process. A gradient is estimated in the x direction by one and in the y direction by another. Because of its extreme sensitivity, the Sobel detector successfully highlights noise in pictures as edges. This Sobel operator is typically used to identify the vertical and horizontal directions of edges in a picture. Already there are some mathematical formulations of Sobel operator exists.

The magnitude of the gradient at each pixel is computed as follows:

$$\text{Gradient Magnitude} = \sqrt{(G_x * I)^2 + (G_y * I)^2} \dots \dots \dots (3.1)$$

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

PSNR compares an image's original to a distorted or compressed version to determine how good the image is. It is used to assess how imperceptible the stego image is in relation to the original image when it comes to image steganography. The mean squared error (MSE) and the maximum pixel value—typically 255 for 8-bit images—are used to calculate the PSNR.

$$\text{PSNR} = 10 \log_{10} \frac{\text{MAX}_x^2}{\text{MSE}} \dots \dots \dots (3.2)$$

MAX= the highest value that can be assigned to a pixel in an image (255 (11111111) for an 8-bit image).

**Mean Square Error (MSE):**

(Maximum inaccuracy between the original and compressed image)

The difference between the corresponding pixels of the original and stego pictures is computed using the mean squared error, or MSE. Reduced MSE values are indicative of improved imperceptibility.

$$\text{MSE} = \frac{1}{a * b} \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} [X(i, j) - Y(i, j)]^2 \dots \dots \dots (3.3)$$

*a*= The height of the frames (number of rows of pixels).

*b*= The width of the frames (number of columns of pixels).

*X(i, j)*= The original frame's pixel intensity at *j*<sup>th</sup> row and *j*<sup>th</sup> column.

*Y(i, j)*= The stego frame's pixel intensity at row and column *j*<sup>th</sup>

Greater reliability is indicated by a lower Mean Squared Error (MSE) value, which shows the least amount of error between the stego and original image.

### **Structured Similarity Index Measurement (SSIM):**

This metric is based on structural content. SIM is a metric that considers the luminance, contrast, and structure of images. It verifies the similarity between the cover image and the stego-image. Compared to PSNR and MSE, it is a more complete indicator of image quality. The range of SSIM values is -1 to 1, with 1 denoting perfect similarity.

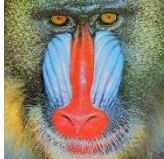


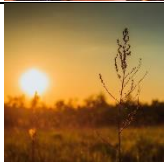
$$\text{SSIM}(X,Y)=\frac{(2\mu_X\mu_Y+C_1)(2\sigma_{XY}+C_2)}{(\mu_X^2+\mu_Y^2+C_1)(\sigma_X^2+\sigma_Y^2+C_2)} \dots \dots \dots (3.4)$$

- $\mu_X, \mu_Y$  are the average pixel values of frames  $X$  and  $Y$  respectability.
- $\sigma_X^2, \sigma_Y^2$  are the variances pixel values in frames  $A$  and  $B$  respectability.
- $\sigma_{XY}$  is the covariance of pixel values between frames  $A$  and  $B$ .
- $C_1, C_2$  are constant.

The SSIM ranges from 0 to 1, and a video is considered high quality if its value is close to 1.

These metrics are frequently used to assess image steganography methods; higher PSNR and SSIM values and lower MSE indicate better stego image imperceptibility.

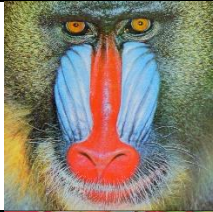
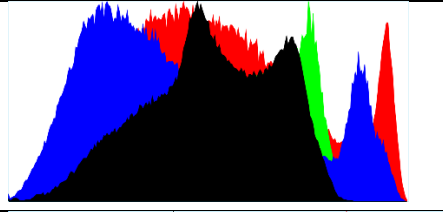
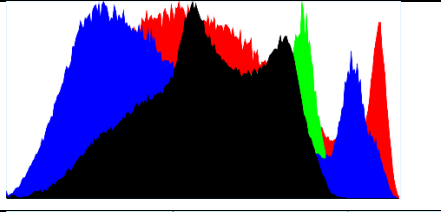

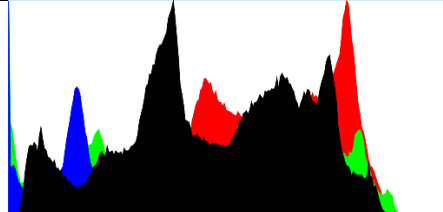
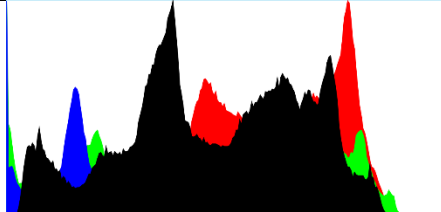

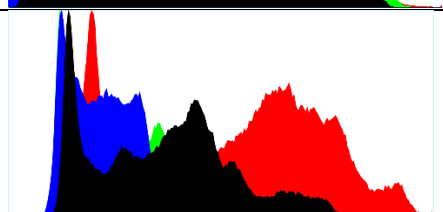
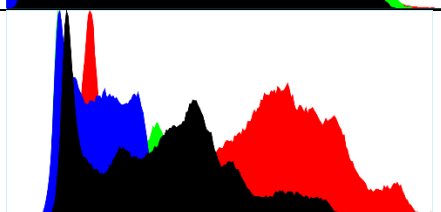
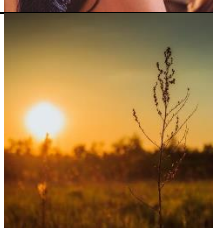
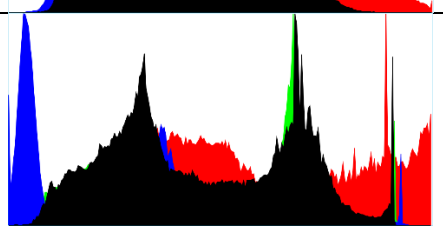
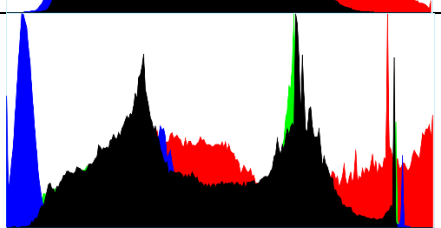
## Chapter-4: Result Analysis and Discussion

Image	Payload	Image Size	PSNR	MSE	RMSE
	12	512×512	89.5286668187	0.0000000011	0.0000333862
	24	512×512	88.3950981781	0.0000000014	0.0000380404
	32	512×512	85.5952242488	0.0000000028	0.0000525096
	32	512×512	85.4142420279	0.0000000029	0.0000536152

**Table 4.4: Result analysis table of the Stego images**

The performance of our suggested technique is described in the experimental results given in this section. Our steganography employs well-known embedding techniques based on LSB. In order to carry out our research, we tested our LSB embedding and Edge-based pixel selection scheme on a variety of standard images at varying resolutions, such as baboon.png, fruit.png, lena.png, and so forth. Table 1 displays these test pictures. Stego-image quality is typically evaluated from two angles. First, we utilize the Peak Signal-to-Noise Ratio (PSNR) measurement to evaluate the difference between the stego and cover images. The Mean Square Error (MSE) is located between the cover and stego pictures. The MSE is described as follows with a cover image of width and height  $m$  and  $n$ , a stego image designated by  $K$ , and a cover image indicated by  $I$ :

$$MSE = \frac{1}{a * b} \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} [X(i, j) - Y(i, j)]^2 \dots \dots \dots (4.1)$$

Image	Histogram of the Cover Picture	Histogram of the Stego Picture
		
		
		
		

**Table 4.4: Comparative analysis of Cover Image and Stego Image Histogram**

Take a look at the 512x512 pixel cover images, which are baboon.png, fruit.png, lena.png, and nature.png. It becomes clear that there is no appreciable difference between the cover and stego images' histograms after inserting a secret message into them. The distribution of pixel intensities in an image is shown by histograms. They act as a visual aid for comprehending the overall composition and tonal changes in an image. Typically, if a secret message or piece of data is embedded into an image, there could be variations in the pixel values that cause noticeable changes to the histogram. On the other hand, interesting questions are raised by the comparative analysis's lack of such differences. The efficiency of the steganographic method used may be one reason for the lack of discrepancy in the histograms. Steganography strives to conceal data within a cover medium, such as an image, without drawing attention to itself. Sophisticated steganographic techniques guarantee that the alterations done to the cover photo are undetectable to statistical tools like histograms as well as the human eye. The ability of the cover photos to hold the embedded data is another factor to take into account. It is easier to conceal more data without appreciably

altering the overall histogram if the cover images have a broad range of pixel intensities and color variations. The fact that there are no appreciable variations in the histograms between the stego and cover images shows that the steganographic procedure was successful in preserving the cover images' visual integrity. It also emphasizes the difficulties faced by anyone trying to use pixel intensity distribution analysis alone to uncover hidden information.

```

1 clear
2 clc
3 close all
4
5 coverDir = '/MATLAB Drive/cover image';
6 stegoDir = '/MATLAB Drive/stego image';
7
8 coverFrames = dir(coverDir);
9 stegoFrames = dir(stegoDir);
10

```

Command Window

```

The MSE value is 0.000000117
The RMSE value is 0.0001081384
The SNR value is 18.9938491497
The MAE value is 0.0000029819
The SSIM value is 0.9999996502
The MMD value is 0.0039215686
The Peak-SNR value is 79.3204035355
The BER value is 0.0000000000

```

Zoom: 100% UTF-8 CRLF script Ln 7 Col 1

```

1 clear
2 clc
3 close all
4
5 coverDir = '/MATLAB Drive/cover image';
6 stegoDir = '/MATLAB Drive/stego image';
7
8 coverFrames = dir(coverDir);
9 stegoFrames = dir(stegoDir);
10

```

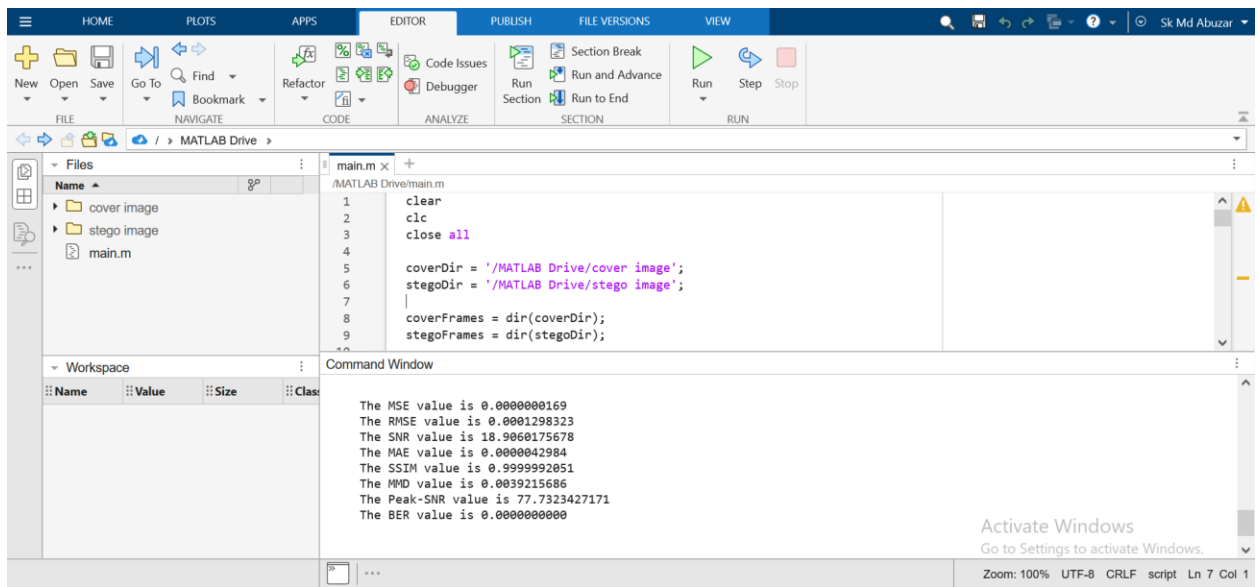
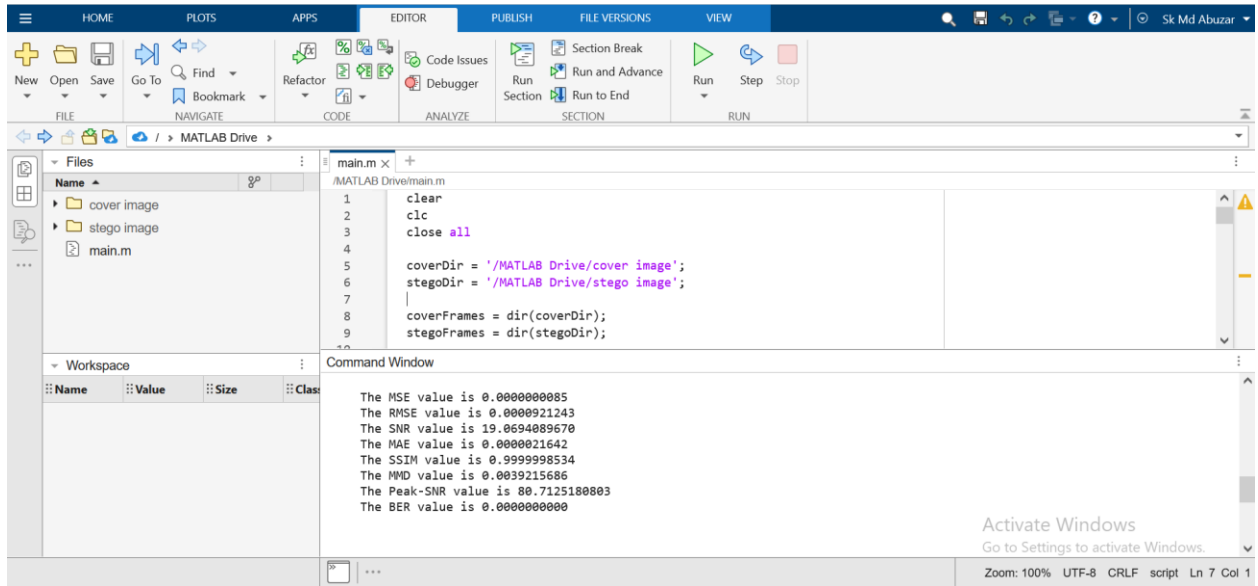
Command Window

```

The MSE value is 0.000000159
The RMSE value is 0.0001262432
The SNR value is 18.9196005796
The MAE value is 0.0000040640
The SSIM value is 0.9999997850
The MMD value is 0.0039215686
The Peak-SNR value is 77.9758392880
The BER value is 0.0000000000

```

Zoom: 100% UTF-8 CRLF script Ln 7 Col 1

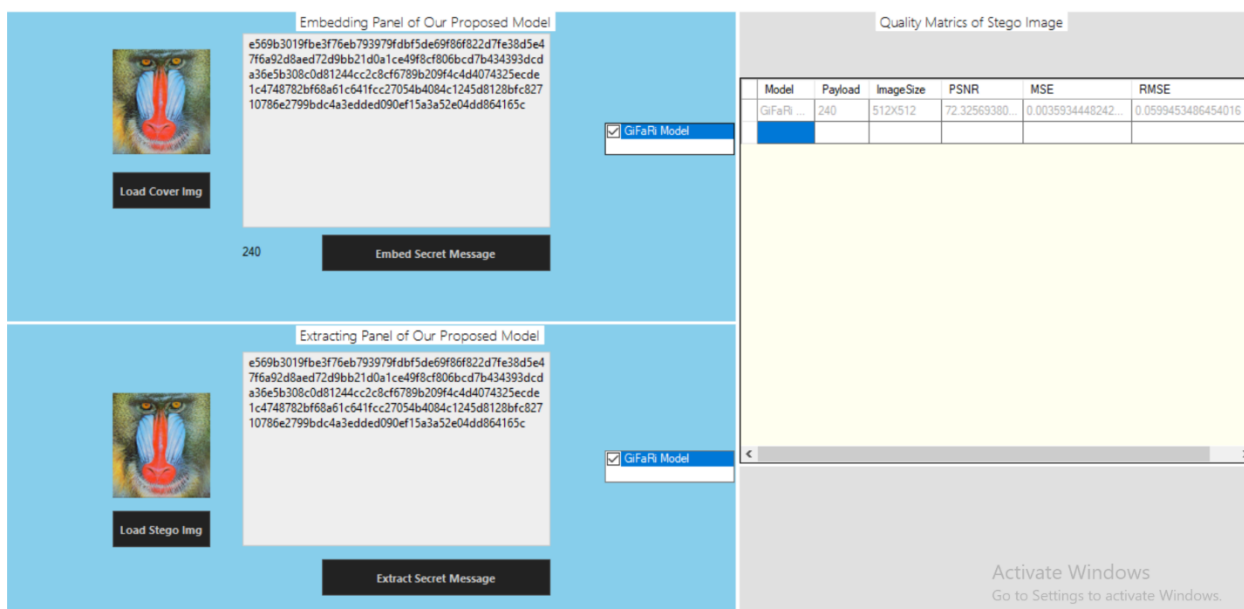


**Figure 4.1: Matlab Code Implementation**

On the figure 1, we can see that we mainly tested PSNR, MSE, RMSE value in the MATLAB for the four images named baboon.png, fruit.png, lena.png, nature.png. For the first image, we can observe that we get PSNR, MSE and RMSE value consequently 89.5286668187, 0.0000000011, 0.0000333862. For the second image, we can observe that we get PSNR, MSE and RMSE value consequently 88.3950981781, 0.0000000014, 0.0000380404. For the third image, we can observe



that we get PSNR, MSE and RMSE value consequently 85.5952242488,0.0000000028, 0.0000525096. For the fourth image, we can observe that we get PSNR, MSE and RMSE value consequently 85.4142420279, 0.0000000029, 0.0000536152. Even if we notice at figure 2, we saw that difference between histogram of the cover and stego images are less noticeable. Our model works efficiently that is proved when we observe in the figure 3 because comparative analysis of previous researchers our combination model works very well. We measure PSNR and MSE for better imperceptibility in image. We saw that every image PSNR values are upto 80 that means our model can less detectable after embedding which is tremendous to improve that our model is far better. We know that  $PSNR > 40$  is very good to prove that the image is less detectable. But we get 80+ value when we tested it in the MATLAB.



**Figure 4.2: Proposed model code implementation**

We implemented C# code for our proposed model in Visual Studio Code Editor. Basically, this is the interface/GUI of our proposed model. We want to describe the whole process of this model. There is an embedding and extracting panel in the model and in the right side there are PSNR, MSE, RMSE calculation for our model. In the embedding panel, we here get an input box for embedding the secret message. We apply Blowfish algorithm here. There are various online blowfish encryption tool to encrypt our secret data. We encrypt four data or message by using these tools and in the input box we input those encrypted data and click embed. After embedding the secret data is embedded with the cover image and convert it to the stego image. When we embed

secret data inside some image pixels suddenly, we will notice PSNR, MSE and RMSE calculation in the right side of our GUI. Here, from this panel we can get those values. We apply this process for the four stego images. This is the overview of the embedding process. Let's talk about extracting process. In the down side, basically extracting panel we will notice, if we load the stego images one by one and click extract suddenly confidential data will be visible here by using this proposed model. Thus data will be securely communicate via transmission channel.

Image	Reference [18]	Reference [19]	References [20]	Proposed Model Result
Peppers	76.39	-	54.88	88.39
Lena	77.90	-	54.82	85.59
Baboon	78.49	72.62	72.48	89.52

**Table 2.3: Comparison PSNR of our proposal with other researchers**

We compare three image's PSNR value with other existing technique. When we compare, we see that in the Lena image, the PSNR value found was 54.82 and another researcher found was 77.90. But our model performs better than others because we find the result 85.88. Similarly, from the Baboon image, the PSNR value was found consequently 72.48, 72.62, 78.49 whether our model's result is found 85.78 which is obviously better than other's. At the last Peppers image, the previous researcher's result was consequently 54.88 and 76.39 whether our model's result is found 85.68 which is great. So, from the discussion we can say that our model performs far better than others. The PSNR value is increased when we embed hidden text in the image.

Image	References [21]	References [22]	References [22]	Proposed Model Result
Lena	0.002893	0.0032	0.4268	0.0000000026
Baboon	0.002825	0.0031	0.4253	0.0000000026
Peppers	0.002712	0.0037	0.4243	0.0000000027

**Table 4.4: Comparison MSE of our proposal with other researchers**

Here we compare MSE values with other existing techniques. We know that low MSE is associated with better perceptual quality. We added three researchers gathered MSE result from their papers. From the Lena image we found MSE value was 0.002893, 0.0032 and 0.4268. But our MSE value we get 0.0000000026 that is very best. Consequently, from the Baboon image, we get MSE value from the previous researchers that was 0.002825, 0.0031 and 0.4253. But our model performs well

here because our value is 0.000000026. From the 3<sup>rd</sup> image Peppers, the value was 0.002712, 0.0037 and 0.4243 but our value is we get 0.000000027 that is cool. It's important to note that a low MSE is a common goal in image steganography. So, our model is working better than other existing techniques.

## **Chapter-5: Conclusion**

In conclusion, the Proposed model presented in this thesis represents a robust and advanced approach to secured data hiding in LSB-based image steganography. By combining the XOR LSB method, the Blowfish algorithm, and the precision of edge-based pixel selection through the Sobel edge detection algorithm, we have achieved a synergistic enhancement in imperceptibility and security. The utilization of XOR LSB ensures a sophisticated embedding process, making it resistant to traditional steganalysis techniques. The integration of the Blowfish algorithm adds an additional layer of security, fortifying the confidentiality of the hidden data. Furthermore, the incorporation of edge-based pixel selection through the Sobel edge detection algorithm contributes to both the visual quality and security of the steganographic image by focusing on regions less likely to be visually detected. The suggested method not only solves the present problems but also lays the groundwork for further developments in the area. While the Proposed model has demonstrated commendable imperceptibility in LSB-based image steganography, future researchers are encouraged to explore specific avenues that address the challenges of increasing data capacity and fortifying robustness. By delving into these tailored research directions, future investigators can contribute to advancing the capabilities of this model, making significant strides in both data hiding capacity and robustness..

## References

- [1] D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," *Proceedings of the IRE*, vol. 40, no. 9, 1952, doi: 10.1109/JRPROC.1952.273898.
- [2] A. A. Arab, M. J. B. Rostami, and B. Ghavami, "An image encryption algorithm using the combination of chaotic maps," *Optik (Stuttg)*, vol. 261, 2022, doi: 10.1016/j.ijleo.2022.169122.
- [3] H. W. Tseng and H. S. Leng, "A steganographic method based on pixel-value differencing and the perfect square number," *J Appl Math*, vol. 2013, 2013, doi: 10.1155/2013/189706.
- [4] J. J. Roque, "SLSB: Improving the steganographic algorithm LSB," in *Security in Information Systems - Proceedings of the 7th International Workshop on Security in Information Systems - WOSIS 2009 In Conjunction with ICEIS 2009*, 2009. doi: 10.5220/0002169700570066.
- [5] I. Almomani, A. Alkhayer, and W. El-Shafai, "A Crypto-Steganography Approach for Hiding Ransomware within HEVC Streams in Android IoT Devices," *Sensors*, vol. 22, no. 6, 2022, doi: 10.3390/s22062281.
- [6] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Trans Pattern Anal Mach Intell*, vol. 11, no. 7, 1989, doi: 10.1109/34.192463.
- [7] F. Marcelloni and M. Vecchio, "A simple algorithm for data compression in wireless sensor networks," *IEEE Communications Letters*, vol. 12, no. 6, 2008, doi: 10.1109/LCOMM.2008.080300.
- [8] M. A. Saleh, "Image Steganography Techniques - A Review Paper," *IJARCCCE*, vol. 7, no. 9, 2018, doi: 10.17148/ijarccce.2018.7910.
- [9] V. Nagaraj, V. Vijayalakshmi, and G. Zayaraz, "Color Image Steganography based on Pixel Value Modification Method Using Modulus Function," *IERI Procedia*, vol. 4, 2013, doi: 10.1016/j.ieri.2013.11.004.

- [10] X. Zhang and S. Wang, "Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security," *Pattern Recognit Lett*, vol. 25, no. 3, 2004, doi: 10.1016/j.patrec.2003.10.014.
- [11] C. H. Yang, C. Y. Weng, S. J. Wang, and H. M. Sun, "Adaptive data hiding in edge areas of images with spatial LSB domain systems," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, 2008, doi: 10.1109/TIFS.2008.926097.
- [12] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on lsb matching revisited," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, 2010, doi: 10.1109/TIFS.2010.2041812.
- [13] M. Hussain and M. Hussain, "Information hiding using edge boundaries of objects," *International Journal of Security and its Applications*, vol. 5, no. 3, 2011.
- [14] J. G. Yu, E. J. Yoon, S. H. Shin, and K. Y. Yoo, "A new image steganography based on 2k correction and edge-detection," in *Proceedings - International Conference on Information Technology: New Generations, ITNG 2008*, 2008. doi: 10.1109/ITNG.2008.101.
- [15] I. Kich, E. B. Ameer, and Y. Taouil, "Image steganography based on edge detection algorithm," in *2018 International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS 2018*, 2018. doi: 10.1109/ICECOCS.2018.8610603.
- [16] S. K. Salim, M. M. Msallam, and H. I. Olewi, "Hide text in an image using Blowfish algorithm and development of least significant bit technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 1, 2023, doi: 10.11591/ijeecs.v29.i1.pp339-347.
- [17] N. A. Kofahi, "An empirical study to compare the performance of some symmetric and asymmetric ciphers," *International Journal of Security and its Applications*, vol. 7, no. 5, 2013, doi: 10.14257/ijisia.2013.7.5.01.
- [18] M. Damrudi and K. J. Aval, "Image steganography using LSB and encrypted message with AES, RSA, DES, 3DES, and blowfish," *Int J Eng Adv Technol*, vol. 8, no. 6 Special Issue 3, 2019, doi: 10.35940/ijeat.F1033.0986S319.

- [19] M. Damrudi and K. J. Aval, "Two stage steganography on compressed and encrypted message," *International Journal of Circuits, Systems and Signal Processing*, vol. 15, 2021, doi: 10.46300/9106.2021.15.54.
- [20] "A Development of Least Significant Bit Steganography Technique," *Iraqi Journal of Computer, Communication, Control and System Engineering*, 2020, doi: 10.33103/uot.ijccce.20.1.4.
- [21] J. Chandrasekaran, G. Arumugam, and D. Rajkumar, "Ensemble of logistic maps with genetic algorithm for optimal pixel selection in image steganography," in *2nd International Conference on Electronics and Communication Systems, ICECS 2015*, 2015. doi: 10.1109/ECS.2015.7124769.
- [22] M. M., A. A., and F. A., "An Improved Image Steganography Method Based on LSB Technique with Random Pixel Selection," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 3, 2016, doi: 10.14569/ijacsa.2016.070350.
- [23] M. Kalita, T. Tuithung, and S. Majumder, "A New Steganography Method Using Integer Wavelet Transform and Least Significant Bit Substitution," *Computer Journal*, vol. 62, no. 11, 2019, doi: 10.1093/comjnl/bxz014.
- [24] V. Singhal, D. Singh, and S. K. Gupta, "A Novel Approach for Enhancement of Blowfish Algorithm by using DES, DCT Methods for Providing, Strong Encryption and Decryption Capabilities," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 7 S, 2023, doi: 10.17762/ijritcc.v11i7s.7008.
- [25] T. Sanida, A. Sideris, and M. Dasygenis, "A Heterogeneous Implementation of the Sobel Edge Detection Filter Using OpenCL," in *2020 9th International Conference on Modern Circuits and Systems Technologies, MOCASST 2020*, 2020. doi: 10.1109/MOCASST49295.2020.9200249.
- [26] R. A. A S and S. Gopalan, "Comparative Analysis of Eight Direction Sobel Edge Detection Algorithm for Brain Tumor MRI Images," in *Procedia Computer Science*, 2022. doi: 10.1016/j.procs.2022.03.063.

- [27] H. A. W. Jasim Albayati and S. A. Ali, "A Comparative Study of Image Steganography Based on Edge Detection," in *Journal of Physics: Conference Series*, 2021. doi: 10.1088/1742-6596/1818/1/012032.
- [28] N. Jain, S. Meshram, and S. Dubey, "Image Steganography Using LSB and Edge – Detection Technique," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, no. 3, 2012.
- [29] N. D. Lynn, A. I. Sourav, and A. J. Santoso, "Implementation of Real-Time Edge Detection Using Canny and Sobel Algorithms," *IOP Conf Ser Mater Sci Eng*, vol. 1096, no. 1, 2021, doi: 10.1088/1757-899x/1096/1/012079.
- [30] P. Vinista and M. Milton, "A Novel Modified Sobel Algorithm for Better Edge Detection of Various Images," *International Journal of Emerging Technologies in Engineering Research (IJETER)*, vol. 7, no. 3, 2019.
- [31] G. Umamaheswari and C. P. Sumathi, "Pixel selection based on the difference between secret message and cover image pixel for efficient information hiding," in *Proceedings of the 10th International Conference on Intelligent Systems and Control, ISCO 2016*, 2016. doi: 10.1109/ISCO.2016.7726877.
- [32] L. Han, Y. Tian, and Q. Qi, "Research on edge detection algorithm based on improved sobel operator," *MATEC Web of Conferences*, vol. 309, 2020, doi: 10.1051/mateconf/202030903031.
- [33] R. Tian, G. Sun, X. Liu, and B. Zheng, "Sobel edge detection based on weighted nuclear norm minimization image denoising," *Electronics (Switzerland)*, vol. 10, no. 6, 2021, doi: 10.3390/electronics10060655.