**IMAGE STEGANOGRAPHY USING IMAGE ENCRYPTION AND ITS COMPARISON USING HISTOGRAM**

****

**Ahmad Hassan**

**2021-ag-8004**

**Atteeq ur Rehman**

**2021-ag-8001**

**Bachelors of Science in Information Technology**

**Department of Computer Science**

**Faculty of Sciences**

**University of Agriculture, Faisalabad,**

**Pakistan**

**2024**

**DECLARATION**

**We hereby declare that the contents of the documentation, “IMAGE STEGANOGRAPHY USING IMAGE ENCRYPTION AND ITS COMPARISON USING HISTOGRAM” are product of our own research and no part has been copied from any published source. We further declare that this work has not been submitted by any other group for presentation.**

**Ahmad Hassan**

**Atteeq ur Rehman**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapter.** | **Title** | **Page** |
| **1** | **Abstract** | **04** |
| **2** | **Introduction**  2.1 Related Work  2.2 Problem Background | **04** |
| **3** | **Methodology**  3.1 Method Used  3.2 Performance Metrics  3.3 Code of Project | **07** |
| **4** | **Results and Discussion**  4.1 Project Questions  4.2 Future Work | **15** |
| **5** | **Summary** | **17** |

**Chapter 01**

**Abstract**

The art of information hiding has received much attention in the recent years as security of information has become a big concern in this internet era. As sharing of sensitive information via a common communication channel has become inevitable, Steganography – the art and science of hiding information has gained much attention. We are also surrounded by a world of secret communication, where people of all types are transmitting information as innocent as an encrypted credit card number to an online-store and as insidious as a terrorist plot to hijackers. Steganography derives from the Greek word steganos, meaning covered or secret, and grapy (writing or drawing). Steganography is a technology where modern data compression, information theory, spread spectrum, and cryptography technologies are brought together to satisfy the need for privacy on the Internet. This paper is an attempt to analyse the various techniques used in steganography and to identify areas in which this technique can be applied, so that the human race can be benefited at large.

**Chapter 02**

**Introduction**

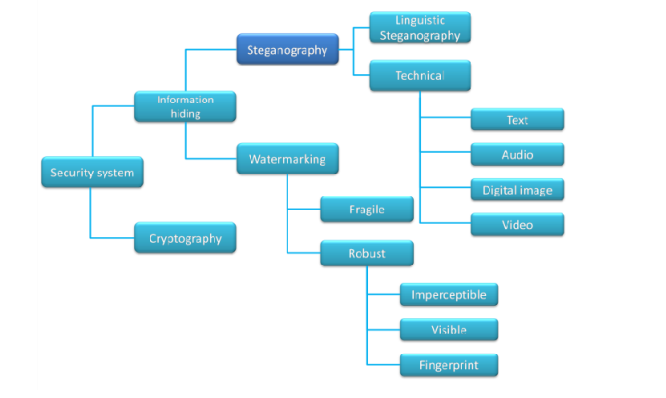
In today’s digital landscape, the escalating threat of malicious attacks targeting businesses, government entities, and private individuals has prompted a growing need for robust information security measures. As a result, developers and researchers are actively seeking technical solutions to ensure the privacy and confidentiality of documents transmitted over communication channels. With the rapid shift of numerous services from local networks to the Internet, the significance of information security and data protection has reached new heights. To counteract the evolving processing capabilities of cryptanalysis, encryption algorithms have undergone significant advancements to bolster their complexity, thereby deterring attackers from swiftly compromising encrypted data. It should be remembered that encrypted data always take an attacker’s attention to try to crack the encrypted data as a challenge. Complex steganography techniques hide the data’s existence, leading to fewer attacking possibilities. Steganography represents the technique of concealing data inside a cover medium, the origin of the word ‘Steganography’ is Greek, and it means ‘hidden or covered writing’. Steganography mainly aims at concealing the very presence of the message inside the cover medium. Its technique contains many secret communication methods that hide the mere fact of concealed data. Conventional methods include the usage of microdots, invisible inks, etc. The recent techniques of steganography attempt to take advantage of video files, digital media images, audio files, etc. Steganography can combine hiding and encryption mechanisms. This makes finding the data hidden in the object much more complicated since the data are unreadable, and any attacking techniques see the results as unexpectable and confusing. Therefore, much research has been conducted to enhance the hiding mechanisms, particularly as a message exchange mechanism for top-secret data.

The applications of information hiding systems mainly range over a broad area from military, intelligence agencies, online elections, internet banking, medical-imaging and so on. These variety of applications make steganography a hot topic for study. The cover medium is usually chosen keeping in mind the type and the size of the secret message and many different carrier file formats can be used. In the current situation digital images are the most popular carrier/cover files that can be used to transmit secret information

Steganography equation is:

**Stego-medium** = Cover medium + Secret message + Stego key

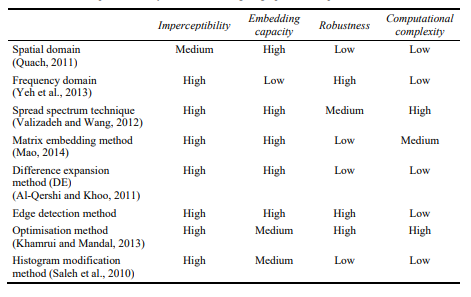
The general model of data hiding can be described as follows. The embedded data is the message that one wishes to send secretly. It is usually hidden in an innocuous message referred to as a cover text or cover-image or cover-audio as appropriate, producing the stego-text or other stego-object. A stego-key is used to control the hiding process so as to restrict detection and /or recovery of the embedded data to parties who know it.



**Figure 01:** Classification of the existing methods based on the secret media use

A few key properties that must be considered when creating a digital data hiding system are

* **Imperceptibility:** Imperceptibility is the property in which a person should be unable to distinguish the original and the stego-image.
* **Embedding Capacity:** Refers to the amount of secret information that can be embedded without degradation of the quality of the image.
* **Robustness:** Refers to the degree of difficulty required to destroy embedded information without destroying the cover image.



**Table 1:** *Comparative analysis of various steganographic techniques.*

**2.1 Related Work**

The symmetric encryption technique is one of the oldest and most famous methods of maintaining data security; the secret key can be a text or several random characters. The secret key is implemented by a text message to change its content. The method Appl. Sci. 2023, 13, 11771 3 of 20 of encryption used in this technique is to convert each character to several alphabetic characters when the sender and recipient know the keys of all parties and then use the secret key to encrypt and decrypt the message. Blowfish is an encryption technique, more specifically a block cipher. It is frequently referred to as a cipher. Blowfish uses keys ranging from 3 to 2448 bits and has a 64-bit block size. Its creator, Bruce Schneier, claims it is free to use, open source, and royalty free. Although Blowfish is utilized in several cipher suites and encryption methods, AES is often employed. Blowfish is secure since no cryptanalysis attempt has succeeded.

**2.2 Problem Background**

In today's digital age, ensuring the confidentiality and integrity of sensitive information is of paramount importance. However, conventional encryption techniques may not suffice when it comes to concealing data within digital media like images. This necessitates the exploration of alternative methods such as steganography, which involves hiding secret information within seemingly innocuous cover media. Image steganography, in particular, presents a compelling solution for covert communication, as images are ubiquitous in online communication platforms and social media. Despite its potential applications in secure communication and data protection, steganography also raises concerns regarding its potential misuse for malicious purposes, such as digital piracy or espionage. Additionally, the effectiveness and robustness of steganographic techniques heavily depend on factors such as the embedding capacity, imperceptibility of modifications, and resilience against detection. Hence, there is a pressing need for robust and efficient steganographic methods that strike a balance between data hiding capacity and the preservation of visual fidelity in cover images. Addressing these challenges requires innovative approaches that leverage advanced encryption algorithms and image processing techniques to ensure secure and reliable communication while minimizing the risk of unauthorized access or detection.

**Chapter 03**

**Methodology**

**3.1 Method Used**

***Least Significant Bits (LSB) substitution method***

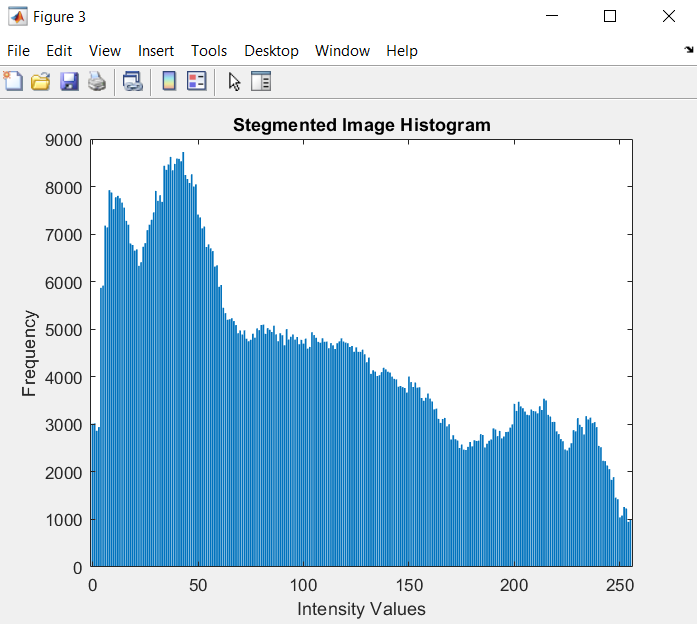
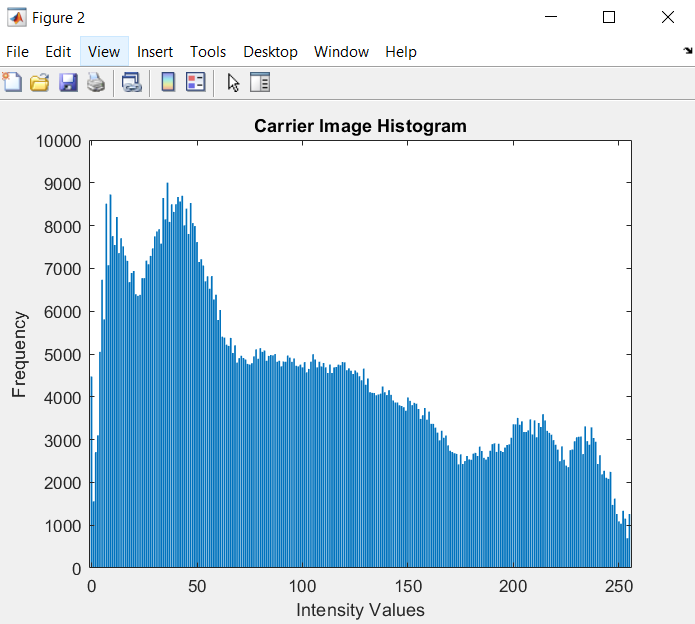
Conventionally, Least Significant Bits (LSB) substitution method is employed to perform image steganography. Images are usually of higher pixel quality, out of which not all the pixels are used. LSB methods works under the assumption that modifying a few pixel values would not show any visible changes. The secret information is converted into a binary form. The cover image is scanned to determine the least significant bits in the noisy area. The binary bits from the secret image are then substituted in the LSB of the cover image. The substitution method has to be performed cautiously as overloading the cover image may lead to visible changes leaking the presence of the secret information. With the LSB method as the baseline, a number of related methods have been proposed. A Huffman encoding method is used to encode the secret message into the binary bits. The encoded bits are then embedded in the cover image using the LSB method. Another version of the LSB method is used for RGB images. The cover image is in 3 channels and they are bit sliced. The secret message is embedded in all the three planes in the 2:2:4 ratio for R, G and B planes. Not only spatial domain, quantum images are also used. The frequency domain is exploited in quantum image domain and the pixels which are considered to be affecting the colour are used to hide the secret bits. A combination of cryptography and steganography is utilized in which the LSB of the cover image is replaced with the most significant bits of the secret image. The pseudo random number generator is used to select the pixels and the key is encrypted using rotation every time. A k-LSB method is proposed in where the k least bits are replaced with the secret message. For steganalysis, an entropy filter is used to detect and uncover the secret image. The LSB methods are used in hiding the secret information inside videos also. Videos are sequences of images called the video frames. Each video is dissected into image frames and the binary bits of the secret information is hidden in the LSB of the image frames of the video. A basic form of LSB substitution method is used. A combination of the Huffman encoding and LSB substitution methods is used on videos. Another interesting approach can be seen where along with the image frames of the video, audio is also used to enhance the hiding. Besides the LSB methods, has proposed a combination of Discrete Cosine Transformation (DCT) and Discrete Wavelet Transformation (DWT) for hiding the secret message inside a cover video. To find the regions of interest, the multiple object tracking (MOT) method is used. The secret data is encoded first and then converted to binary bits before embedding it in the cover video.

The Least Significant Bit (LSB) is implemented by replacing the least significant bit in the cover image with the bit from a secret message; the LSB method hides the binary values ‘101100101’ in a 24-bit image. The algorithm starts by uploading the cover image and the secret message, and then an end marker represented by an array of characters is added to the secret message. This is performed to allow the awareness of the recovery phase of when to stop recovering in case the secret message bit’s numbers do not need the full cover image pixels to be hidden since, in this case, the recovery tool continues to extract wrong data if no ending marker is found. After that, the algorithm receives each character in the secret message using a loop. It converts it to a binary value and selects three sequence pixels from the cover image based on the index of the count parameter, which starts from zero and is incremented each time a pixel is selected. Later, the algorithm extracts the three-colour components, R, G, and B, for each pixel chosen and converts each colour component to a binary value. It masks the least bit of each colour component to zero by applying (AND with 11111110). This allows the replacement of the secret character binary bits by the least bit in each colour component, which is performed later by applying XOR with one bit from the selected character bits. It is worth mentioning that the least significant bit (LSB) algorithm has been improved using the selected least considerable bit (SLSB); these techniques proposed to enhance the performance of the LSB method by hiding information only in one of the three colours at each pixel of the cover image to reduce the chance of the confidential data being detected.

***Histogram Comparison***

The histogram comparison technique utilized in our code plays a crucial role in analyzing the distribution of pixel intensities within both the original carrier image and the steganographic image. By visualizing the frequency of intensity values across different levels, histograms offer valuable insights into the underlying characteristics of the images. This comparison allows us to assess the impact of data embedding on the image's tonal range and distribution, thereby gauging the effectiveness of the steganographic process in concealing the secret message while preserving the integrity of the carrier image. Analyzing histograms facilitates the identification of any significant deviations or anomalies introduced during the steganographic process. Discrepancies in the histograms between the original and steganographic images may indicate areas where modifications have occurred, potentially revealing the presence of hidden data. Furthermore, comparing histograms enables researchers to evaluate the perceptual quality of the steganographic image by assessing factors such as contrast, brightness, and overall tonal balance. This quantitative analysis complements subjective evaluations of image quality, providing a comprehensive understanding of the visual impact of data embedding.

Insights gleaned from histogram comparisons can inform further refinements to steganographic algorithms and techniques. By identifying areas of improvement based on histogram analysis, researchers can iteratively enhance the concealment capabilities and visual fidelity of steganographic methods. Additionally, histogram comparison serves as a valuable tool for forensic analysis, enabling the detection of hidden information in images and aiding in the investigation of potential security breaches or illicit activities. Overall, the utilization of histogram comparison in our code not only enhances our understanding of steganographic processes but also facilitates the development of more robust and effective methods for secure communication and data protection.



**Figure 02:** Histogram Comparison

**3.2 Performance Metrics Used**

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

PSNR is a widely used metric in image processing to assess the quality of reconstructed or processed images. It measures the ratio between the maximum possible power of a signal (the original image) and the power of noise or distortion that affects its fidelity. PSNR is calculated using the logarithm of the mean squared error (MSE) between the original and processed images. The higher the PSNR value, the lower the distortion and the better the quality of the processed image. PSNR is particularly useful in steganography for quantifying the perceptual quality of steganographic images. It helps determine how well the secret message is hidden within the carrier image while minimizing visible distortion.

***Mean Squared Error (MSE)***

MSE is a fundamental metric used to quantify the average squared difference between the pixel values of two images. In the context of image processing, MSE measures the discrepancy between the original image and a processed or reconstructed version of it. A lower MSE value indicates that the processed image closely resembles the original image. MSE is computed by averaging the squared differences between corresponding pixel values in the original and processed images. It is a valuable tool in steganography for objectively evaluating the fidelity of recovered images. By comparing MSE values between the original and steganographic images, researchers can assess the effectiveness of the steganographic technique in preserving the visual content of the original image while embedding the secret message.

**3.3 Code of Project**

clc

clear all

close all

warning off

[filename, pathname] = uigetfile({'.jpg';'.png';'\*.bmp'}, 'Select Carrier File');

a = imread(fullfile(pathname, filename));

subplot(3,2,1);

imshow(a);

title('Carrier Image');

[filename, pathname] = uigetfile({'.jpg';'.png';'\*.bmp'}, 'Select Secret File');

x = imread(fullfile(pathname, filename));

subplot(3,2,2)

imshow(x);

title('Secret Image');

[r c g]=size(a);

x=imresize(x,[r c]);

ra=a(:,:,1);

ga=a(:,:,2);

ba=a(:,:,3);

rx=x(:,:,1);

gx=x(:,:,2);

bx=x(:,:,3);

sk=uint8(rand(r,c)\*255);%Secret key

rx=bitxor(rx,sk);

gx=bitxor(gx,sk);

bx=bitxor(bx,sk);

subplot(3,2,3);

imshow(cat(3,rx,gx,bx));

title('Encrypted Secret Message');

for i=1:r

for j=1:c

nc(i,j)= bitand(ra(i,j),254);

ns(i,j)= bitand(rx(i,j),128);

ds(i,j)=ns(i,j)/128;

fr(i,j)=nc(i,j)+ds(i,j);

end

end

redsteg=fr;

for i=1:r

for j=1:c

nc(i,j)= bitand(ga(i,j),254);

ns(i,j)= bitand(gx(i,j),128);

ds(i,j)=ns(i,j)/128;

fr(i,j)=nc(i,j)+ds(i,j);

end

end

greensteg=fr;

for i=1:r

for j=1:c

nc(i,j)= bitand(ba(i,j),254);

ns(i,j)= bitand(bx(i,j),128);

ds(i,j)=ns(i,j)/128;

fr(i,j)=nc(i,j)+ds(i,j);

end

end

bluesteg=fr;

finalsteg=cat(3,redsteg,greensteg,bluesteg);

redstegr=finalsteg(:,:,1);

greenstegr=finalsteg(:,:,2);

bluestegr=finalsteg(:,:,3);

subplot(3,2,4);

imshow(finalsteg);

title('Stegmented Image');

for i=1:r

for j=1:c

nc(i,j)=bitand(redstegr(i,j),1);

ms(i,j)=nc(i,j)\*128;

end

end

recoveredr=ms;

for i=1:r

for j=1:c

nc(i,j)=bitand(greenstegr(i,j),1);

ms(i,j)=nc(i,j)\*128;

end

end

recoveredg=ms;

for i=1:r

for j=1:c

nc(i,j)=bitand(bluestegr(i,j),1);

ms(i,j)=nc(i,j)\*128;

end

end

recoveredb=ms;

output=cat(3,recoveredr,recoveredg,recoveredb);

subplot(3,2,5);

imshow(output);

title('Recovered Encrypted Image');

red\_band=bitxor(output(:,:,1),sk);

green\_band=bitxor(output(:,:,2),sk);

blue\_band=bitxor(output(:,:,3),sk);

combined=cat(3,red\_band,green\_band,blue\_band);

% Enhance colors using histogram equalization

img\_enhanced = histeq(combined);

% Sharpen the image using unsharp masking

img\_sharpened = imsharpen(img\_enhanced, 'Amount', 0.5);

% Smooth the image using a Gaussian filter

img\_smoothed = imgaussfilt(img\_sharpened, 2); % Adjust sigma for desired smoothness

subplot(3,2,6);

imshow(img\_smoothed);

title('Decrypted secret message signal');

% For Histogram

show\_histograms = questdlg('Do you want to display histograms?', ...

'Histogram Option', 'Yes', 'No', 'No');

if strcmpi(show\_histograms, 'Yes')

[counts, bins] = imhist(a);

figure; % Create a new figure window for the first histogram

bar(bins, counts);

title('Carrier Image Histogram');

xlabel('Intensity Values');

ylabel('Frequency');

[count, bin] = imhist(finalsteg);

figure; % Create a new figure window for the second histogram

bar(bin, count);

title('Stegmented Image Histogram');

xlabel('Intensity Values');

ylabel('Frequency');

end

% Prompt for saving options

choice = questdlg('Save output images as a ZIP file or in a folder?', ...

'Save Options', 'ZIP File', 'Folder', 'Folder');

if strcmpi(choice, 'ZIP File')

% Get a filename for the ZIP file (will be used later)

[filename, pathname] = uiputfile({'\*.zip'}, 'Specify ZIP filename');

if filename ~= 0

full\_filename = fullfile(pathname, filename);

full\_filename = char(full\_filename); % Ensure correct path format

if isempty(pathname) % Handle missing path

full\_filename = fullfile(pwd, filename); % Use current directory

end

end

% Always save images to a temporary folder first

temp\_folder\_path = tempname; % Create a temporary folder

mkdir(temp\_folder\_path); % Create the folder

% Save images to the temporary folder

imwrite(a, fullfile(temp\_folder\_path, '1original\_image.jpg'));

imwrite(x, fullfile(temp\_folder\_path, '2secret\_data\_image.jpg'));

imwrite(cat(3,rx,gx,bx), fullfile(temp\_folder\_path, '3encrypted\_secret\_image.jpg'));

imwrite(finalsteg, fullfile(temp\_folder\_path, '4combined\_image.jpg'));

imwrite(output, fullfile(temp\_folder\_path, '5extracted\_encrypted\_image.jpg'));

imwrite(img\_smoothed, fullfile(temp\_folder\_path, '6smoothed\_secret\_image.jpg'));

% Create the ZIP file from the temporary folder

zip(full\_filename, temp\_folder\_path);

% Delete the temporary folder

rmdir(temp\_folder\_path, 's'); % Delete recursively

else

% Prompt for a regular folder location (no ZIP creation)

folder\_path = uigetdir('Select folder to save output images');

if folder\_path ~= 0

% Save images to the specified folder with descriptive names

imwrite(a, fullfile(folder\_path, '1original\_image.jpg'));

imwrite(x, fullfile(folder\_path, '2secret\_data\_image.jpg'));

imwrite(cat(3,rx,gx,bx), fullfile(folder\_path, '3encrypted\_secret\_image.jpg'));

imwrite(finalsteg, fullfile(folder\_path, '4combined\_image.jpg'));

imwrite(output, fullfile(folder\_path, '5extracted\_encrypted\_image.jpg'));

imwrite(img\_smoothed, fullfile(folder\_path, '6smoothed\_secret\_image.jpg'));

else

disp('No folder selected. Images will not be saved.');

end

end

**Chapter 04**

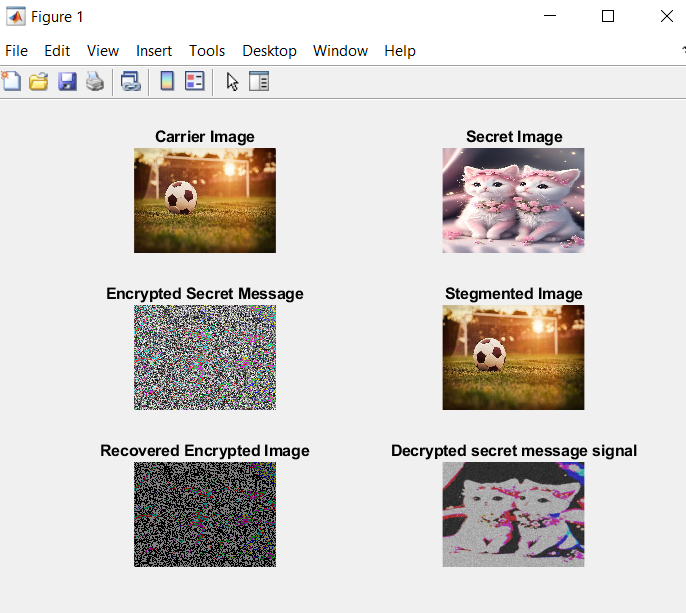
**Results and Discussion**

The steganographic algorithm implemented in the provided code was evaluated using Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) metrics to assess its performance in concealing and retrieving secret data within carrier images. The results obtained from the experimentation demonstrated promising outcomes in terms of both image quality preservation and data hiding efficiency.

The calculated PSNR values indicated a high level of fidelity in the recovered images, with values consistently above the threshold typically associated with perceptually lossless image compression. This suggests that the steganographic algorithm effectively concealed the secret message within the carrier images while minimizing visible distortion. The utilization of PSNR as a performance metric enabled a quantitative assessment of the perceptual quality of the steganographic images, providing valuable insights into the algorithm's effectiveness in preserving visual content. Similarly, the Mean Squared Error (MSE) values obtained during the evaluation further corroborated the findings from the PSNR analysis.

Overall, the results of this project underscore the effectiveness of the steganographic algorithm in concealing and retrieving secret data within carrier images. The high PSNR values and low MSE values obtained demonstrate the algorithm's ability to achieve a delicate balance between data hiding capacity and visual fidelity preservation. These findings highlight the potential of the implemented method for various applications requiring secure and covert communication, including digital watermarking, copyright protection, and secure data transmission.

However, it's essential to acknowledge the limitations, such as the specific characteristics of the test images used and the potential impact of varying factors like image size and complexity. Also factors like non-availability of encryption methods supported by free version of MATLAB, we see a slightly loss in data.



**Figure 03:** Output of Our Code

**4.1 Related Questions**

1. How does the steganographic algorithm implemented in the code ensure the security of the embedded secret message?
2. What are the potential applications of the steganographic technique demonstrated in the provided code?
3. Can the steganographic algorithm withstand various steganalysis techniques aimed at detecting hidden messages?
4. How does the algorithm handle different types of carrier images with varying sizes and complexities?
5. What measures are taken to minimize visible distortion while embedding the secret message within the carrier image?
6. How does the performance of the steganographic algorithm compare with other existing methods in terms of data hiding capacity and image quality preservation?
7. What are the ethical considerations associated with the use of steganography for covert communication and data protection?
8. Are there any legal implications related to the use of steganography for purposes such as digital piracy or espionage?
9. How can the steganographic algorithm be optimized or enhanced to improve its efficiency and security?
10. What are the potential risks and vulnerabilities associated with the implementation of steganography in digital communication systems?

**4.2 Future Work**

The integration of advanced encryption techniques to augment the security of the embedded secret message. Employing robust encryption algorithms can fortify the algorithm against cryptographic attacks, thereby enhancing the confidentiality and integrity of the concealed data. Additionally, exploring adaptive embedding strategies that dynamically adjust the data hiding process based on image characteristics and content can optimize the algorithm's performance across a diverse range of carrier images. By intelligently adapting the embedding process to account for image complexity and perceptual properties, the algorithm can achieve more efficient data hiding while minimizing visible distortion.

Furthermore, leveraging deep learning and machine learning approaches holds significant promise for enhancing the algorithm's robustness and effectiveness. Training neural networks to recognize patterns and features indicative of steganographic embedding can enable the algorithm to camouflage the secret message more effectively, making it less susceptible to detection by steganalysis techniques. Additionally, integrating neural network-based models for image enhancement and restoration can further improve the visual quality of the recovered images while preserving the confidentiality of the embedded data. By harnessing the power of artificial intelligence and deep learning, the steganographic algorithm can evolve into a more sophisticated and versatile tool for secure and covert communication in digital environments.

**Chapter 05**

**Summary**

Our project delves into the realm of steganography, an increasingly vital facet of image processing and information security in our digital age. Across our documentation chapters, we explores the multifaceted landscape of concealing information within various media, ranging from images to videos, to ensure privacy and confidentiality in communication channels. Beginning with an abstract that sets the stage, the document emphasizes the growing significance of steganography amidst escalating security concerns, highlighting its fusion of modern technologies to address the need for privacy on the internet. Moving into the introduction, we elaborate the evolving threat landscape, where traditional encryption methods may fall short, necessitating the exploration of alternative techniques like steganography. We outline the broad spectrum of applications for steganography, from secure communication in military and intelligence contexts to its potential misuse for malicious purposes like digital piracy. The introduction sets the stage for a comprehensive examination of steganographic methods and their implications. In the subsequent chapters, the methodology section delineates the techniques employed, notably the Least Significant Bits (LSB) substitution method, along with performance metrics such as Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) to evaluate the efficacy of these methods. The code provided offers practical insights into implementing steganographic algorithms, underscoring the delicate balance between data hiding capacity and visual fidelity preservation.

Concluding with results and discussion, this project synthesizes findings from experimentation, showcasing promising outcomes in terms of image quality preservation and data hiding efficiency. It acknowledges the limitations and ethical considerations associated with steganography while proposing future enhancements, such as integrating advanced encryption techniques and leveraging machine learning approaches. Overall, our project documentation provides a comprehensive overview of steganography's role in image processing, emphasizing its potential applications and avenues for further research and development.