



Examining the Application of Deep LSTM Neural Networks in Steganography of Textual Information in Digital Images

Mohammad Ali Yasmifar¹, Sattar Mirzakuchaki^{2*}, Mohammad Norouzi³

¹Department of electrical engineering, faculty of electrical and computer engineering, qazvin branch, islamic azad university, qazvin, iran

<https://orcid.org/0009-0005-3780-4488>

yasamimohammad@yahoo.com

²Electrical Engineering Department of Iran University of Science and Technology, Narmak, Tehran, Iran

<https://orcid.org/0000-0003-0232-9267>

m_kuchaki@iust.ac.ir

³Mechatronics Research laboratory, Electrical and Computer Engineering Department, Islamic Azad University, Qazvin, Iran

<https://orcid.org/0000-0002-9222-1358>

mh.norouzi@gmail.com

Abstract

Information security has emerged as a critical concern alongside the development of multimedia technology. Among the myriad security challenges, the secure transmission of sensitive information between parties has become a focal point of researchers. Encryption, involving mathematical techniques to ensure data security, is explored in this study. Specifically, the application of deep LSTM neural networks in concealing textual information within digital images is investigated. The approach involves embedding one image within another in a manner that prevents detection of the hidden image within the cover image, while textual content is covertly embedded within the image. The proposed method demonstrates superior performance based on three evaluation metrics—Peak Signal-to-Noise Ratio (PSNR) in decibels, Mean Squared Error (MSE), and accuracy rate in percentage—compared to three other benchmark images (lena.png, peppers.png, mandril.png, and monkey.png), achieving values of 93.665275 dB, 0.6945 MSE, and 97.23% accuracy, respectively.

Keywords: Neural networks, LSTM, digital images, steganography

Introduction

In contemporary societies, the increasing use of digital media has raised concerns about the security of these media files, particularly against users with malicious intent, a concern amplified in the context of the internet. Encryption involves the use of mathematical techniques to ensure data security. In other words, encryption is the process of altering the plaintext or information using a secret key and an encryption algorithm, allowing only the individual possessing the algorithm key to extract the original information from the encrypted data. In cases where secure exchange of encrypted information is challenging, covert communication becomes necessary. Indeed, steganography, or stegānographia, is a process where data is



concealed within other data formats such as image or text files. The overall process of steganography is depicted in Figure 1.

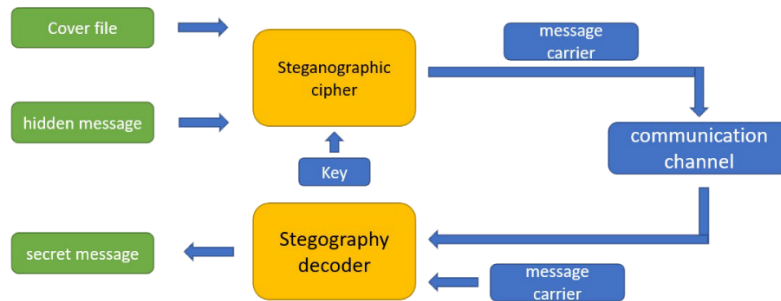


Figure 1: The general process of steganography

The approach of this research involves the steganography of textual information within images. This process entails embedding one image within another such that the hidden image cannot be observed within the cover image, and textual content is steganographed within the image. The primary challenge here is ensuring that the edges of the hidden image, or stego image, are not visible within the cover image, and accurately determining the textual area is not required. Steganography can be utilized to prevent the illegal dissemination of proprietary content and files. Additionally, encryption of sensitive information within an organization is crucial. For example, in a stock exchange organization, text messages may need to be steganographed to ensure secure communication from source to destination. Similarly, an organization might need to send an image containing encrypted information to a system where only the intended recipient can interpret the content, making steganography essential. While the goal of steganography is to conceal information and avoid detection, steganalysis, or the science of detecting hidden information, serves to uncover such concealed content. Steganalysis can be likened to the work of a detective.

and steganography is like the criminal. One tries to find the other (this does not imply that steganography is inherently bad; the analogy is simply for better understanding). Steganalysis attempts to uncover hidden information, but often the hidden texts concealed using steganography

software do not display any distinct markers. For instance, if several images are provided with the

goal of finding a hidden text within them, it must first be determined which image contains the hidden text, as there are no specific markers to identify it. Even if the original and the modified images are available, the differences are not easily detectable since they do not vary significantly

in appearance or size. Various generations of steganography software exist, with steganalysis being one type among them. Generally, steganographic methods are secure if the host or container

image does not exhibit detectable signs. In other words, the statistical properties of the host or container image should be similar to those of the cover image. The ability to detect the hidden message depends on the length of the hidden message. Clearly, the smaller the amount of



information embedded in an image, the less likely it is for detectable markers to appear. The choice of image format also significantly impacts the steganography system. Uncompressed formats such as BMP (Bitmap) provide ample space for steganography but are suspicious due to their large file size.

One of the classic methods in steganography is the Least Significant Bit (LSB) technique, which has various models developed for both image steganography and steganography in general. Due to weaknesses in the LSB methods and their developed models, newer methods need to be integrated with them. This opens a hot area in the field where new methods with higher speed, accuracy, and efficiency can be proposed. In this research, different methods for steganography will be reviewed, and alongside examining the basic algorithms, a new method based on steganography using a hybrid genetic algorithm and chaos theory is proposed. The proposed method involves using a genetic algorithm with three components for thresholding and selection. For embedding, the LSB method will be used. For data extraction, threshold values and widths derived from non-edge pixels of the stego image will be utilized, performed by the genetic algorithm. To extract these threshold values and widths from non-edge pixels of the stego image, a hybrid genetic-chaotic algorithm will be employed. The reason for using the chaotic method is that the issue of image steganography is highly sensitive to initial conditions, a characteristic also present in chaotic systems. Specifically, this research focuses on the Rossler map due to its high sensitivity to noise and initial conditions. The genetic algorithm is used to address the challenges and slow performance issues and to improve noise sensitivity in the LSB method.

Steganography has various models, including text in image, text in signal, text in video, image in image, image in video, signal in image, image in signal, and signal in video. This research focuses on text-based steganography within images. Essentially, several different images are considered, and the second image is used to store the texts from the first image. While this might require the principles of image processing, machine learning, especially deep learning algorithms, plays a more significant role. Hence, this research utilizes deep learning LSTM based on gate mechanisms in the input, hidden, and output layers for training. To validate and evaluate the proposed approach, evaluation metrics including Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), accuracy, sensitivity, and feature rate will be used.

Research Background

In (S. Khan and B. Tiziano 2018), a steganography method based on the ant colony algorithm was presented. In the proposed method, the ant colony algorithm is used for edge detection, and then confidential information is embedded in complex regions (edges) using the LSBF method. Complex regions are identified using a pheromone matrix, where each element of this matrix corresponds to a pixel of the image. The pheromone matrix is developed based on the movement of ants, which is determined by local differences between pixels and the intensity of the image pixels. To hide confidential information, the LSBs of the complex region pixels are replaced with bits of the message. One of the features of the proposed method is the ability to adjust the embedding capacity based on the chosen parameters.

Shah and colleagues (P.D. Shah and R.S. Bichkar, 2018) proposed a steganography method using a genetic algorithm. This method utilizes genetics for hiding the coefficient matrix



instead of the secret message itself. Instead of using the conventional LSB method for embedding into the image, it uses a data mapping method. Specifically, two bits of the message are stored in two bits of the cover image pixel, but these bits are placed in the best two bits of the pixel rather than in the LSBs. The results show that this method provides better PSNR and is more resistant to histogram analysis attacks compared to the conventional LSB method, which embeds 2 bits into the image.

Shah and colleagues (P.D. Shah and R.S. Bichkar, 2018) also proposed a high-capacity and visually acceptable image steganography method in the spatial domain, based on the genetic algorithm. In this method, two bits of the confidential data are embedded in each pixel of the cover image. Data embedding in the cover image is not performed sequentially; instead, a pseudo-random path is generated using the Linear Congruential Generator (LCG) function. The genetic algorithm is used to refine the parameters of LCG. Confidential data is modified using direction and polarity before embedding in the cover image, and finally, after embedding the entire message in the image, the Optimal Pixel Adjustment Process (OPAP) is applied to further enhance the quality of the steganographed image.

Thabti and colleagues (Thabti, S. S. Fayazi, H., Bahar, 1999) proposed methods where the main idea is to combine the LSB Matching (LSBM) method with the genetic algorithm, and all of them follow a uniform framework. The first proposed method is called GLSBM. The GLSBM method is essentially the LSBM method, but with the genetic algorithm used to decide whether to increase or decrease the value of pixels whose LSB does not match the message bit. The goal of the genetic algorithm in this method is to produce a steganographed image whose histogram has minimal differences from the cover image's histogram. To improve this method, the idea of repeating the GLSBM algorithm with different keys and finding the best key was proposed. By changing the key, the order of pixel selection for embedding changes.

In the research conducted by Wang and colleagues (Wang, Sh 2010), a method is presented that has the capability to embed confidential information into the carrier image such that it is resistant to RS attacks. This method uses the genetic algorithm. Initially, it works like the simple LSB method, embedding secret bits into the host image. Subsequently, the genetic algorithm adjusts the pixel values to ensure that the RS parameters meet the normal image conditions.

In the research conducted by Subramanian (Subramanian B, 2011), an algorithm based on an extended AES (Advanced Encryption Standard) is used, where the encryption process utilizes a unique bit or a set of pixels. The keys are independently used on the sender and receiver sides based on the AES key expansion process. AES provides high encryption quality with minimal memory and computational requirements.

According to Zhenhao Zhu et al., 2013, and Zohreh Fourouzesh and Jihad Al Jaam, 2015, adaptive edge steganography using the Sobel edge detection method was presented. Their research indicated that this method uses grayscale images for data embedding.

Mamta Juneja and Parvinder Singh Sandhu, 2013, focused on securing information with a two-component approach based on the LSB algorithm. This method randomly embeds hidden data into the blue and green pixels at the edges of the image. The overall approach utilizes adaptive LSB steganography, showing better performance than the AES encryption method. The PSNR measurement resulted in a value of 50 dB.

M. B. Ould Memdeni and El Mammoun Souidi, 2010, presented a method for information hiding using the spatial domain in grayscale images. This method segments the cover image



into non-overlapping blocks using pixel value variation and modifies the pixels to embed data. The results show increased embedding capacity, with a PSNR measurement of 42.68 dB.

P. Thiyagarajan et al., 2013, proposed a new high-capacity steganographic scheme using three-dimensional geometric modeling. Their method involves a triangular mesh network where information is hidden, resistant to operations like cropping, rotation, and scaling. A stego key is generated from the embedded messages. The PSNR measurement was 55 dB, and the Mean Squared Error (MSE) varied between 0 and 0.1.

Noor Kareem Jumaa, 2015, used randomized post-encryption text hiding with the LSB algorithm, generating pixels randomly for steganography. The input image is a grayscale image, and the proposed method incorporates several security features, including random encryption for key generation that selects pixels for hiding. Four steganography methods were used to hide 327,680 characters in a grayscale image.

M. Ghebleh and A. Kanso, 2014, proposed a robust steganography method for digital images based on chaos theory and the three-dimensional discrete wavelet transform (DWT). Unconventional outputs from mappings are used for message hiding in the cover image. The DWT ensures robustness in steganography, and the Swedish method for DWT application guarantees correct transformations. The proposed method is fast, efficient, effective, and flexible, with a PSNR measurement of 52.298 dB.

Anastasia Ioannidou et al., 2012, introduced a new high-capacity image steganography method based on edge detection. The method combines edge detection for embedding encrypted data in a color image. It integrates Laplacian and fuzzy edge detection methods, achieving a PSNR of 46.88 dB.

Hamidreza Rashidi Kanan and Bahram Nazeri, 2014, presented an image steganography scheme with high embedding capacity and balanced visual quality based on the genetic algorithm. Their research proposed a method with minimal data loss.

Hamidreza Rashidi Kanan and Bahram Nazeri, 2014, proposed an image steganography scheme based on the genetic algorithm in the spatial domain. The main idea of their method is to model the steganography problem as a search and optimization issue. Their approach aimed to increase data embedding capacity, achieving an average Peak Signal-to-Noise Ratio (PSNR) of 54.30 dB.

Methodology

The proposed method for data extraction involves four steps as outlined below:

Step 1: Separate the RGB components from the cover image and select the appropriate components, namely R for red, G for green, and B for blue, for data embedding.

Step 2: Extract the luminance intensities of the encrypted pairs using the shared key.

Step 3: Extract the message bits from the luminance intensities obtained in step 2 using the Rossler chaotic map based on the genetic algorithm.

Step 4: Convert the binary message bits into ASCII values and then convert the ASCII values into their corresponding characters to retrieve the original message.

To ensure the automatic processing of text in subsequent images using the proposed Rossler chaotic map-based algorithm with the genetic algorithm for enhancing the LSB method, a deep LSTM neural network is employed in a probabilistic structural manner, referred to as P-LSTM. P-LSTM, one of the variants of LSTM presented in this research, uses the cell state (c) instead of the hidden state (h) for regulating the forget gate, input gate, and output gate.



Figure 2 illustrates the internal connections of a P-LSTM unit, where the red arrows indicate the new probabilistic connections.

Figure 2: Internal Connections of a P-LSTM Unit](image_link_placeholder)

Key Points:

- P-LSTM provides a structural improvement over standard LSTM by utilizing cell state (c) for gate regulation.
- This method enhances the robustness and efficiency of text steganography in digital images.

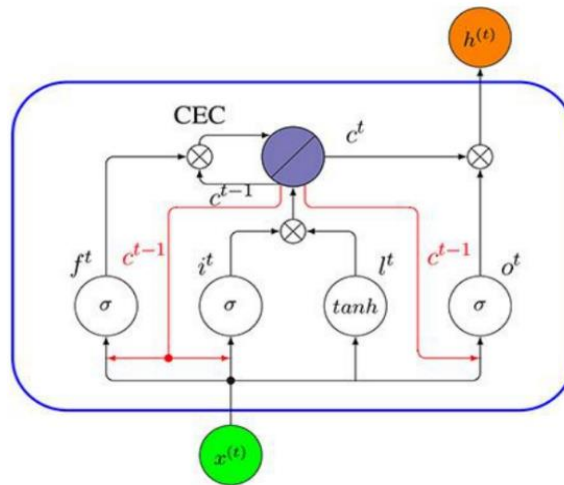


Figure 2: Structure of P-LSTM in this Research

The key difference between P-LSTM and standard LSTM lies in their input gate, forget gate, and output gate not utilizing (h_{t-1}) as input. Instead, these gates use the cell state (d_{t-1}) . To grasp the fundamental idea behind P-LSTM, consider that the output gate (h_{t-1}) in a traditional LSTM network is closed. Thus, according to the equation related to LSTM in Chapter 2, the output of the network at time $(t-1)$ will be zero, and in the subsequent step, the mechanism for adjusting all three gates depends solely on the network input (h_{t-1}) . Consequently, historical information will be entirely lost. A P-LSTM circumvents this issue by using the cell state instead of (h) as output to control the gates. Equations (3-3) to (8-3) formally describe a P-LSTM.

$$i_t = \sigma(X_i \cdot y_t + V_i \cdot d_{t-1} + c_i) \quad (1)$$

$$m_t = \tanh(X_m \cdot y_t + c_m) \quad (2)$$

$$f_t = \sigma(X_f \cdot y_t + V_f \cdot d_{t-1} + c_f) \quad (3)$$

$$o_t = \sigma(X_o \cdot y_t + V_o \cdot d_{t-1} + c_o) \quad (4)$$

$$d_t = f_t \cdot d_{t-1} + i_t \cdot m_t \quad (5)$$

$$h_t = o_t \cdot d_t \quad (6)$$



Classification Structure with P-LSTM

The classification structure using P-LSTM is depicted in Figure 10-3.

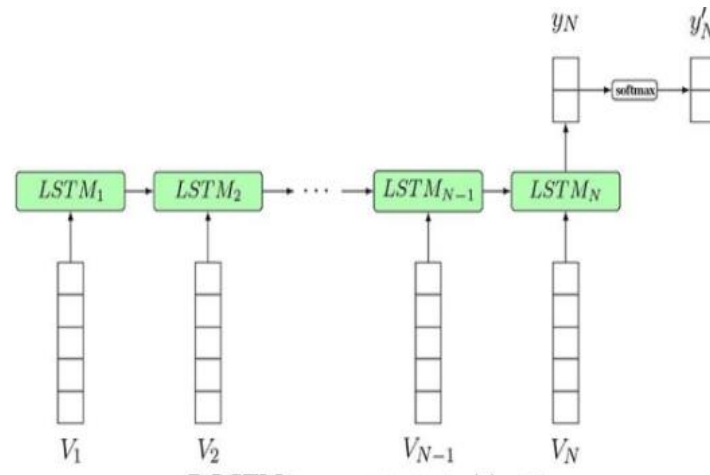


Figure 3: Classification Structure with P-LSTM

In Figure 3, the length of the input sequence corresponds to the number of cover images and texts for embedding. The input begins with a sequence of embedded vectors from the number of cover images, which are used as inputs to the model at different time steps. The prime (') represents the final result of embedding text into the image and placing texts for embedding into the cover image.

Simulation and Results Analysis

In the simulation phase, standard images such as Lena and other images are utilized to demonstrate the effectiveness of the proposed method for steganography of text in images. It is crucial that all target images are in PNG format and in RGB color space. The input image is depicted in Figure 4.

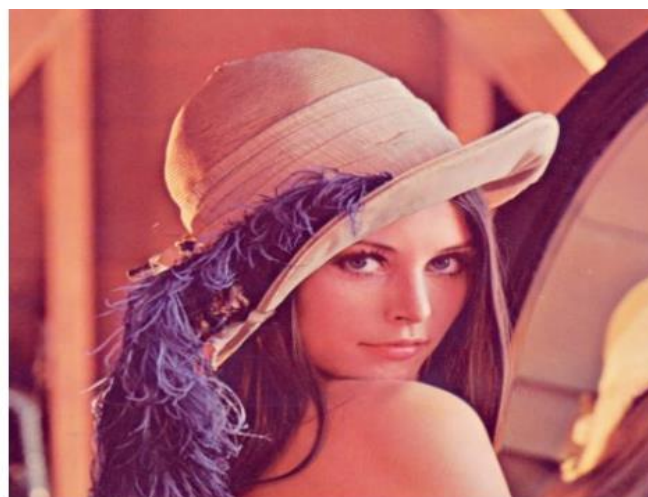


Figure 4: Input Image



After reading the image, the color channels of the image (R, G, and B) are separated. Then, a text "Nima" is chosen to be embedded into the cover image and made retrievable. The size of the message is designed such that each character occupies 8 bits. All values are encoded in ASCII from the chosen text and converted into binary format. Then, binary numbers are placed separately in distinct columns to determine the size of the binary message. Afterwards, all character arrays are transformed into numerical arrays. A color channel such as red (R) is selected for embedding the text into the image. A counter for the number of embedded bits is set to terminate after embedding the chosen text, and then the exit from the outer loop occurs once the entire embedded message is placed, and indicators for channels G and B are set for passage during embedding the chosen text. Then, the passage stage of the image is passed with the application and examination of the chaotic genetic algorithm to ascertain whether additional bits remain for embedding. At this stage, finding the least significant bit of the current important pixel with the Least Significant Bit (LSB) method and finding the second LSB for embedding two bits in the greenChannel is applied, and after filling all the columns, it moves to the next row. Likewise, one bit is embedded in the blueChannel and three bits in the greenChannel. Then, two bits in the blueChannel and two bits in the greenChannel are embedded, and again two bits in the blueChannel and three bits in the greenChannel, and again three bits in the blueChannel are embedded. Subsequently, the RGB color channels are connected, and the neural network as described in Chapter 3 is applied to automatically embed text using the Russell scrambling algorithm based on genetic algorithms to enhance the LSB method. For this purpose, a deep neural network LSTM is used as a probabilistic structure, known as P-LSTM. The output result on the image where the text is hidden is shown in Figure 5.



Figure 5: Output Result with Hidden Text in the Image

Visually, there is no observable difference between the cover image and the output image. However, the change in the image can be determined by its file size. The size of the input image in Figure 4 is 463 kilobytes, while the steganographically embedded text image in Figure 5 is 465 kilobytes.

Figure 6: Histogram Display During Application of Chaotic Genetic Algorithm Using Russell Mapping

This figure illustrates the histogram during the application of the chaotic genetic algorithm using Russell mapping.

Figure 7: Histogram Display During Application of P-LSTM Neural Network



This figure shows the histogram during the application of the P-LSTM neural network.

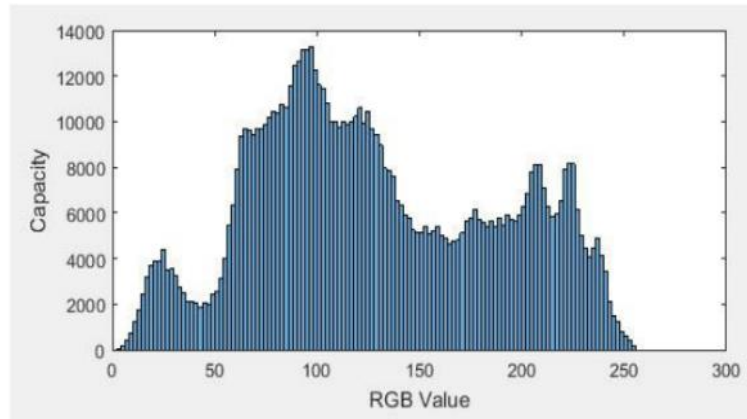


Figure (6): Histogram Display During Application of Chaotic Genetic Algorithm Using Rasselr Mapping

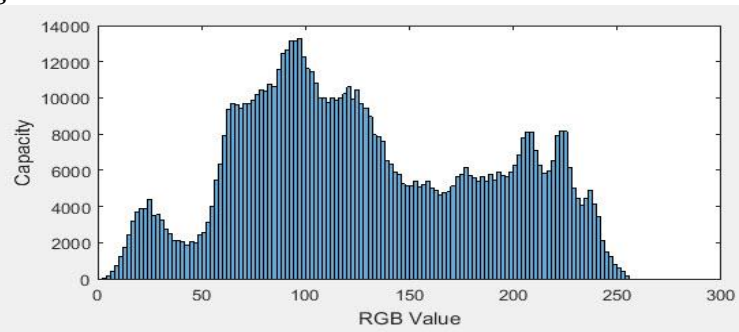


Figure (7) :illustrates the histogram display during the application of the P-LSTM neural network

To display a histogram during the application of the P-LSTM neural network, text is extracted by separating the color channels R, G, and B from the image. Then, the text stored in the image lena.png in Figure (7) must be extracted. Subsequently, binary numbers are separately placed in a separate column, and the size of the binary message is determined. Then, all text character arrays are converted into numerical arrays. A color channel such as red (R) is intended for detecting text in the image. A counter is used to track the number of extracted bits. The step of passing through the image to identify and extract text in the desired image is performed, which is at the heart of the combined method. The condition is that if more bits are available, the remaining text is extracted. Then finding the least significant bit (LSB) of the current pixel by finding the second LSB means there are two bits in green and one bit in the blue channel, which is the focus of the exploration. Then LSB pixel is stored in the Extracted_bits. Then the power of two is used to obtain the binary value of the hidden ASCII character, and all bits are obtained in an 8-column table to identify each line of hidden character bits. The extracted bits are converted to characters by multiplying by powers of 2 and print the hidden text which results in 'Nima'. Figure (8) refers to displaying a histogram during the text extraction process using the proposed method of this research.

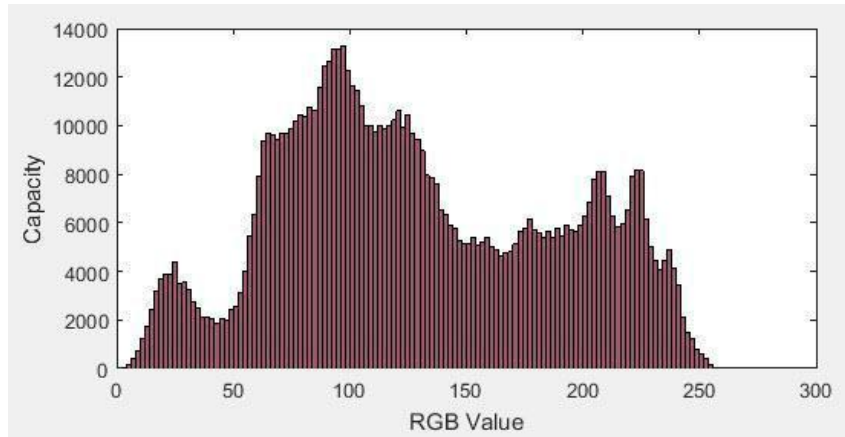
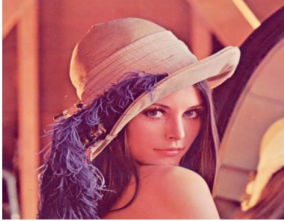

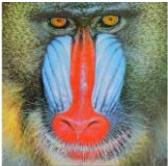



Figure (8): shows the histogram display during text extraction using the proposed method

Evaluation results of the proposed method on various images with the text "Nima" saved in each are presented in Table (1).

Table (1) presents the evaluation results of the proposed method.

Image	PSNR (dB)	MSE (Mean Squared Error)	Accuracy (%)
lena.png 	92.1738	0.394	97.50
peppers.png 	95.6261	0.178	97.84
mandril.png 	92.3265	0.190	96.42



 <p>monkey.png</p>	94.5347	0.228	97.16
Average	93.4306	0.209	96.79

dB 92.3265 mandril.png

Based on the obtained results, the average of each evaluation metric was determined, demonstrating superior performance compared to previous methods in Table (2).

Table (2): Comparison of results with previous methods

Accuracy rate in percentage	Mean Squared Error (MSE)	Peak Signal-to-Noise Ratio (PSNR) in decibels (dB) Peak Signal-to-Noise Ratio (PSNR) in decibels (dB)	Reference
% 97.23	0.6945	dB 93.665275	Proposed Method
% 95.72	0.6971	dB 92.16	Lingamallu Naga Srinivasu, and Vijayaraghavan Veeramani, 2022
% 96.68	0.6837	dB 93.10	A. Yousefian Darani, et al, 2023
% 96.73	0.6949	dB 92.78	Keshav Kaushik, and Akashdeep Bhardwaj, 2021

Summary:

Information security has become increasingly crucial with the advancement of multimedia technology. Among various security issues, the secure transmission of confidential information between parties has garnered significant attention from researchers. Currently, text messages, images, recorded sounds, and video files are all used for secure information transmission. To covertly transmit such information, steganography plays a vital role. Steganography hides data within media covers in such a way that the hidden information remains undetectable to anyone not aware of its presence. It focuses on the performance of embedded data and the imperceptibility of hidden information, ensuring that the embedded information is invisible both to future analysts and human eyes.

Steganography typically involves two main components: the cover image and the hidden information that needs to be concealed. The hidden data can include text, images, videos, or



audio information. Traditional steganography algorithms embed confidential information by modifying statistical features of carriers. Methods using images as carriers can be categorized into spatial domain methods, which vary based on the location of the modified cover. Additionally, spatial domain methods like LSB (Least Significant Bit), WOW, HUGO, and S-UNIWARD are commonly used. Furthermore, transform domain methods with stronger robustness and higher security, such as DWT (Discrete Wavelet Transform), DCT (Discrete Cosine Transform), IWT (Integer Wavelet Transform), etc., are employed. Transform domain directly modifies source image components (pixels) or its sub-bands, potentially reducing the perceived quality of the source image, which may mislead cryptographic analysts into assuming that the data is hidden.

In the proposed approach, it was observed that three evaluation metrics: PSNR (Peak Signal-to-Noise Ratio) in dB, MSE (Mean Squared Error), and Accuracy rate in percentage, with average values across four images (lena.png, peppers.png, mandril.png, and monkey.png) of 93.665275 dB, 0.6945 MSE, and 97.23% Accuracy, respectively, exhibit superior performance compared to three other references

Reference:

1. Anastasia Ioannidou, Spyros T. Halkidis, and George Stephanides. "A novel technique for image steganography based on a high payload method and edge detection." *Expert Systems with Applications*, Vol. 39, pp. 11517-11524, 2012.
2. Chia-Chun Wu, Shih-Jen Kao, Ming-Shi Hwang. "A high quality image sharing with steganography and adaptive authentication scheme." *Journal of Systems and Software*, Vol. 84, Issue 12, pp. 2196-2207, 2011.
3. Deepali Singla, and Mamta Juneja. "An Analysis of Edge Based Image Steganography Techniques in Spatial Domain." *Proceedings of 2014 RAECS UIET Punjab University Chandigarh*, pp. 6-8, 2014.
4. Hamidreza Rashidi Kanan, and Bahram Nazari. "A Novel Image Steganography Scheme With High Embedding Capacity and Tunable Visual Image Quality Based on Genetic Algorithm." *Expert Systems with Applications*, Vol. 41, pp. 6123-6130, 2014.
5. M. B. Ould Memdeni, and El Mammoun Souidi. "A Generation of the PVD Steganographic Method." *International Journal of Computer Science and Information Security*, Vol. 8, No. 9, 2010.
6. M. Ghebleh, and A. Kanso. "A robust chaotic algorithm for digital image steganography." *Communications in Nonlinear Science and Numerical Simulation*, Vol. 19, pp. 1898-1907, 2014.
7. Noor Kareem Jumaa. "Hiding of Random Permuted Encrypted Text using LSB Steganography with Random Pixels Generator." *International Journal of Computer Applications*, Vol. 113, No. 13, pp. 20-27, 2015.
8. P. Thiyagarajan, V. Natarajan, G. Aghila, V. Pranna Venkatesan, and R. Anitha. "Pattern Based 3D Image Steganography." *3D Research Center, Kwangwoon University and Springer, 3DR Express*, 2013.
9. P.D. Shah and R.S. Bichkar. "Genetic Algorithm Based Imperceptible Spatial Domain Image Steganography Technique with High Payload Capacity." *International Journal of Recent Technology and Engineering (IJRTE)*, Vol. 7, No. 5, 2019.



10. S. Khan and B. Tiziano. "Ant Colony Optimization (ACO) based Data Hiding in Image Complex Region." *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 8, No. 1, pp. 379-389, 2018.
11. V. Lokeswara Reddy, A. Subramanyam, and P. Chenna Reddy. "Implementation of LSB Steganography and its Evaluation for Various File Formats." *International Journal of Advanced Networking and Applications*, Vol. 02, Issue 05, 2011.
12. Zhenhao Zhu, Tao Zhang, and Baoji Wan. "A Special Detector for the Edge Adaptive Image Steganography Based on LSB Matching Revisited." *10th IEEE International Conference on Control and Automation (ICCA) Hangzhou, China, June 12-14, 2013.*
13. Zohreh Fouroozesh, and Jihad Al jaam. "Image Steganography based on LSBMR using Sobel Edge Detection." *Third International Conference on e-Technologies and Networks for Development (ICeND)*, 2014.