

An Enhanced Steganography Technique for Hiding Data in Digital Images Based on Lsb Substitution Method

T.S.R Krishna Prasad¹, P. Varun², T. Sri Dattasai Praveen³, P. Kusuma⁴, K. Vijay Ramprasad⁵

¹Associate Professor, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India

^{2,3,4,5}Under Graduate Student, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India

Abstract

Technology has advanced further recently, and data security is now crucial for communication. Steganography is a method for securely communicating with others by concealing confidential information in images. Steganography is a method for securely communicating with another person by concealing confidential information in images. This study suggests an Enhanced steganography approach focused on the LSB substitution procedure. The proposed method utilizes Huffman coding to encode the message difference, then encrypt bits, convert to gray code and embedding the encoded data into a randomly selected pixels in image channel. Experimental results demonstrate that the proposed approach achieves reduced distortion, improved visual quality, enhanced security, and increased embedding capacity when compared to another method. These findings underscore the potential of the proposed technique for enabling robust and secure data communication.

Keywords: Data Hiding, Image Steganography, Huffman coding algorithm, LSB, Gray Code, Information Hiding.

1. INTRODUCTION

In an era defined by widespread digital communication and data exchange, making sure data is secure is a critical concern. There are several techniques for communicating data securely such as cryptography and steganography. Steganography, with origins in ancient Greece, is the practice and discipline of communicating in such a way that a spectator would be unable to determine that communication is happening.

Steganography and cryptography are two related but independent approaches for securing data. Encryption transforms data into an unreadable format, thereby preventing unauthorised access to the content. However, the presence of an encrypted message is evident, which may attract unwanted attention. In contrast, steganography focuses on concealing the very existence of a message, making it difficult to detect. Steganography does not replace cryptography but supplements it.

Steganography can be applied to various types of digital media, including image, audio, video, and text

files, as well as network protocols. In image steganography, secret data is embedded within an image. Audio steganography conceals information within audio signals. Video steganography can embed vast amounts of data within digital video formats. Network steganography uses network protocols to embed secret information.

Image steganography techniques are primarily grouped into spatial-domain based approaches and transform domain based approaches[1]. Spatial domain methods, that includes LSB replacement steganography method [2, 3, 4] and Pixel Value Differencing (PVD), directly manipulate the pixels to insert the message in the cover image. In contrast, transform domain approaches use transforms like the Discrete Cosine Transform (DCT) to alter an image in the frequency domain.

LSB substitution is the popular steganographic method due to its simplicity, imperceptibility, and high capacity. The LSB Replacement[2] approach can increase robustness in communication while minimizing the error rate in embedding procedure. Recent techniques in LSB steganography focus on enhancing security and imperceptibility.

To improve security, encryption can be integrated into the steganographic process. For example, the secret message can be encrypted using algorithms like Vigenère cipher[5], Blowfish, or the MLE Encryption Algorithm[2] prior to being inserted into the cover image. Other techniques involve shuffling the order of pixels[6] using methods like the Magic Matrix or Gray code to increase imperceptibility. Furthermore, bit-flipping methods can be employed to increase imperceptibility by decreasing variations in the cover image's bit values.

2. Literature Survey

The simple LSB Replacement steganography is well-known for its relatively simple method of substituting some of the Right most bits of pixel values in a cover image with other secret data bits in order to insert important data.

The simple LSB Replacement steganography method embeds messages directly through manipulation of pixel value LSB's according to the binary representation of the secret message. It is popular because of its imperceptibility and relatively large message storage capacity. However, standard LSB steganography is vulnerable to hacking. The challenge lies in enhancing security without sacrificing image quality or embedding capacity. There are several methods that increase the imperceptibility of this method.

The NLSB Technique[2] is an, enhanced, and improved Stego algorithm that gives a superior security using the techniques such as embedding difference between Red Channel and Secret message, Shuffling channel with Magic Matrix and furthermore, encryption with MLEA and. But, the security and this method's performance could be further enhanced.

Bit Flipping Methods [3, 7] and Inverted LSB Methods [8] are another approach to improve LSB steganography, which can make it harder to detect of message in a Stego image. When more than 50% of the pixel values in the cover image have been changed, this method replaces the message bit values with the bit's inverted value. However, when bit flipping was implemented and employed in the Novel LSB Algorithm, it did not seem to improve any image quality metrics like MSE, PSNR, SSIM, or RMSE.

To improve the information security and imperceptibility of the message, several layers of confusion can be added in the embedding process. There are various ways we can do this such as using Encryption, Compression etc.

Encryption is the most basic confusion step we can add in steganography to improve the security and imperceptibility of the method. Even if an attacker extracts an LSB message, it will be more difficult for them to decipher the underlying message if binary form of ascii values of the secret message are encrypted before embedding.

The High capacity gray code Algorithm[9] uses Blowfish Encryption to encrypt the message before it is placed in pixel values. This is secure, but takes more time for computation. The Novel LSB Technique[1] uses the MLEA Encryption [5] technique for encrypting the binary representation of the message by this uses just bunch of xor, shifting operations. This is much more computationally efficient while being as secure as possible.

Another Technique to improve the imperceptibility of the message is to shuffle the order of pixels[6] while embedding and unshuffling them. Novel LSB Technique uses the magic matrix to shuffle the channel before embedding. The Gray code Shuffling[9] can also be utilized to shuffle order of pixels in channel.

The order of the pixels in which message bits are inserted can be jumbled by shuffling using gray code ordering [2] of blocks of pixels. The values from N-bit Gray code generation are used in this approach to identify the locations of embedded message bits[9]. A bit-cycling mechanism like this makes it possible to offer a better level of security.

To improve embedding capacity of LSB techniques, we can compress the message data before embedding. Huffman coding [4] can be used to pre compress the data before embedding, thus reducing the amount of data to be embedded and potentially improving imperceptibility and capacity. A lossless data compression technique called Huffman coding represents symbols according to their frequency using variable-length codes. More frequent symbols are represented with fewer bits, while less frequent ones use more bits, effectively reducing the overall data size.

A steganography method using both cryptography and data compression techniques improves the secrecy of information through implementation of several levels of security stages.

3. Proposed Method

We Proposed a method to enhance the novel LSB Steganography method. This is done by first converting message to Huffman bits and then after encryption of these Huffman bits, encrypted bits are converted to gray code. This method ensures that the imperceptibility, security of the method is improved, while also improving the image metrics and the embedding capacity for the method.

In this section, we will demonstrate the Enhanced LSB Steganography method for concealing the hidden message within images. This method is mainly divided into two phases, they are embedding phase and extraction phase.

During embedding process, the message is first converted to ascii values, then these ascii values are converted to Huffman bits, then these bits are encrypted and converted to gray code and finally embedded into shuffled blue channel of the image.

The algorithm for extraction phase is exact reverse process to embedding phase and this ensures the secure embedding and extraction of data.

3.1 Embedding Algorithm

The steps to follow for hiding secret message into a cover image are:

Step 1: Take the cover image, Flip and Transpose the image then divide it into Red channel(Rc), Blue channel(Bc) and Green channel(Gc).

Step 2: Convert secret message characters into ascii values, then take the difference of the message Ascii values and the Red channel pixel values of the image. Let this difference value be SMdv.

Step 3: The SMdv values are compressed with Huffman encoding and to generate the Huffman encoded bits.

Step 4: The converted Huffman bits are then encrypted with MLEA Encryption Algorithm and converted to gray code.

Step 5: The blue channel is segmented into four square shaped blocks, which are randomized using a magic matrix.

Step 6: Gray code bits of secret message are Embedded into the Blue channel blocks cyclically, 1 bit a block for every four bits.

Step 7: The shuffled channel is now unshuffled and all channels are combined, flipped and transposed to give Stego Image as final Output.

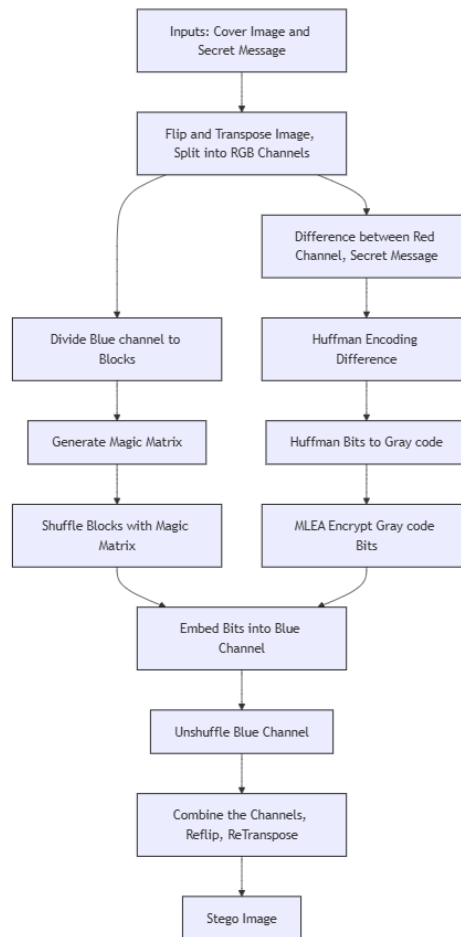


Figure 1: Embedding Algorithm

3.2 Extraction Algorithm

The steps for process of extracting secret message from the Stego image are:

Step 1: Take the Stego image, Flip and transpose it, then divide into Red Channel(Rc), Blue channel(Bc) and Green Channel(Gc).

Step 2: Divide Blue channel (Bc) into four equally sized square blocks and shuffle them using the magic matrix function in MATLAB.

Step 3: Extract the Gray code bits from these shuffled blocks in the cyclical manner, one bit from each block for every 4 bits.

Step 4: Convert the Gray coded bits sequence to Huffman encoded binary bits.

Step 5: Decode these Huffman encoded binary bits to recover the Secret Message difference values (SMdv values).

Step 6: Convert the SMdv values to the Secret message Ascii values by taking difference of SMdv values and the Red Channel (Rc).

Step 7: Finally, convert these Secret message ascii values to the character sequence to recover the secret message.

Step 8: Display the output Secret message.

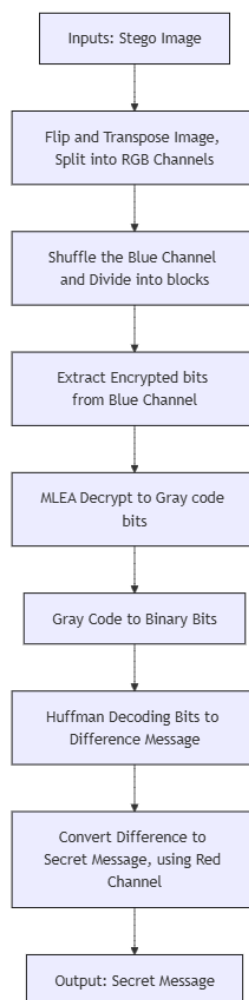


Figure 2: Extraction Algorithm

4. Performance Evaluation and Results

The experimental results pertaining to the suggested Enhanced LSB Steganography Method will be discussed in this part along with a thorough analysis. All experiments on proposed method were conducted utilizing MATLAB 2019a on a 64-bit Windows 11 operating system. The proposed methodology was analyzed using a diverse set of images obtained from the Div2k image dataset, along with the benchmark USC-SIPI Dataset, in addition to various supplementary test images. For the

purpose of consistency, all images were resized, adjusted, and cropped to the dimensions of 512x512 pixels.

 Baboon	 Barbara	 GEC	 Female
 House	 House2	 Lake View	 Lena
 Pebbles	 Peppers	 Plane	 Ramdan

Figure 3: USC-SIPI Dataset Images and Other Images











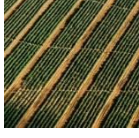

 Starfish	 Butterfly	 Tiger	 Train
 Building	 Husky	 Parliament	 Coral
 Door-Lock	 Castle	 Farm	 Fern

Figure 4: Div2k Dataset Images and other Images

The effectiveness of the Enhanced LSB method was assessed using a variety of Image Quality metrics: Peak Signal-to-Noise Ratio (PSNR), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Structural Similarity Index (SSIM). These metrics provide a comprehensive evaluation of both the imperceptibility and visual impact on the Stego images.

4.1 Image Quality Assessment Metrics

We used the following metrics to assess the robustness and reliability of Enhanced LSB method:

Mean Squared Error (MSE):

The mean of all squared disparities between the cover image's and the Stego image's pixel values is measured by the Mean Squared Error. A lower MSE indicates that the embedded message has caused minimal distortion, making it a crucial metric for assessing the imperceptibility of steganographic methods. The MSE can be calculated as:

$$MSE = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (C(m, n) - S(m, n))^2 \tag{1}$$

Peak Signal-to-Noise Ratio (PSNR):

The PSNR measures the imperceptibility of the Stego image in comparison to the cover image. Higher PSNR values indicate that the Stego image is visually almost identical to the cover image. Generally, a PSNR above 30 dB is considered acceptable for human perception. The PSNR can be calculated as:

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \tag{2}$$

Root Mean Squared Error (RMSE):

The RMSE offers insights into the average magnitude of error and is measure by taking square root of mean of all squared errors. A lower RMSE signifies that the pixel-wise error between cover image pixels and corresponding pixels in Stego images is minimal. The formula for calculation of RMSE is:

$$RMSE = \sqrt{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (C(x, y) - S(x, y))^2} \tag{3}$$

Structural Similarity Index (SSIM):

The SSIM metric evaluates the compositional similarity between the cover and Stego images, focusing on luminance, structure and also the image contrast. An SSIM value approaching 1.0 indicates that the structural integrity of the image has been effectively maintained. The SSIM can be calculated as:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{4}$$

4.2 Effectiveness of Proposed Method

The results for proposed Enhanced LSB method on images from the USC-SIPI Dataset are presented in Table 1, and results for images from the Div2k dataset are presented in Table 2. The performance was compared against the traditional NLSB method across all four image assessment metrics.

The proposed method demonstrates a substantial improvement in PSNR by 3.5 to 4 dB, indicating higher imperceptibility. The MSE was reduced by approximately 50%, suggesting that the embedded message caused significantly less distortion. The RMSE values also decreased by 30-40%, ensuring that pixel-wise errors were minimized. The SSIM values remained extremely high, confirming that the structural integrity of the images was well-preserved.

Table 1: Image metrics on USC-SIPI and additional images for the proposed method.

Image	PSNR		MSE		RMSE		SSIM	
	NLSB	Proposed	NLSB	Proposed	NLSB	Proposed	NLSB	Proposed
Baboon	75.75	79.03	0.00172	0.00081	0.0415	0.0284	0.999995	0.999996
Barbara	75.61	79.57	0.00178	0.00081	0.0422	0.0285	0.999986	0.999991

GEC	75.58	79.05	0.00179	0.0008	0.0423	0.0284	0.999976	0.999985
Female	75.70	79.57	0.00174	0.00071	0.0417	0.0267	0.999985	0.999991
House	75.67	79.03	0.00175	0.00081	0.0419	0.0285	0.999986	0.999992
House2	75.72	79.93	0.00173	0.00066	0.0417	0.0256	0.999989	0.999994
Lakeview	75.90	79.36	0.00167	0.00075	0.0408	0.0274	0.999985	0.999991
Lena	75.63	79.04	0.00177	0.00081	0.0421	0.0284	0.999996	0.999997
Pebbles	75.45	79.94	0.00185	0.00065	0.0430	0.0256	0.999985	0.999992
Peppers	75.88	79.07	0.00167	0.00080	0.0409	0.0283	0.999997	0.999998
Plane	75.72	79.28	0.00174	0.00076	0.0417	0.0276	0.999945	0.999969
Ramdan	75.79	79.50	0.00171	0.00072	0.0413	0.0270	0.999949	0.999969

Table 2: Image metrics for the Proposed Method on the Div2k dataset images.

Image	PSNR		MSE		RMSE		SSIM	
	NLSB	Proposed	NLSB	Proposed	NLSB	Proposed	NLSB	Proposed
Starfish	75.60	79.31	0.00179	0.00076	0.0423	0.0275	0.999985	0.999991
Butterfly	75.78	79.57	0.00171	0.00071	0.0414	0.0267	0.999994	0.999997
Tiger	75.79	79.32	0.00171	0.00076	0.0413	0.0275	0.999989	0.999993
Train	75.58	79.18	0.00179	0.00078	0.0280	0.0423	0.999987	0.999992
Building	75.66	78.85	0.00176	0.00084	0.0420	0.0290	0.999992	0.999995
Husky	75.52	79.32	0.00182	0.00076	0.0426	0.0275	0.999928	0.999956
Parliment	75.66	79.14	0.00176	0.00079	0.0419	0.0281	0.999993	0.999996
Coral	75.53	79.29	0.00181	0.00076	0.0426	0.0276	0.999975	0.999985
Door-Lock	75.68	78.94	0.00175	0.00082	0.0419	0.0288	0.999988	0.999992
Castle	75.60	78.93	0.00179	0.00083	0.0423	0.0288	0.999994	0.999996
Farm	75.63	79.04	0.00177	0.00081	0.0421	0.0284	0.999994	0.999996
Fern	75.53	79.09	0.00181	0.00080	0.0426	0.0282	0.999979	0.999987

Comparison of PSNR for NLSB and Proposed Method

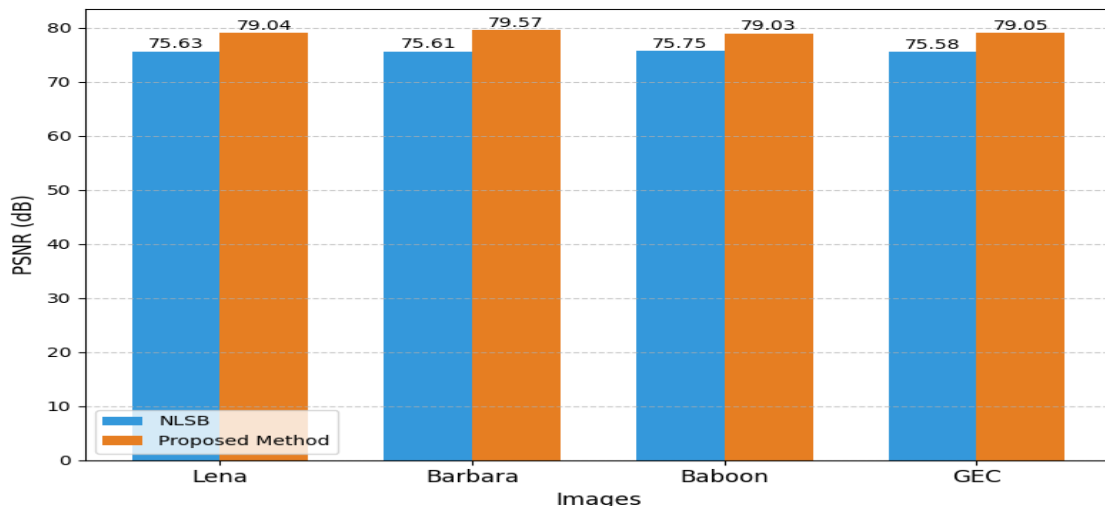


Figure 5: Comparison of PSNR values for proposed and NLSB Method

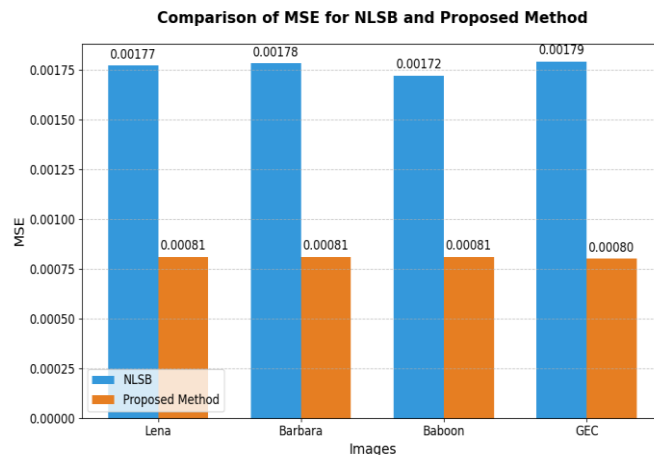


Figure 6: Comparison of MSE values for proposed and NLSB Method.

For the Div2k dataset, the proposed method achieved PSNR values between 78.85 dB and 79.57 dB, showing a consistent improvement of +3.5 to +4 dB over NLSB. The MSE was reduced approximately by 56%, demonstrating the effectiveness of Huffman encoding and Gray Code in minimizing pixel changes. The RMSE showed a reduction of 33.8%, confirming improved pixel-wise accuracy. The SSIM values, which were nearly 1, highlight that the proposed method preserved the structural and perceptual quality of the images exceptionally well.

5. Conclusion

The experimental results on method justify that the proposed Enhanced LSB Method not only outperforms the traditional NLSB method but also ensures high imperceptibility, minimal distortion, and superior security. The integration of Huffman Encoding, MLEA Encryption, and Gray Code Conversion has proven to be highly effective in achieving higher PSNR and lower MSE values across various image datasets. This confirms that the proposed method is a robust and secure solution for image steganography.

References

1. F. Qasim Ahmed Alyousuf, R. Din, A. J. Qasim, “Analysis review on spatial and transform domain technique in digital steganography”, Bulletin of Electrical Engineering and Informatics, April 2020, 9 (2).
2. S. Rahman, J. Uddin, H. U. Khan, H. Hussain, A. A. Khan, M. Zakarya, “A Novel Steganography Technique for Digital Images Using the Least Significant Bit Substitution Method”, IEEE Access, 2022, 10, 124053–124075.
3. E. Z. Astuti, D. R. I. M. Setiadi, E. H. Rachmawanto, C. A. Sari, M. K. Sarker, “LSB-based Bit Flipping Methods for Color Image Steganography”, Journal of Physics: Conference Series, March 2020, 1501 (1), 012019.
4. J. C. T. Arroyo, “An Efficient Least Significant Bit Image Steganography with Secret Writing and Compression Techniques”, International Journal of Advanced Trends in Computer Science and Engineering, June 2020, 9 (3), 3280–3286.
5. K. Muhammad, J. Ahmad, H. Farman, Z. Jan, M. Sajjad2, S. Wook Baik, “A Secure Method for Color Image Steganography using Gray-Level Modification and Multi-level Encryption”, KSII

Transactions on Internet and Information Systems, May 2015, 9 (5).

6. K. Kordov, S. Zhelezov, “Steganography in color images with random order of pixel selection and encrypted text message embedding”, PeerJ Computer Science, January 2021, 7, e380.
7. S. Kamil, S. N. H. S. Abdullah, M. K. Hasan, F. A. Bohani, “Enhanced Flipping Technique to Reduce Variability in Image Steganography”, IEEE Access, 2021, 9, 168981–168998.
8. S. Rustad, D. R. I. M. Setiadi, A. Syukur, P. N. Andono, “Inverted LSB image steganography using adaptive pattern to improve imperceptibility”, Journal of King Saud University - Computer and Information Sciences, June 2022, 34 (6), 3559–3568.
9. A. Seif, W. Alexan, “A High Capacity Gray Code Based Security Scheme for Non-Redundant Data Embedding”, 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), IEEE, February 2020, 130–136.