An Edge Detection with LSB and AES Encryption Based Image Steganography

Sheikh Thanbir Alam
Faculty of Electronic Engineering & Technology
University Malaysia Perlis Perlis, Malaysia
Army War Game Centre, Bangladesh Army
Dhaka, Bangladesh
sk.tamim56@gmail.com

Munira Tabassum Mou
Department of Software Engineering
Daffodil International University
Dhaka, Bangladesh
muniramou.swe@gmail.com

Md Maruf Hassan*
Dept. of Computer Science & Engineering
Southeast University, Bangladesh
Washington University of Science & Technology
Virginia, USA
ancssf@gmail.com

Touhid Bhuiyan
School of IT
Washington University of Science and Technology
Virginia, USA
touhid.bhuiyan@wust.edu

2025 8th International Conference on Information and Computer Technologies (ICICT) | 979-8-3315-0518-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICICT64582.2025.00078

***Abstract*—The protection of confidential data has been a critical concern since ancient times. Steganography and cryptography are two key techniques used to enhance data security. Cryptography transforms confidential data into an unreadable format, while steganography conceals the very existence of the communication. Both are crucial for safeguarding information in today's rapidly expanding networks. This paper proposes a novel method for secure data transmission over unsecured channels. The process begins by encrypting the secret data using the AES algorithm, which ensures robust security even if the hidden communication is detected. The encrypted data is then embedded into a cover medium at its edge locations, identified using a Sobel edge detector, and placed with the help of a hash function. The method employs the 1-bit Least Significant Bit (LSB) technique, implemented in C#, offering high data hiding capacity and minimal distortion in the stego image. Experimental results demonstrate that the proposed approach effectively conceals data while maintaining the quality of the cover image, making it highly suitable for secure data transmission.**

***Index Terms*—Sobel edge detector; Steganography; Cryptography; LSB**

I. Introduction

The Internet provides an accessible medium for data transmission, making security a critical concern. Techniques like cryptography ensure message confidentiality, but concealing the existence of information is sometimes necessary. Steganography, derived from Greek (stego: "cover"; grafia: "writing"), achieves this through covert communication. Image steganography embeds secret messages within images, detectable only by intended recipients [1]. Unlike cryptography, which hides content, steganography hides the communication itself. A combined approach enhances security, with PNGs preferred for their lossless compression, preserving sharpness and compatibility. In the spatial domain, the least significant bit (LSB) technique is a common steganographic method. Existing methods often use pseudorandom generators, causing distortion in smooth image regions. The proposed position-adaptive scheme embeds data behind edges, reducing embedding rates and preserving quality. Historically, steganography dates back to 440 BC, with methods like wax-covered tablets or messages inscribed on shaved heads. Invisible ink, used during World War II, and microdots, developed by the Germans, were notable advancements [2]. The British allegedly hid messages in vaccine barrels during India's colonization. Safe communication has been crucial throughout history. From ancient methods to advanced algorithms, techniques have evolved to protect data and ensure its undetectability during transmission. Cryptography, as noted by, originates from the Greek term meaning "secret writing" and ensures confidentiality, granting access only to authorised recipients. With roots in ancient Egypt and Rome, it has become crucial for securing modern communications. According to cryptography converts plain text into cipher text using encryption algorithms and keys, making it unreadable to unauthorized users. As highlights, only recipients with the correct decryption key can restore the original data, ensuring secure exchange in the digital era. According to steganography, derived from steganos (covered) and graptos (writing), conceals data within media for secure communication. Unlike cryptography, which encrypts content, steganography hides the message's existence, embedding data in formats like images, audio, or video. Techniques like LSB and DCT offer distinct strengths and limitations. The pris-

This research was supported by the Center for Research, Washington University of Science and Technology, VA, USA.

2769-4542/25/$31.00 ©2025 IEEE
DOI 10.1109/ICICT64582.2025.00078

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.

oner's problem illustrates covert communication through cover media [1]. These methods have evolved for improved security and reliability. The Sobel method, detects edges by applying 3x3 masks to estimate image gradients, highlighting intensity contrasts such as edges. It effectively identifies transitions from light to dark. The Advanced Encryption Standard (AES), encrypts and decrypts data securely Operating on 128-bit blocks, it supports key sizes of 128, 192, or 256 bits and employs steps like SubBytes, ShiftRows, and MixColumns to resist attacks. AES is efficient in hardware and software, widely used in securing data. Integrating cryptography, steganography, Sobel edge detection, and AES encryption addresses data security and undetectability, achieving enhanced security.

II. Related Work

This section highlights previous works in steganography that have utilised various systems and methodologies. A range of distinct approaches has been developed for the secure communication of information. In 2019, Dhargupta et al. proposed a steganographic method based on fuzzy edge detection to effectively embed data within images. This technique utilised fuzzy edge detection to estimate a greater number of pixels where data could be hidden. Initially, the cover image was masked, and fuzzy edge detection was applied to retain edge information. The number of bits embedded in each pixel depended on whether the pixel was classified as a base pixel, in which case more bits were embedded. For pixels that were neither base nor background pixels, the amount of data embedded was determined by the Euclidean distance from the nearest edge pixel, calculated using a Gaussian function [3]. Also in 2019, Banik et al. introduced a data embedding technique within images using the Kirsch operator, known for its capability to identify maximum edge strength in various directions. Based on a threshold value of the Kirsch operator and the intensity of each pixel in the cover image, a scale with three levels was created. This scale guided the selection of 2, 3, or 4 least significant bit (LSB) layers for steganographic encoding. The edge value was shared with the intended recipient as a key [4]. Bhardwaj et al., in the same year, proposed a steganographic approach combining edge detection-based methods with the Optimal Pixel Adjustment Process (OPAP). This approach enhanced LSB substitution by incorporating edge detection techniques, allowing different regions within the cover image to store varying amounts of data. Complex or textured regions were less sensitive to alterations and could hide more data compared to smooth regions [5]. In 2020, Wang et al. presented a hybrid steganography technique utilising LSB substitution alongside Hamming law (HLAH). This method improved information security by integrating two steganographic techniques. Given that sharp regions of an image can withstand more alterations than smooth areas, more data was embedded in the edge regions, while only minimal data was embedded in smoother areas [6]. Delmi et al. also proposed a steganographic technique in 2020, employing Least Significant Bit Matching Revisited (LSBMR). This method targeted the edge regions of digital images using Canny edge detection, ensuring the embedded message remained visually imperceptible [7]. In the same year, Prasad and Pal enhanced the embedding capacity of cover images by embedding a greater number of secret message bits into the edge regions compared to the smooth regions. The embedding process was based on a modulus function, with the secret message bits secured using a stego-key. This scheme was tested on standard greyscale images [8]. Ayub and Selwal also contributed in 2020 with an improved image steganography method that embedded data within the edge pixels of the carrier image. Since edge pixels naturally vary from their neighbouring pixels, an intruder would have less suspicion regarding the presence of hidden data, enhancing security. This method employed various edge detection filters, such as Prewitt, Sobel, Laplacian, and Canny, alongside edge-based data embedding within the DCT domain. The proposed approach significantly reduced the size of the stego-image due to compression [9]. In another work [10], the authors proposed the KVL technique, where an RGB image was converted into a binary format and compressed using Run Length Encoding (RLE), achieving up to 35% compression. The compressed image was then encrypted using the Triple-DES algorithm and embedded within a cover medium for secure transmission. Lastly, in [11], an edge-based LSB embedding technique was introduced, using a greyscale image as input. The Canny edge detection method, which relies on mean and standard deviation parameters, was utilised to identify edge regions suitable for data embedding. In [12], the limitations of the LSB approach were addressed by utilising two BMP images, one dark and one light, where embedding was performed on corresponding bits of both images. In [13], the authors implemented data hiding within video frames, specifically in the region of interest. This approach involved splitting the video into frames, selecting the target region, and embedding the data, with the stego video then transmitted to the destination. In [14], data embedding was executed across all three colour components of an image (RGB). The process involved splitting the image into three matrices (red, green, and blue) and embedding data sequentially, generating a stego image ready for transmission. In [15], a combination of the hash-LSB and RSA algorithms was proposed to enhance security. The hash function was used to generate a pattern for embedding secret data in the LSB of the cover image, while RSA encryption prevented data leakage. In [16], the authors combined the RSA and DES algorithms to embed secret data within cover media. This hybrid approach aimed to mitigate the individual weaknesses of both methods, yielding improved results. In [17], a method was proposed to ensure secure storage and communication of medical data. The methodology employed a combination of steganography and cryptography as a lossless synthetic technique to address human error in medical centres. In [18], Alam et al. introduced a technique using an eight-directional pixel selection method for embedding secret data. Although the method achieved good PSNR with 1-bit LSB embedding, the static pixel selection was identified as a weakness. In [19], the authors proposed a basic LSB replacement technique,

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.

commonly used for its simplicity. While the method achieved better PSNR and low MSE values, it relied on a standard zig-zag technique, which could be easily exploited by intruders. To address these limitations, this study employs a 1-bit LSB approach in the spatial domain. Before embedding, the data is encrypted using the Advanced Encryption Standard (AES) with a 128-bit secret key, randomly generated using C#. For embedding, a Sobel edge detection method is used in conjunction with LSB, ensuring better performance in random permutations. PNG images, known for their superior visual quality, are used as cover images. The effectiveness of the proposed method is evaluated using quality metrics such as Peak Signal-to-Noise Ratio (PSNR), Mean Square Error (MSE), and Root Mean Square Error (RMSE) [20].

III. PROPOSED WORK

In this study, an automated two-layered secure data concealing algorithm for image steganography has been developed, leveraging 1-bit LSB and a Sobel-based pixel selection approach. The encryption system and steganography implementation will be discussed in detail in the following subsections.

A. Encrypting and decrypting the Secret Message

The proposed method encrypts user input using AES with a 128-bit key, chosen for its balance of speed, memory usage, and strong security [21]. AES involves ten rounds of operations: SubBytes (substitution), ShiftRows (cyclic shifting for diffusion), MixColumns (linear transformation for diffusion), and AddRoundKey (XORing with the round-specific key). After encryption, the ciphertext is embedded into the cover image. Decryption reverses these steps to restore the original data, ensuring data confidentiality and integrity. The AES operations are implemented in C#, enabling efficient encryption and decryption.

B. Embedding the Encrypted Message

The ciphertext is embedded into PNG-formatted images using a 1-bit LSB approach. The Sobel edge detection method is used to select suitable pixels, focusing on edge regions to enhance imperceptibility while maintaining data integrity. PNG images are specifically chosen due to their lossless compression, which preserves the embedded data without introducing visual artifacts. All embedding operations are executed in C# to ensure seamless integration and high performance. This approach guarantees robust data confidentiality and imperceptibility, providing an effective solution for secure image steganography.

C. Pixel Filtring Algorithm

Edge detection techniques can be broadly categorized into grading and Laplacian methods. The grading technique, used in methods like Roberts, Prewitt, and Sobel, detects edges by identifying sharp transitions in the image. In contrast, the Laplacian method, such as Laplacian of Gaussian or Marr-Hildreth, focuses on zero crossings in the image's expansion to find edges. While both methods highlight significant features, Sobel's grading technique is generally more effective in detecting sharp edges and key image features.

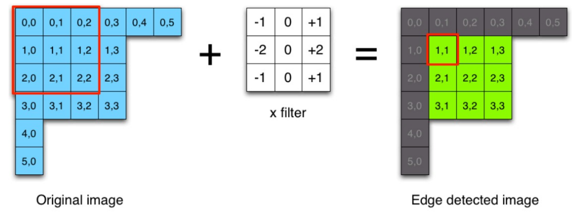


Fig. 1. Edge Detecting Technique

These formulae, which include gradient magnitude and gradient direction, apply the operation.

$$\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \tag{1}$$

$$\theta = \tan^{-1}\left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}\right) \tag{2}$$

These formulas calculate the gradient direction and magnitude of an image, which are crucial for determining edge orientation and intensity. The gradient direction shows the angle of the edge relative to the horizontal axis, while the gradient magnitude measures the edge's strength. These computations enable accurate edge identification, which is vital for selecting regions suitable for data embedding. The analysis of pixel detection from cover images demonstrates the effectiveness of an edge-based pixel selection strategy for identifying suitable regions for data embedding. For example, an image of Leena shows a pixel selection ratio of 27%, indicating relatively fewer edge regions. In contrast, an image of a baboon has a higher edge density with a pixel selection ratio of 58%. Additionally, an image of a group of parrots has a pixel selection ratio of 30%, reflecting moderate edge complexity. These results highlight the adaptability of the pixel selection method to varying image complexities, ensuring optimal embedding regions while maintaining image quality and imperceptibility.

D. Steganographic Process

The embedding strategy forms the first phase of the steganographic process, while the retrieval approach constitutes the second. The proposed system's embedding process, illustrated in Figure 2, begins with the encryption of user-provided secret plaintext using the 128-bit AES encryption method.

$$\text{1st-pixel position } (X_1, Y_1) = \left(\frac{W}{2} - 2, 1\right) \tag{3}$$

$$\text{2nd-pixel position } (X_2, Y_2) = \left(W, \frac{H}{2} - 2\right) \tag{4}$$

$$\text{3rd-pixel position } (X_3, Y_3) = \left(\frac{W}{2} + 2, H\right) \tag{5}$$

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.

$$\text{4th-pixel position } (X_4, Y_4) = \left(1, \frac{H}{2} + 2\right) \tag{6}$$

This method ensures a high level of security by converting plaintext into ciphertext, making it unintelligible to unauthorized users. The Sobel edge detection system is then employed to filter and identify specific pixel positions within the cover image. These positions serve as embedding locations for the secret data.During the embedding phase, the system interacts with the filtered pixel positions to insert the encrypted data bits. The secret data, once decoded, is transformed into 8-bit double data format.

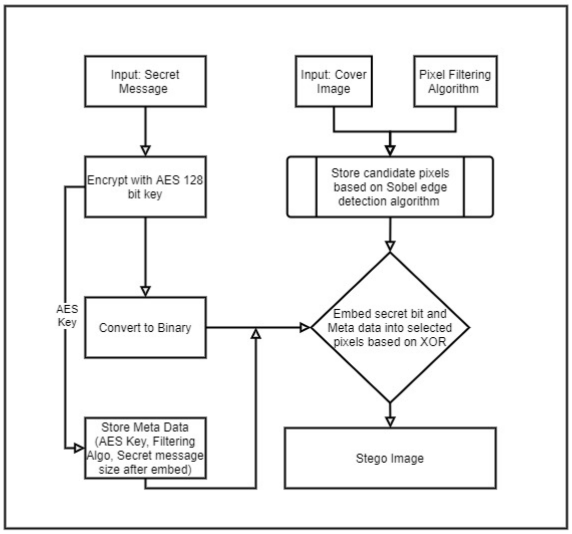


Fig. 2. Embedding Process

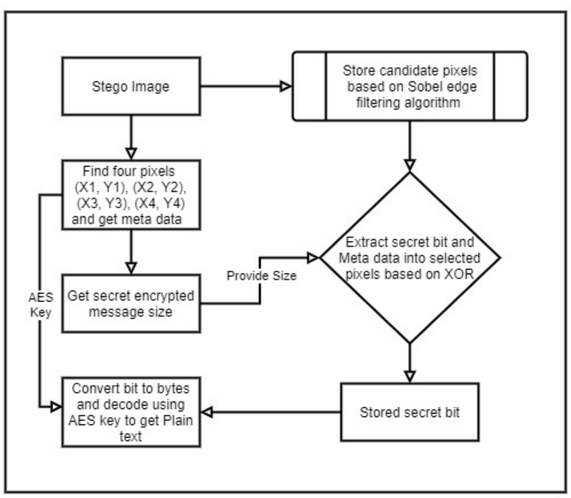


Fig. 3. Retrieving Process

This data is then paired with the 1-bit LSB positions of the filtered pixels using an XOR operation to securely embed the information. The final indexed RGB blocks are updated, with the secret data bit replacing the sixth indexed bit through the XOR operation, ensuring seamless integration of the data within the cover image. The retrieval process, as depicted in Figure 3, requires specific metadata to recover the hidden data. This metadata includes the secret data embedding key, the size of the hidden message, and the pixel filtering algorithm. The system uses this metadata to determine the fixed pixel positions, calculated through Equations (3), (4), (5), and (6), which correspond to specific pixel coordinates in the cover image. These fixed pixel positions enable the system to locate the filtering pixels where the secret message bits were embedded. The retrieval process involves performing an XOR operation on the sixth and seventh indexed bits of the RGB blocks to extract the embedded message bits. Once the message size is determined, the AES key is used to decrypt the ciphertext, recovering the original plaintext. This ensures a secure and reliable process for both embedding and retrieving sensitive data. Figure 2 illustrates the embedding process of the proposed steganographic method, which involves encrypting the secret data using AES, followed by Sobel edge detection to select suitable pixel positions for data embedding. This process ensures that the secret data is securely embedded within the cover image, leveraging edge regions to enhance imperceptibility while maintaining the quality of the stego image. Figure 3 outlines the retrieval process, demonstrating how the embedded secret data is extracted from the stego image. It involves identifying filtering pixels, extracting the secret bit using XOR operations, and decrypting the message with the AES key. This retrieval approach effectively recovers the original plaintext, ensuring data confidentiality and integrity even after embedding.

E. Algorithm for embedding and retrieving

The Sobel edge detection technique involves using a cover image, a pixel filtering algorithm, and a secret message during the embedding process.

Embedding

Result: Stego Image
$S_m \leftarrow$ input
$C_I \leftarrow$ input
E_K = AESKey();
ES_m = AESEncryption(S_m, E_K);
FilteredPixels[] = SobelEdgeDetect(C_I);
$M_D = F_{XB}(E_K + Size(ES_m))$;
(x1, y1), (x2, y2), (x3, y3), (x4, y4) $\leftarrow M_D$
$S_{MB} = F_{XB}(ES_m)$;
For a $\leq$ FilteredPixels[]
    RGB = Read(FilteredPixel[n]);
    embedLSBXOR(RGB, S_{MB})
    UpdatedRGB();
End For

Fig. 4. Algorithm for Embedding Process

The system loads the data into memory and constructs a 128-bit AES key for encryption. Pixels are filtered to hide se-

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.

Retrieving

Result: Secret Message
$S_I \leftarrow$ input
$M_D \leftarrow$ (x1, y1), (x2, y2), (x3, y3), (x4, y4)
FilteredPixels[] = SobelEdgeDetect(S_I);
$M_S = M_D$[Size(ES_m)]
For a $\leq$ FilteredPixels[]
    SecretBit[] = retrieveLSBXOR(RGB, S_{MB})
    if(SecretBit[] >= M_S)
        Break;
End For
encryptedMessage = BitToBytes(SecretBit[])
Plaintext = Decode(encryptedMessage, Key)

Fig. 5. Algorithm for Retrieving Process

cret bits in the cover image, ensuring secure transmission. The process follows a reverse procedure for decoding. Initially, the secret message is encrypted using AES, producing ciphertext that is secure from unauthorized access. Sobel edge detection identifies high-gradient areas in the image, and encrypted bits are embedded using the 1-bit LSB method. This results in a stego image that ensures secure data concealment without compromising visual quality, as shown in Figure 4 and Figure 5.

IV. Experiment and Result Analysis

This section provides a graphic explanation and assessment of the cover and stego picture to highlight the implications. Additionally, to support the efficacy, the outcomes of the proposed method are linked to other well-known steganographic techniques. The statistical analysis of the study is well satisfied with six quality dimension criteria that are comparable to mean square error (MSE), root mean square error (RMSE), and peak signal-to-noise rate (PSNR).

The experimental test uses three pictures in Table I (Baboon, Lena, and Parrot) that have been utilized in several works [22]. The three movies have 512 × 512 size and are in the PNG format. The NET Framework, a Microsoft.NET framework of C sharp programming, version 4.7.2, provides the continuation of the suggested system. Table II presents the quality metrics of the proposed method, including PSNR, MSE, and RMSE, for different payload sizes across various images.

Lena, Baboon, and Parrot were represented in this table using 512 × 512 pictures, with payload sizes of 512 bytes, 256 bytes, and 128 bytes progressively taken into account. The MSE values obtained by the suggested method were 0.0027, 0.0013, and 0.0007 for Lena, 0.0026, 0.0013, and 0.0006 for Baboon, and 0.0017, 0.0010, and 0.0005 for Parrot. Lenna, Baboon, and Parrot had PSNR values of 68.3211, 71.4562, and 74.6741, respectively; 68.4654, 71.9154, and 76.3126; and 69.5645, 72.4221, and 77.5822. As we've seen, the parrot's quality was superior than others. The comparison of two steganographic algorithms using a 512 byte payload and a 512 x 512 sized frame is shown in Table III [18,19]. These results indicate that the proposed technique provides superior image quality, with low distortion and high imperceptibility, even for larger payloads. Table III compares the proposed method with recent steganographic techniques in terms of PSNR and MSE for different images and payload sizes. The proposed method consistently outperforms other techniques, achieving better image quality and lower distortion.

Table IV showcases the visual differences between cover and stego frames through histograms for various images, illustrating the effect of embedding on image intensity. The histograms for the 512 x 512 cover and stego images corre-

TABLE I
Provided Detecting Pixels from Cover Image

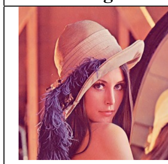
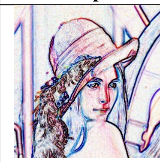
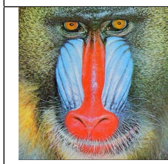
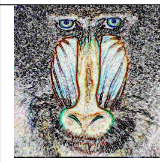
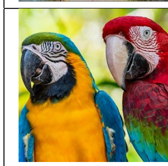
Image	Output	Pixel Selection Ratio
		27%
		58%
		30%

TABLE II
Quality measurement metrics of the projected method

Image	Dimension	Payload	PSNR	MSE	RMSE
Lenna	512X512	512 Bytes	68.3211	0.0027	0.0519
		256 Bytes	71.4562	0.0013	0.0363
		128 Bytes	74.6741	0.0007	0.0260
Babbon	512X512	512 Bytes	68.8945	0.0026	0.0512
		256 Bytes	71.9154	0.0013	0.0356
		128 Bytes	76.3126	0.0006	0.0252
Parrot	512X512	512 Bytes	69.9417	0.0017	0.0417
		256 Bytes	72.4221	0.0010	0.0299
		128 Bytes	77.5822	0.0005	0.0223

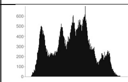
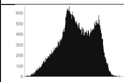
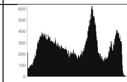
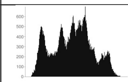
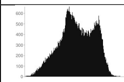
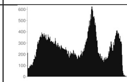
TABLE III
Comparison among recent steganographic techniques

Image	Model	Dimension	Payload	PSNR	MSE
Lenna	Model 1	512X512	512 Bytes	65.5456	0.0029
	Model 2	512X512	512 Bytes	67.5615	0.0028
	P-Model	512X512	512 Bytes	68.3211	0.0027
Babbon	Model 1	512X512	512 Bytes	65.9655	0.0028
	Model 2	512X512	512 Bytes	67.5315	0.0027
	P-Model	512X512	512 Bytes	68.8945	0.0026
Parrot	Model 1	512X512	512 Bytes	64.5154	0.0029
	Model 2	512X512	512 Bytes	67.3265	0.0027
	P-Model	512X512	512 Bytes	69.9417	0.0026

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.

sponding to the three photos are presented in Table 4.

TABLE IV
COMPARISON AMONG RECENT STEGANOGRAPHIC TECHNIQUES

Frames Type	**Lenna**	**Babbon**	**Parrot**
Cover			
Stego			

The minimal differences between cover and stego histograms underscore the effectiveness of the proposed method in maintaining the visual quality of the stego images. Consistent with a result of the histogram, the difference between two frames is inconsequential, these changes cannot be predicted with naked eyes.

V. CONCLUSION

In this research, we propose our safe data hiding technique for picture steganography using LSB and XOR with a Sobel chosen pixel selection strategy, which hides the secret data in the cover image using 128-bit AES encryption. The suggested steganography data concealing technique offers redundant security and reduced imperceptibility compared to various other data hiding methods, as demonstrated by the overhead discussion and reasonable result analysis. Future research might improve the suggested approach by incorporating machine learning techniques to improve resistance to sophisticated steganalysis assaults and optimize pixel selection. The security and adaptability of the system might further be increased by investigating the usage of other picture formats and dynamic key generation techniques.

REFERENCES

[1] Challita, Khalil, and Hikmat Farhat. "Combining steganography and cryptography: new directions." International Journal on New Computer Architectures and Their Applications (IJNCAA) 1.1 (2011): 199-208.

[2] Kumari, Pritam, Chetna Kumar, and Jaya Bhushan Preeyanshi. "Data security using image steganography and weighing its techniques." International Journal Of Scientific Technology Research 2.11 (2013): 238-241.

[3] Gupta Banik, Barnali, Manish Kumar Poddar, and Samir Kumar Bandyopadhyay. "Image steganography using edge detection by Kirsch operator and flexible replacement technique." Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2018, Volume 3. Springer Singapore, 2019.

[4] Banik, B. G., Poddar, M. K., & Bandyopadhyay, S. K. "Image Steganography Using Edge Detection by Kirsch Operator and Flexible Replacement Technique". In Emerging Technologies in Data Mining and Information Security (pp. 175-187). Springer, Singapore, 2019.

[5] Bhardwaj, M., Singh, L., & Saini, K. K. "An Efficient Approach to Information Hiding through Image Steganography using Edge Detection". In 2019 4th International Conference on Information Systems and Computer Networks (ISCON) (pp. 564-571). IEEE, November 2019.

[6] Wang, Y., Tang, M., & Wang, Z." High-capacity adaptive steganography based on LSB and Hamming code". Optik, 164685, 2020.

[7] Delmi, A., Suryadi, S., & Satria, Y. "Digital image steganography by using edge adaptive based chaos cryptography". In Journal of Physics: Conference Series (Vol. 1442, No. 1, p. 012041). IOP Publishing, January 2020.

[8] Prasad, S., & Pal, A. K." Stego-key-based image steganography scheme using edge detector and modulus function". International Journal of Computational Vision and Robotics, 10(3), 223-24, 2020.

[9] Ayub, N, & Selwal, A." An improved image steganography technique using edge-based data hiding in DCT domain". Journal of Interdisciplinary Mathematics, 23(2), 357-366, 2020

[10] Lakhwani, Kamlesh, and Kiran Kumari. "Kvl algorithm: Improved security psnr for hiding image in image using steganography." International Journal of Computational Engineering Research 3.10 (2014): 1-6.

[11] Krishna Nand Chaturvedi, Amit Doeger, "A Novel Approach for Data Hiding using LSB on Edges of a Gray Scale Cover Images",International Journal of Computer Applications (0975 – 8887) Volume 86 – No 7, January 2014, pp 36-40.

[12] Atallah M. Al-Shatnawi, "A New Method in Image Steganography with Improved Image Quality," Applied Mathematical Sciences, Vol. 6, 2012, no. 79, pp. 3907 – 3915.

[13] Abbas Cheddad, Joan Condell, Kevin Curran and Paul Mc Kevitt, "Enhancing Steganography In Digital Images," Canadian Conference on Computer and Robot Vision, IEEE 2008, pp: 326-322.

[14] J. K. Mandal and Debashis Das, "Colour Image Steganography Based on Pixel Value Differencing in Spatial Domain," International Journal of Information Sciences and Techniques (IJIST) Vol.2, No.4, July 2012, pp 83-93.

[15] Anil Kumar, Rohini Sharma, "A Secure Image Steganography Based on RSA Algorithm and Hash-LSB Technique," International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 7, July 2013. pp. 363-372.

[16] Smita P. Bansod Vanita M. Mane Leena R. Ragha, "Modified BPCS steganography using Hybrid Cryptography for Improving Data embedding Capacity," International Conference Multimedia Medical Records and Their Associations", IEEE 978-1-4244-3298, 1 sept 2009.

[17] Bourbakis, N., et al. "A synthetic stegano-crypto scheme for securing multimedia medical records and their associations." 2009 16th international conference on digital signal processing. IEEE, 2009.

[18] Alam, Sheikh Thanbir, Nusrat Jahan, and Md Maruf Hassan. "A New 8-Directional Pixel Selection Technique of LSB Based Image Steganography." Cyber Security and Computer Science: Second EAI International Conference, ICONCS 2020, Dhaka, Bangladesh, February 15-16, 2020, Proceedings 2. Springer International Publishing, 2020.

[19] Bhuiyan, Touhid, et al. "An image steganography algorithm using LSB replacement through XOR substitution." 2019 International Conference on Information and Communications Technology (ICOIACT). IEEE, 2019.

[20] Ansari, A.S., Mohammadi, M.S. and Parvez, M.T., A comparative study of recent steganography techniques for multiple image formats. International Journal of Computer Network and Information Security, 11(1), pp.11-25, 2019.

[21] K. Patel, "Performance analysis of aes, des and blowfish cryptographic algorithms on small and large data files," International Journal of Information Technology 11(4), 813–819 (2019).

[22] Alam, S. T., Hassan, M. M., Ahmad, R. B., Yaakob, N., Mou, M. T., Ong, B. L., & Kahar, N. F. "A Modified Lsb Image Steganography Method Using Msb-Based Pixel Filtering Algorithm." Journal of Advanced Research in Applied Sciences and Engineering Technology, 32-48, Oct. 2024, doi: https://10.37934/araset.62.1.3248

Authorized licensed use limited to: Southeast University. Downloaded on May 09,2026 at 10:10:22 UTC from IEEE Xplore. Restrictions apply.