Mesopotamian journal of Cybersecurity
Vol.5, No.3, **pp**. 913–926
DOI: https://doi.org/10.58496/MJCS/2025/052; ISSN: 2958-6542
https://mesopotamian.press/journals/index.php/cybersecurity

Research Article

Multilevel Text Protection System Using AES and DWT-DCT-SVD Techniques

Nuha Mohammed Khassaf [1],*, , Nada Hussein M. Ali [2] ,

[1] *Informatics Institute for Postgraduate Studies, Information Technology & Communications University, Iraq*

[2] *Department of Computer Science, College of science, University of Baghdad, Iraq*

ARTICLEINFO

Article History
Received 14 Mar 2025
Revised 19 May 2025
Accepted 23 Jul 2025
Published 6 Sep 2025

Keywords
AES-GCM
DWT
DCT
SVD
Zigzag

ABSTRACT

In the digital age, protecting intellectual property and sensitive information against unauthorized access is of paramount importance. While encryption helps keep data private and steganography hides the fact that data are present, using both together makes the security much stronger. This paper introduces a new way to hide encrypted text inside color images by integrating discrete wavelet transform (DWT), discrete cosine transform (DCT), and singular value decomposition (SVD), along with AES-GCM encryption, to guarantee data integrity and authenticity. The proposed method operates in the YCbCr color space, targeting the luminance (Y) channel to preserve perceptual quality. Embedding is performed within the HL subband obtained from DWT decomposition via SVD coefficients extracted from DCT-transformed images in the midfrequency band. A content-aware strategy combining Gaussian blurring, Canny edge detection, and zigzag scanning is employed to increase robustness against image processing attacks. The experimental results demonstrate the effectiveness of the proposed approach, which achieves up to a 10.4% improvement in PSNR, an SSIM score of 0.996, and a 0.10% increase in NCC over those of previous methods, which mostly rely on grayscale images. These results reflect the ability of the system to maintain high visual quality while offering strong resilience and security for embedded data in full-color images.

1. INTRODUCTION

In the rapidly advancing digital era, safeguarding intellectual property and sensitive information against cyber threats has emerged as a critical challenge for researchers and developers [1]. Digital data such as remote sensing images are widely used in applications ranging from flood analysis and tsunami modelling to military surveillance and infrastructure monitoring [2]. These images often include sensitive information that describes them, known as image captions. This information can reveal location-based images or contain secret information [3]. Because of this, the data are at risk of being stolen, copied without permission, or altered by people with bad intentions [4]. To address these risks, two main methods are used: encryption, which keeps the textual data secret by making it unreadable without the right permission [4], and steganography, which hides the textual data within the image in a way that is not noticeable [5]. Although both methods work well, several challenges persist, especially when they are used with high-quality color images:

- Difficulty in selecting embedding regions that remain robust under attacks such as noise, cropping, and geometric transformations [6].
- The visual quality of the image after embedding is preserved, avoiding perceptible distortion [5].
- The performance of many previous methods that focus primarily on grayscale images is limited, which restricts their practicality in real-world applications [6].

This study addresses the need for a hybrid framework capable of securely embedding ciphertexts into color images, achieving a balance between visual imperceptibility and resilience against attacks. The literature lacks comprehensive systems that integrate transformation-based processing, perceptual modelling, and encryption simultaneously, particularly those optimized for full-color images.

To achieve this, we suggest a mixed system that uses discrete wavelet transform (DWT), discrete cosine transform (DCT), and singular value decomposition (SVD) [7] to make the hidden message more reliable. In addition, we use AES-GCM encryption to protect and verify the hidden text [8]. The system works in the YCbCr color format, and it uses the brightness part (Y) to hide information on the basis of the principles of human visual perception [6]. Furthermore, Canny edge detection, Gaussian filtering, and zigzag scanning are employed to identify and prioritize regions for embedding, enhancing robustness against various image processing attacks [9] [10].

*Corresponding author. Email: phd202230705@iips.edu.iq

The main contributions of this study are as follows:

- A novel hybrid steganography framework that effectively combines DWT, DCT, and SVD to embed encrypted text within color images is proposed.
- Enhancement of data confidentiality by incorporating the AES-GCM algorithm ensures both encryption and authentication.
- The embedding is optimized by targeting the luminance (Y) channel in the YCbCr color space, preserving perceptual quality.
- A content-well embedding strategy that leverages Canny edge detection, Gaussian blur, and zigzag scanning is presented to improve robustness.
- A superior trade-off between imperceptibility and robustness, validated by experiments that outperform existing methods in both quality and resilience, is introduced.

2. RELATED WORK

Previous studies have focused on developing embedding techniques with the goal of improving robustness, embedding quality, and security. Several approaches, such as DWT, DCT, LSB, and advanced cryptography, have sought to combine multiple algorithms to achieve a balance between security and efficiency [6] [8]. However, despite significant progress in these studies, a number of challenges and limitations remain that affect the effectiveness of these systems in practical application environments.

Sana H. et al. (2021) proposed a system for encrypting a secret message via a chaotic stream cipher and then hiding the encrypted data in an RGB or grayscale wrapper image by modifying the least significant bits (k-LSB). The system achieved minimal distortion of the hidden image [11]. Prakash M. (2021) proposed data encryption via the RSA algorithm integrated with Hadoop, where the data are partitioned into attributes and stored separately for later retrieval of the secret image while increasing the execution speed [12]. Osama F. et al. (2021) used a combination of RSA, Huffman coding, and DWT, reducing the message size by 25% and achieving good image quality with a PSNR of over 40 dB and an SSIM close to 1 [13]. Mohamed H. et al. (2021) designed a system that is resistant to geometric attacks (shearing, rotation, and scaling) via DWT-DCT-SIFT, which provides significant stability in copyright protection applications [14]. Muhammed K. et al. (2022) proposed a technique that is based on the Chinese remainder theorem (CRT) algorithm and LSB and encrypts the text before it is embedded in the image, which achieves an accuracy of 99.5387 [15]. In the field of medical images, Fakhrytin H. et al. (2022) used DWT and SVD techniques to embed watermarks in high-resolution medical images, and the system demonstrated resistance to attacks [16]. Zear A. and Pradeep K. (2022) combined LWT-DCT-SVD with the MD5 algorithm to embed two watermarks in color images, enhancing security and integrity [17]. Bhargavi et al. (2022) presented a hybrid system based on dual biometric features to enhance security and robustness via DWT-DCT-SVD [18]. Bharathkumar et al. (2023) presented a hybrid algorithm that uses DCT, Arnold's cat map, and multilayer encryption (TPE, RSA), along with data compression via LZW and Huffman, with a PSNR of 93.620 [19]. Mustafa T. et al. (2023) used a one-time plate (OTP) algorithm with DWT and LSB, with secure key exchange, and experiments demonstrated the method's superiority over previous methods [20]. Sandra L. et al. (2023) relied on the Hermite transform with DCT and SVD to embed watermarks in grayscale images, with improved security and robustness [21]. Rahman et al. (2023) reviewed steganography techniques and critically analysed them, focusing on embedding methods across various domains (spatial, transformational, and adaptive) to enhance data security [22]. Xinyi L. (2024) presented three data hiding methods: modifications to JPEG tables, Huffman tables, and DCT coefficient corrections, with a focus on protecting digital content in critical applications [23]. Shahid R. et al. (2025) proposed a system that combines an LSB, a magic matrix, a multilevel encryption algorithm (MLEA), and a secret key, achieving good security results [24]. Ahmed et al. (2025) combined encryption with data masking techniques to increase the security of online information transmission by providing two layers of protection: privacy and anonymity. It highlights the challenges and weaknesses of current embedding techniques, highlighting robust and secure solutions [25].

Although previous studies have contributed to improving embedding techniques, they suffer from several limitations, the most prominent of which can be summarized as follows:

- Most of the current research has focused on standard images, such as grayscale or RGB images, and little attention has been given to remote sensing images or digital elevation models.
- Many studies use basic methods such as least significant bits (LSBs), which are easy to detect and reverse.
- There is a notable lack of research that uses detailed evaluation tools such as the Kullback–Leibler Divergence (KLD), Jensen–Shannon Divergence (JSD), and percentage residual difference (PRD), which are important for properly measuring image quality and how similar images are statistically.
- Modern encryption methods, such as AES-GCM, are not often used in watermarking or data hiding techniques. Additionally, very few studies have looked at hiding encrypted data, instead of visible data such as logos or pictures, inside cover images.

3. METHODOLOGY

This research aims to embed descriptive texts within digital images via advanced techniques (DWT, SVD, DCT) after encrypting them via the AES-GCM algorithm to ensure the security of the embedded information. The methodology includes three main phases: embedding, extraction, and evaluation after embedding and under different attacks.

3.1 Phase 1: Embedding encrypted text into images

In this phase, the encrypted text is hidden within the image via a combination of cryptographic and transform-domain techniques while aiming to preserve visual quality and enhance robustness against attacks. The overall workflow of this phase is shown in Figure 1 and detailed in Algorithm 1. The process involves the following key stages:

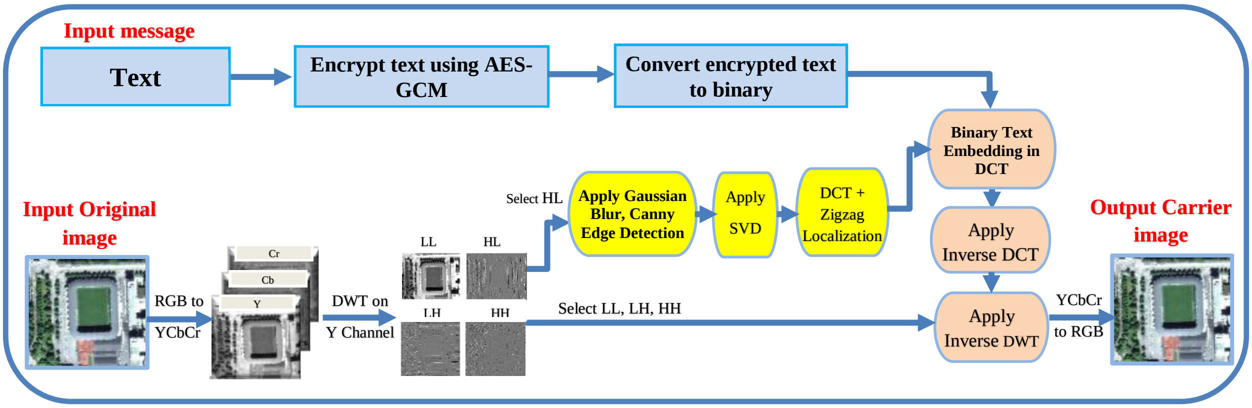


Fig. 1. Workflow of the proposed text embedding process

3.1.1 Text Encryption via the AES-GCM

In the first stage, the text is encrypted, which relies on the AES-GCM (advanced encryption standard - Galois/counter mode) algorithm, a symmetric encryption algorithm that combines encryption and reliability, providing both confidentiality and authentication. The encryption process is performed according to (1) [26]:

$$(C, T) = AES\_GCM\ (P, K, N) \tag{1}$$

where C denotes the ciphertext and T is the 16-byte authentication tag generated by the AES-GCM encryption algorithm to ensure the integrity and authenticity of the ciphertext. P represents the plaintext, K is a 128-bit secret key that is automatically generated, and N is a randomly generated nonce (number used once) that ensures the security and uniqueness of the encryption process.

3.1.2 Conversion from RGB to YCbCr

The RGB image type is converted to the YCbCr domain, where the cipher information is embedded in the Y channel. This process makes embedding less visible to the human eye and more resistant to image compression and noise. The color image is converted to the YCrCb color space via (2), (3), and (4) [27]:

$$Y = 0.299R + 0.587G + 0.114B \tag{2}$$

$$Cb = 128 - 0.168736R - 0.331264G + 0.5B \tag{3}$$

$$Cr = 128 + 0.5R - 0.418688G - 0.081312B \tag{4}$$

where R, G, and B refer to the red, green, and blue components of the RGB color. Y is the luminance channel, which represents brightness, Cb is the blue-difference chroma component, and Cr is the red-difference chroma component of the YCbCr color.

3.1.3 Applying DWT to the Y Channel

The DWT is applied to the Y channel to divide it into four submatrices [28]:

- LL: Low-frequency coefficients (representing the approximation of the original image).
- LH: Low-frequency coefficients horizontally and high-frequency coefficients vertically.
- HL: High-frequency coefficients horizontally and low-frequency coefficients vertically.

- HH: High-frequency coefficients in both directions.

Selecting the HL domain for text embedding provides a balance between masking and quality, as it contains horizontal image detail, making it less affected by noise than other domains, while maintaining good resistance to attack.

3.1.4 Gaussian Blur and Edge Detection Using Canny

Gaussian blur is applied to the HL subband via Gaussian filters to reduce noise and smooth fine details, which contributes to enhancing the stability of the image during both the embedding and extraction phases [9]. The Canny edge detection method is subsequently employed to identify edges by detecting regions with rapid pixel intensity changes. This helps in locating relatively stable areas within the image. These regions are considered ideal for embedding, as they are more resistant to distortions caused by compression or various attacks [29].

3.1.5 Singular Value Decomposition (SVD) and Normalization of Singular Values

The process of this stage can be summarized in the following steps:

- After applying the Canny algorithm to detect edges, SVD is applied to the resulting HL subband to extract its structural features and identify suitable regions for ciphertext embedding [30]. SVD of the edge-enhanced image is calculated via (5):

$$A = USV^T \tag{5}$$

where A is a matrix representing the resulting image after applying the Canny edge detection algorithm, U is an orthogonal matrix containing the eigenvectors of the productAA^T, S is a diagonal matrix containing the singular values that represent the energy and important information in the image, and V^T is the transpose of matrix V, which is an orthogonal matrix containing the eigenvectors of the product A^TA.

- Once the singular values in S are calculated, they are normalized such that they are within a specified range, which helps ensure that the embedding is consistent and does not cause noticeable visual distortions. The normalization formula is defined in (6)[31]:

$$S_{norm} = \frac{S - S_{min}}{S_{max} - S_{min}} \tag{6}$$

where S is the singular value matrix extracted from the SVD of the image block, S_{min} is the smallest singular value in matrix S, S_{max} is the largest singular value in S, and S_{norm} is the resulting normalized matrix where all singular values are scaled between 0 and 1.

- A singular value map is then generated on the basis of the normalized values to identify optimal regions for embedding. Regions with higher singular values are selected, as they tend to be more stable and resistant to degradation from compression or attacks [30].

Thus, the use of SVD enhances the robustness and reliability of the embedding process, improving the system's ability to recover the embedded text even after various image manipulations.

3.1.6 DCT and Embedding Binary Text

DCT is employed to convert the image into the frequency domain, facilitating information embedding with minimal impact on visual quality. The binary ciphertext is embedded into selected DCT coefficients, guided by the singular value map generated earlier, enhancing robustness against noise and common image attacks [32]:

- DCT converts the image to a matrix representing frequencies to embed information at medium and high frequencies without significantly affecting image quality.
- A zigzag scanning path is used within the DCT matrix to select medium-frequency coefficients, which offers a good trade-off between imperceptibility and robustness.
- After conversion to binary text, the encrypted text is embedded by modifying selected DCT coefficients according to the importance indicated by the SVD map. The magnitude of the modifications is scaled on the basis of the strength of the corresponding singular values to ensure stability under various distortions.
- After embedding, the inverse DCT (IDCT) is applied to reconstruct the HL subband, followed by the inverse DWT (IDWT) to integrate all subbands and restore the full image. Finally, the image is converted back from the YCbCr color space to the RGB format.

Algorithm (1): Embed Encrypted Text in Color Images

```
Input:
   - Color image I of size M × M
   - Plain text P
Output:
   - Stego image I′ with embedded encrypted text
Begin
//Step 1: Text encryption
Generate symmetric key K and random nonce N
Encrypt plaintext P via the AES-GCM to obtain ciphertext C
//Step 2: Color Space Conversion
Convert image I from the RGB to the YCrCb color space
Extracting luminance channel Y from YCrCb
//Step 3: DWT Transformation
Discrete wavelet transform (DWT) is applied to channel Y
Obtain subbands: LL, LH, HL, HH
Select the HL subband for embedding
//Step 4: Edge detection
Gaussian blur (kernel size 5×5) is applied to HL.
 Canny edge detection is performed to obtain an edge map
//Step 5: SVD decomposition
Singular value decomposition (SVD) is applied to the HL
Extract the singular values matrix S and normalize the values to [0, 1]
//Step 6: Apply DCT and perform Zigzag Scanning
Apply the discrete cosine transform (DCT) to HL
Zigzag scanning is performed on the DCT (HL) to determine the embedding positions
//Step 7: Embedding process
For each selected embedding position (i, j), do
   Compute step size: $step \leftarrow base\_step \times (1 + 0.5 \times S[i, j])$
   Modify the DCT coefficient at (i, j) using the corresponding bit of C
End for
Apply the inverse DCT (IDCT) to reconstruct the modified HL
Apply inverse DWT using LL, LH, modified HL, and HH to reconstruct channel Y
Merge the reconstructed Y with the original Cr and Cb channels
Convert the image back to the RGB color space
//Step 8: Output
Save the resulting stego image I′
End
```

3.2 Phase 2: Text extraction and retrieval

In this phase, reverse steps of the embedding process are applied to extract the hidden text, as illustrated in Figure 2, Algorithm 2, and summarized below:

- The stego image is first converted from the RGB to the YCrCb color space, and the luminance (Y) channel is extracted.
- The discrete wavelet transform (DWT) is applied to decompose the image and extract the HL subband.
- Gaussian blur is applied to the HL subband to reduce noise and enhance edge stability. Canny edge detection is then used to identify regions with strong structural features.
- Singular value decomposition (SVD) is applied to the HL subband to recreate the singular value map, which guides the identification of robust embedding locations.
- The discrete cosine transform (DCT) is applied to the HL subband to analyse the frequency components.
- A zigzag scanning path is then used to determine the frequency coefficients most suitable for extraction within the DCT.
- The embedded bits are extracted by computing the differences between the DCT coefficients of the original and stego images. Since text embedding relies on subtle modifications, referencing the original image ensures higher extraction

accuracy. The extracted values are interpreted on the basis of their alignment with the SVD-guided map, and the binary bitstream is reconstructed.

- Finally, the binary bitstream is decrypted via the AES-GCM decryption algorithm to recover the original plaintext.

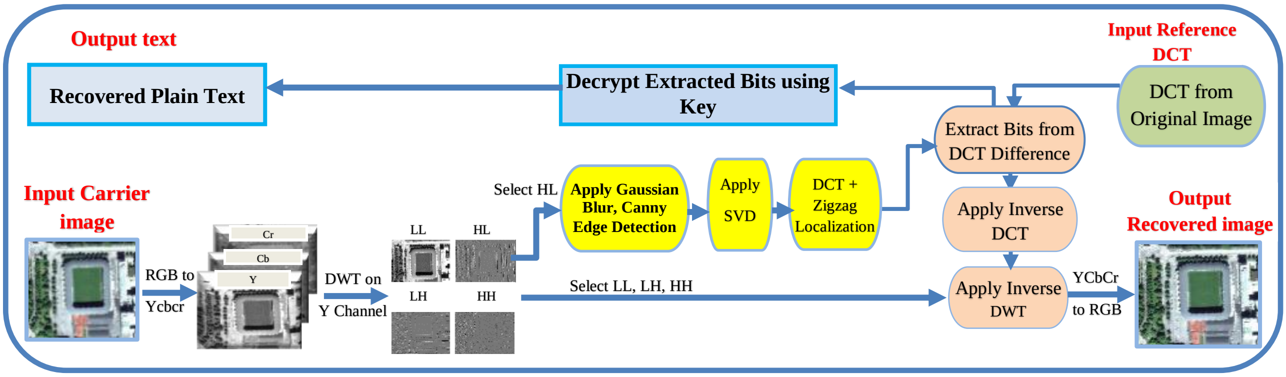


Fig. 2. Workflow of the proposed text extraction process

Algorithm (2): Extracting and Decrypting Text from Stego Images

```
Input:
    - Stego image I′
    - Original image I
    - Encryption key K
Output:
    - Recovered plaintext P′
Begin
 //Step 1: Color Space Conversion
Convert stego image I′ from the RGB space to the YCrCb color space
Extract luminance channel Y′ from YCrCb
//Step 2: DWT Transformation
Discrete wavelet transform (DWT) is applied to channel Y′
Obtain subbands: LL′, LH′, HL′, HH′
Select the HL′ subband for data extraction
//Step 3: Edge detection and SVD mapping
Gaussian blur (kernel size 5×5) is applied to HL′
 Canny edge detection is performed to locate strong features
Singular value decomposition (SVD) is applied to HL′
Extract the singular values matrix S and normalize the values to [0, 1]
//Step 4: Determine the embedding locations
Zigzag scanning is performed on HL′ to identify embedding positions
//Step 5: DCT Coefficient Analysis
Apply the discrete cosine transform (DCT) to HL′
For each embedding position (i, j), do
    Estimate the original embedding range via S and the base step
    Compare the current DCT coefficient at (i, j) with the estimated range
    Recover the corresponding embedded bit
End for
//Step 6: Bitstream Recovery
Reconstruct the complete encrypted payload from the recovered bitstream
Extract the nonce N from the beginning of the bitstream
Extract the ciphertext C and the authentication tag from the remaining bits
//Step 7: Decryption via the AES-GCM
The symmetric key K is used to decrypt ciphertext C
Obtain the recovered plaintext P′
End
```

3.3 Phase 3: Quality evaluation and resistance to attacks

Various assessments are applied to measure the strength of the proposed model after embedding and attacks.

3.3.1 Quality evaluation

- The peak signal-to-noise ratio (PSNR) is used to evaluate the quality of the resulting image after embedding or after attack. The PSNR is calculated via (7) [33]:

$$PSNR = 10 * log_{10}(\frac{255^2}{MSE}) \tag{7}$$

where the PSNR is a metric used to measure the quality of the image after embedding compared with the original image; it is expressed in decibels (dB). The mean square error (MSE) is the average of the squared differences between the pixel values of the original and processed images, which is calculated as shown in (8), and the constant 255 represents the maximum possible pixel value for an 8-bit image.

$$MSE = \frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}(I(i,j) - I'(i,j))^2 \tag{8}$$

where MSE is the mean squared difference between pixel values in two images, M and N represent the image dimensions (height and width), I (i, j) is the pixel value at location (i, j) of the original image and I´ (i, j) is the pixel value at the same location in the resulting image after embedding.

- The structural similarity index (SSIM) is used to measure the similarity between the original image and the resulting image after embedding or after attacks, as shown in (9) [34]. The SSIM relies on three components, namely, illumination, contrast, and structure (texture), making it more compatible with human visual perception than traditional metrics:

$$ssim = \frac{(2\mu_i\mu_j+C_1)(2\sigma_{ij}+C_2)}{(\mu_i^2+\mu_j{}^2+C_1)(\sigma_i^2+\sigma_j^2+C_2)} \tag{9}$$

where SSIM is an indicator that measures the structural similarity between two images, μ_i, μ_j are the arithmetic means of the pixel values in image i and image j, respectively, and σ_i^2 and σ_j^2 are the variances of image i and image j, respectively. σ_{ij} is the covariance between image i and image j. C_1 and C_2 are small constants added to stabilize the division when the denominators are close to zero.

- The normalized correlation coefficient (NCC) is used to measure the linear similarity between the original image and the image after embedding or attack, reflecting how closely the numerical values of the pixels match[33]:

$$NC = \frac{\sum_i\sum_j(g(i,j)*g'(i,j))}{\sqrt{(\sum_i\sum_j g^2(i,j))*(\sum_i\sum_j g'^2(i,j))}} \tag{10}$$

where NCC is the normalized correlation coefficient between the original image and the image after embedding, where a value close to 1 indicates high matching. g(i,j) is the pixel value at position (i,j) in the original image, and g′(i,j) is the pixel value at position (i,j) in the image after embedding.

- The percentage residual difference (PRD) is used to measure the relative difference between images, where a lower PRD value indicates greater similarity between images. It is used to evaluate quality of steganography methods [35]:

$$PRD = \sqrt{\frac{\sum_{i=1}^{N}(x_i - y_i)^2}{\sum_{i=1}^{N}x_i{}^2}} \times 100 \tag{11}$$

where x_i is the value of the i-th pixel in the original image, y_i is the value of the i-th pixel in the embedding image, and N is the total number of pixels in the image.

- Kullback–Leibler divergence (KLD) is used to measure the differences between the distributions of colors between two images, as shown in (12)[36]:

$$\mathrm{KLD}(\mathrm{P} \parallel \mathrm{Q}) = \sum_{\mathrm{x}\in\mathrm{X}} \mathrm{P}(\mathrm{x})\,\mathrm{Ln}(\mathrm{P}(\mathrm{x})/\mathrm{Q}(\mathrm{x})) \tag{12}$$

where P(x) is the probability distribution of the original image, Q(x) is the probability distribution of the embedded image, and X is the set of all possible pixel values (usually 0--255 in color images). A low KLD value indicates that the embedded image has preserved the properties of the original distribution well.

- Jensen–Shannon Divergence (JSD) is used to evaluate the quality of ciphertext embedding in color images. It is more comprehensive and less sensitive to subtle changes than KLD and more accurately reflects the difference in pixel distributions [36].

$$JSD(P \parallel Q) = \frac{1}{2}KLD\left(P \parallel \frac{P+Q}{2}\right) + \frac{1}{2}KLD\left(Q \parallel \frac{P+Q}{2}\right) \tag{13}$$

where P is the probability distribution of pixel values in the original image, Q is the probability distribution of pixel values in the embedded image, and $\frac{\boldsymbol{P+Q}}{\boldsymbol{2}}$ is the median distribution between P and Q.The importance of equation (13) lies in measuring the significant difference between the original and embedded images. JSD is more stable and balanced than KLD and is used to evaluate how well an image maintains the normal distribution of pixels, which enhances both steganography and robustness [36].

3.3.2 Types of attacks

The proposed system was evaluated under a variety of common image processing attacks [37, 38], including the following:

- Gaussian noise: Gaussian noise introduces random variations in pixel intensity, which can distort frequency components.
- Salt-and-pepper: This method randomly alters pixels to black or white, affecting the structural integrity of the image.
- Uniform Noise: This method applies consistent intensity noise, unpredictably altering pixel values.
- Cropping: Remove portions of the image by reducing its dimensions, potentially discarding embedded data.
- JPEG Compression: This method applies lossy compression that can degrade embedded textual data due to quantization artifacts.
- Scaling: Resizes the image, potentially modifying or removing embedded features.
- Histogram Equalization: This method enhances image contrast but may introduce distortions in frequency or spatial details, impacting hidden data.

The robustness of the proposed method stems from the intelligent embedding of encrypted textual data in carefully selected frequency components and edge-stable regions. This strategic placement improves resilience against attacks while maintaining high visual quality. Moreover, the integration of AES-GCM encryption prior to embedding ensures confidentiality and integrity, making the method well suited for secure digital media.

3.4 Dataset Used

In this study, the RSICD (Remote Sensing Image Caption Dataset), one of the largest and most popular datasets currently available for characterizing remote sensing imagery, was used. This dataset contains 10921 high-quality images collected from multiple sources, such as Google Earth and Baidu Map[39], and covers 30 diverse categories, including airports, ports, forests, residential areas, and agricultural areas. Each image in the RSICD dataset is accompanied by descriptive captions that provide detailed descriptions of the image content from a visual perspective. These captions are stored in a CSV file [40]. The original dimensions of the images are 224×224 pixels [39], but the images were resized to meet the requirements of the embedding process in this research. This dataset was used to experiment with techniques for embedding text within images by hiding the reference captions for each image within them via the proposed method. The diverse content of the dataset also helps evaluate the effectiveness of the proposed method in multiple real-world scenarios.

4. RESULTS ANALYSIS AND DISCUSSION

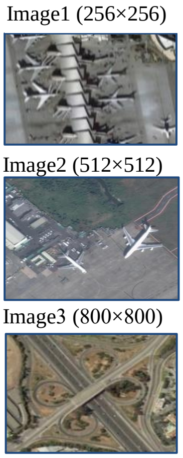
The proposed hybrid embedding method combines DWT, DCT, and SVD techniques and was evaluated via three remote sensing images of varying dimensions: Image 1 (256×256 pixels), Image 2 (512×512 pixels), and Image 3 (800×800 pixels). The descriptive text associated with each image was first encrypted via the AES-GCM algorithm to ensure the confidentiality and integrity of the embedded information, as illustrated in Figure 3, for example, image 1: "Plain text: most planes are docked at gates while one plane is on runway and ready to take-off", and the resulting ciphertext is composed of three critical components: the nonce, the cipher-text, and the authentication tag, which together guarantee both security and data authenticity where the encrypted message used in the embedding process is as follows:

{"nonce": "aba07 cc969816087b9ced8f5db4ad962",

"ciphertext":"1263d407025e7e256ff03ed988b42dc4e42 cc5cabe2245585ee18755bd61cf76620c9d9bcde240728ea2070f085ee688858ed525de86882a9e061229f81ebbe600fffa869 cc5cb268a654e2e5be08ad717d06c7a4b",

"tag": "4bdad217ca73b5dfc0fce4d4ce385df4" }.

The effectiveness of the proposed method was assessed via several performance metrics, such as the PSNR, SSIM, NCC, PRD, KLD, and JSD. The results from the evaluation are shown in Tables 1, 2, and 3, which provide insights into the method’s robustness, imperceptibility, and resistance to distortion.



Plain text: most of planes are docked at gates while one plane is on runway and ready to take-off

Embedding encryption text: {"nonce": "aba07 cc969816087b9ced8f5db4ad962", "cipher-text":
"1263d407025e7e256ff03ed988b42dc4e42
cc5cabe2245585ee18755bd61cf76620c9d9bcde240728ea2070f085ee688858ed525de86882a9e061229f81eb
be600fffa869 cc5cb268a654e2e5be08ad717d06c7a4b", "tag": "4bdad217ca73b5dfc0fce4d4ce385df4"}

Plain text: two planes stopped at intersection one of which was filled with containers on one side

Embedding encryption text: {"nonce": "9bd9a6acd14bfaad7b2c975db2475dc1", "cipher-text":
"174478ae11d4f195c4b3e0eef88e151f6bdaff8ed018f6d92719e1ab80af7f88477056d52eebaac4c01f9f50ada97
22 cc7dd1cf58d638b341eb7 cccfe1ebba3737a19a6c01db905bf31d37b9f1fbe02ef52f5458b49b", "tag":
"684b7ddb5a6ffdb39abec2f036a35278"}

Plain text: a diamond shaped viaduct combined with four ring shaped ramps looking like petals

Embedding encryption text: {"nonce": "92f2095f02013c2809b9ece2a0c4e197", "cipher-text":
"a5018992a22c61583dac17ec083d4fdcdf1b6119cd7e5a5678fdfc7aa0ce519fc1c12c9ce65ef6a1f35fe06575936
9ff644e1f27813e3b24a17a8ef829c081f8a8ff51aabddd6f63c27af743c61068717d", "tag":
"f383e96556e4192c0ada86c9b8aea496"}

Fig. 3. Three remote sensing images of different sizes with the corresponding plaintext and ciphertext used in the proposed method

TABLE I. EVALUATION RESULTS AFTER ATTACKS ON IMAGE 1 (256×256)

Attack	Extracted Cipher-text	Base Step	PSNR	SSIM	NCC	PRD	KLD	JSD	Recovery Time(s)
No attack	✓	10	42.172	0.989	0.999	1.320	0.005	0.001	0.0430
Gaussian noise at 0.01	✓	10	38.679	0.971	0.998	1.891	0.007	0.002	0.0610
Salt & pepper at 0.001	✓	30	30.66	0.901	0.989	2.980	0.013	0.004	0.0752
Uniform noise (high10,low-10)	✓	25	31.275	0.910	0.989	2.900	0.012	0.003	0.0680
Cropping 4%	✓	20	31.356	0.919	0.990	2.500	0.009	0.003	0.0700
JPG Compression (Quality=80)	✓	35	30.406	0.909	0.989	3.010	0.014	0.004	0.0640
Scale factor 0.98	✓	35	30.211	0.908	0.989	2.950	0.012	0.003	0.0533
Histogram equalization	✓	10	29.104	0.972	0.992	2.998	0.013	0.004	0.0658

TABLE II. EVALUATION RESULTS AFTER ATTACKS ON IMAGE 2 (512×512)

Attack	Extracted cipher-text	Base step	PSNR	SSIM	NCC	PRD	KLD	JSD	Recovery Time(s)
No attack	✓	10	44.655	0.996	0.999	1.071	0.004	0.001	0.1376
Gaussian noise at 0.01	✓	10	39.917	0.936	0.996	1.450	0006	0.002	0.1480
Salt & pepper at 0.001	✓	25	33.277	0.914	0.984	2.011	0.011	0.003	0.1520
Uniform noise(high10,low-10)	✓	15	34.863	0.920	0.987	2.009	0.0010	0.003	0.1450
Cropping 8%	✓	20	37.340	0.934	0.995	1.800	0.008	0,002	0.1500
JPG Compression (Quality=75)	✓	35	35.831	0.924	0.991	2.007	0.009	0.003	0.1428
Scale Factor 0.8	✓	35	38.285	0.954	0.995	1.500	0.007	0.002	0.1485
Histogram equalization	✓	10	32.975	0.977	0.993	2.010	0.012	0.003	0.1474

TABLE III. EVALUATION RESULTS AFTER ATTACKS ON IMAGE 3 (800×800)

Attack	Extracted cipher-text	Base step	PSNR	SSIM	NCC	PRD	KLD	JSD	Recovery Time(s)
No attack	✓	10	45.577	0.997	0.999	1.010	0.001	0.0002	0.3429
Gaussian noise at 0.01	✓	10	41.969	0.991	0.997	1.200	0.002	0.0003	0.3590
Salt & pepper at 0.001	✓	20	34.827	0.935	0.989	2.010	0.007	0.0010	0.3679
Uniform noise(high10,low-10)	✓	10	35.963	0.940	0.990	1.900	0.005	0.0007	0.3580
Cropping 8%	✓	20	38.541	0.944	0.996	1.411	0.003	0.0004	0.3625
JPG Compression (Quality=75)	✓	30	37.931	0.954	0.993	1.620	0.003	0.0004	0.3559
Scale Factor 0.8	✓	30	40.585	0.964	0.997	1.330	0.002	0.0003	0.3610
Histogram equalization	✓	10	34.978	0.9831	0.994	1.995	0.006	0.0009	0.3635

4.1 Imperceptibility analysis

Imperceptibility means the degree to which the stego image remains visually indistinguishable from the original image. This is measured via metrics such as the PSNR and SSIM. The results from Tables 1, 2, and 3 show that the proposed method works well in terms of imperceptibility. The PSNR values are over 42 dB, and the SSIM values are close to 1. 0 for all image sizes, which means that the changes are almost not noticeable:

- **Effect of image size:** The results show that as the image size increases, the embedding changes become less noticeable. For example, Image 3 (800×800) had the best performance, with a PSNR of 45. 577 and an SSIM of 0. 997. This means that the method can handle larger images without making them look different.
- **Heatmap Analysis:** Figure 4 shows heatmaps of pixel-level modifications caused by the embedding process. Yellow areas indicate more changes. The heatmaps show that as the image becomes larger, the changes are less spread out. Image 3 had the smallest changes, whereas image 1 (256×256) had more changes. This proves that the method only makes small, local changes that remain visually imperceptible.

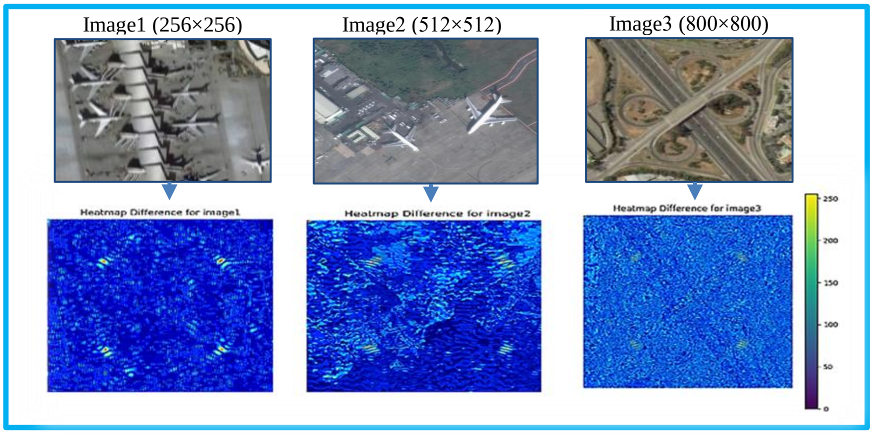


Fig. 4. Pixel-level heatmaps showing the visual differences among the three original images after their ciphertext is embedded

- **Histogram Analysis:** Figure 5 compares the histograms of the original and stego images for all three image sizes. The distributions are nearly identical, indicating that the statistical properties of the pixel intensities are preserved. This further confirms that the embedding process has a negligible effect on the overall appearance of the image.

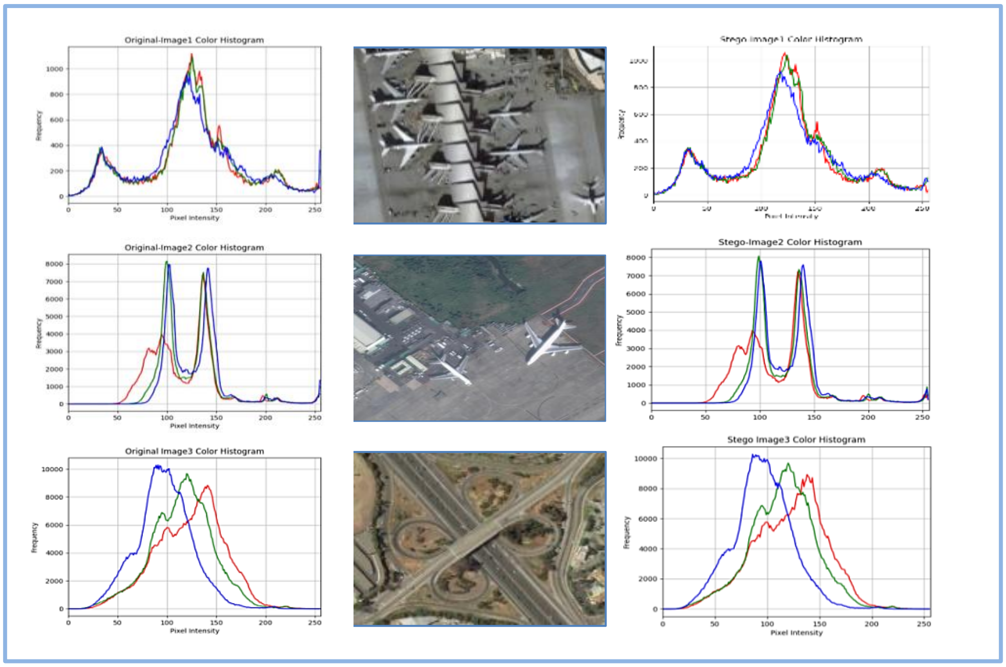


Fig. 5. Histograms of three original images and their corresponding images after embedding the ciphertext

4.2 Robustness Analysis

Robustness evaluates the system’s ability to successfully extract encrypted text after the stego image is subjected to various attacks. The performance of the proposed method was assessed via visual similarity metrics such as the PSNR, SSIM, and NCC and distortion metrics such as the PRD. As shown in Tables 1, 2, and 3, for three images with sizes of size

(256×256), (512×512), and (800×800), respectively, the encrypted ciphertext was correctly extracted in all the cases, regardless of the attack type, indicating the high robustness of the embedding method. While the NCC values (which measure the structural similarity between the original and stego images) remained above 0.984, even under severe distortions such as salt & pepper noise and JPEG compression, this proves that the visual content of the image was highly similar, with high text extraction quality.

The PRD values, which quantify image distortion, ranged from 1.01--3.01, with the highest levels observed under salt & pepper and JPEG attacks, especially in the image (256x256), as shown in Table 1. However, text recovery remained successful in all the scenarios, demonstrating the effectiveness of the embedding scheme in resisting visual degradation.

4.3 Statistical Similarity Analysis

Statistical divergence metrics such as Kullback–Leibler Divergence (KLD) and Jensen–Shannon Divergence (JSD) were used to assess the distribution similarity between the original and stego images. All values were very low (KLD ≤ 0.014, JSD ≤ 0.004) for the three images, even under strong attacks such as histogram equalization and compression. These results indicate that the statistical properties of the images are well preserved, making the system resilient against detection and tampering.

4.4 Capacity and Complexity Analysis

The embedding capacity increases with increasing image resolution, as larger images provide more space for hiding data. For example, Image 3 (800×800) accommodates longer encryption text while maintaining high visual quality, as indicated by the PSNR and SSIM values in Table 3. The system controls embedding strength via a parameter known as the base step, which determines how much the DCT coefficients are adjusted during text embedding. Choosing an appropriate base step is essential for balancing robustness and imperceptibility—larger values enhance resistance to strong attacks but may slightly affect visual quality.

4.5 Retrieval and Decryption Time Analysis

The time required to retrieve and decrypt the ciphertext was also recorded:

Image 1: Recovery time ranged from 0.0430 to 0.0752 seconds.

Image 2: Recovery time ranged from 0.1376 to 0.1520 seconds.

Image 3: Recovery time ranged from 0.3429 to 0.3679 seconds.

Despite attack severity, the retrieval times remained within acceptable bounds, highlighting the suitability of the system for semireal-time applications.

4.6 Comparison with Existing Methods

To evaluate the effectiveness of the proposed technique, it was compared with several existing embedding methods using key performance metrics, namely, PSNR, SSIM, and NCC, as summarized in Table 4 (for 512×512 images).

The proposed method outperformed most compared techniques, achieving the following:

- 13.10% higher PSNR than DWT-DCT-SVD [18],
- 10.78% higher PSNR than the RSA-Huffman coding-DWT [13],
- 12.71% higher PSNR than TPE-DCT [19].

It also showed a 5.39% improvement in SSIM over RSA-Huffman coding-DWT [13] while maintaining a high NCC value of 0.999, indicating strong structural similarity and robustness. Although the Hermite transform-SVD-DCT method [21] achieved a slightly higher SSIM (0.999 vs. 0.996), the proposed method achieved a better PSNR, making it more suitable for applications requiring both high visual quality and secure text embedding.

TABLE IV. PERFORMANCE COMPARISON WITH EXISTING METHODS IN THE LITERATURE (512×512 PIXEL IMAGES)

Technique	PSNR	SSIM	NCC
Proposed method	44.655	0.996	0.999

RSA-Huffman coding-DWT [13]	40.310	0.9451	–
DWT-DCT-SVD [18]	39.484	0.996	0.999
TPE-DCT [19]	39.620	–	–
DHWT-LSB [20]	43.861	–	–
Hermite Transform- SVD-DCT [21]	40.205	0.999	0.998

5. LIMINATIONS

Although the proposed method has demonstrated effectiveness in embedding and extracting encrypted text from images, several limitations should be considered for future enhancement:

- The system was tested via remote sensing images. To make the results more widely applicable and to check how well they work, future research could test them with other kinds of images, such as medical, biometric, or detailed images.
- The method uses traditional techniques such as DWT, DCT, and SVD for embedding. These methods work well in many situations, but they might not be as effective in very noisy conditions compared with newer methods that use deep learning for watermarking or hiding information.
- While the method balances how difficult it is to notice the changes and how well it can withstand attacks, there is room for improvement. Trying different frequency bands or mixing with adaptive or combined techniques could increase the resistance to changes such as noise or compression.
- To extract the hidden information, the system needs the original image's DCT data. This makes it less useful in situations where the original image is not available, such as in blind applications.

6. CONCLUSIONS AND FUTURE RECOMMENDATION

This study represents a significant step toward improving techniques for embedding secret text into remote sensing images via an integrated combination of traditional transformation techniques such as DWT, DCT, and SVD, which are supported by AES-GCM encryption to ensure confidentiality and integrity. The system was designed to achieve an optimal balance between image quality, security, time savings, and robustness against multiple attacks, such as noise, compression, and clipping.

The results showed a significant improvement in performance indicators, with the proposed model recording a 10.4% increase in PSNR and an SSIM value of 0.996, reflecting high image quality and close proximity to ideal values. The system also maintained high NCC values of 0.999, confirming the accuracy and integrity of the extracted text and the system's reliability in environments requiring high confidentiality. The proposed method's significance lies in its integration of various transformations and encryption techniques, as well as its use of YCbCr transformation, edge detection, blurring, and squiggle scanning techniques. This enhances the system's resistance to attacks and ensures the preservation of color image quality compared with previous studies, which were often limited to grayscale images.

As part of future directions, this research seeks to reduce reliance on the original image during text extraction by embedding DCT coefficients more deeply within the image, allowing for more independent and efficient information retrieval. The system can also be applied to high-resolution images and diverse environmental conditions to ensure its effectiveness in real-world scenarios. It can also be combined with artificial intelligence techniques, such as deep neural networks, to increase its resistance to attacks and expand its scope of application.

Conflicts of interest

The authors declare that they have no conflicts of interest.

Funding

There was no funding for this research study.

Acknowledgement

None

References

[1] M. E. Mahdi and N. H. M. Ali, “Copy move image forgery detection using multi-level local binary pattern algorithm,” Journal of Engineering, vol. 30, no. 6, pp. 141–157, 2024, doi: 10.31026/j.eng.2024.06.09.

[2] G. Sumbul, S. Nayak, and B. Demir, “SD-RSIC: Summarization-driven deep remote sensing image captioning,” IEEE Transactions on Geoscience and Remote Sensing, vol. 59, no. 8, pp. 6922–6934, 2020.

[3] N. M. Khassaf and N. H. M. Ali, “Improving pre-trained CNN-LSTM models for image captioning with hyper-parameter optimization,” Engineering, Technology & Applied Science Research, vol. 14, no. 5, pp. 17337–17343, 2024.

[4] J. H. Lim, C. S. Chan, K. W. Ng, L. Fan, and Q. Yang, “Protect, show, attend and tell: Empowering image captioning models with ownership protection,” Pattern Recognition, vol. 122, p. 108285, 2022.

[5] S. A. Shams, A. M. Abdulkareem, and A. A. Qasim, "Theoretical background of steganography," Mesopotamian Journal of CyberSecurity, vol. 1, no. 1, pp. 1–7, 2021.

[6] Y. Hua, X. Xi, C. Qu, J. Du, M. Weng, and B. Ye, “An adaptive watermarking for remote sensing images based on maximum entropy and discrete wavelet transformation,” KSII Transactions on Internet and Information Systems, vol. 18, no. 1, pp. 192–210, 2024.

[7] M. Begum and M. S. Uddin, “Digital image watermarking techniques: a review,” Information, vol. 11, no. 2, p. 110, 2020, doi: 10.3390/info11020110.

[8] A. Ali, S. Albitar, and F. Chio, "Digital steganography in smart healthcare protects patient privacy by hiding sensitive medical data within non sensitive files to prevent unauthorized people," Mesopotamian Journal of CyberSecurity, vol. 4, no. 2, pp. 20–62, 2024.

[9] T. K. Araghi and D. Megías, “Analysis and effectiveness of deeper levels of SVD on performance of hybrid DWT and SVD watermarking,” Multimedia Tools and Applications, vol. 83, no. 2, pp. 3895–3916, 2024.

[10] S. Solak, A. M. Abdirashid, A. Adjevi, and A. K. Sahu, “Robust data hiding method based on frequency coefficient variance in repetitive compression,” Engineering Science and Technology, an International Journal, vol. 56, p. 101756, 2024.

[11] S. Haimour, M. R. Al-Mousa, and R. R. Marie, “Using chaotic stream cipher to enhance data hiding in digital images,” arXiv preprint arXiv:2101.00897, 2021.

[12] P. Mohan, B. Kuppuraj, and S. Chellai, “An enhanced security measure for multimedia images using hadoop cluster,” International Journal of Operations Research and Information Systems (IJORIS), vol. 12, no. 3, pp. 1–7, 2021.

[13] O. F. A. Wahab, A. A. M. Khalaf, A. I. Hussein, and H. F. A. Hamed, “Hiding data using efficient combination of RSA cryptography, and compression steganography techniques,” IEEE Access, vol. 9, pp. 31805–31815, 2021.

[14] M. Hamidi, M. E. Haziti, H. Cherifi, and M. E. Hassouni, “A hybrid robust image watermarking method based on DWT-DCT and SIFT for copyright protection,” Journal of Imaging, vol. 7, no. 10, p. 218, 2021.

[15] M. A. A. Khader, S. Kavileswarapu, R. Sarkar, and S. Ganapathy, “Secure text in image steganography using pixel-based algorithm.”

[16] F. Horasan et al., “DWT-SVD based watermarking for high-resolution medical holographic images,” Complexity, vol. 2022, no. 1, p. 3154650, 2022.

[17] A. Zear and P. K. Singh, “Secure and robust color image dual watermarking based on LWT-DCT-SVD,” Multimedia Tools and Applications, vol. 81, no. 19, pp. 26721–26738, 2022.

[18] B. Mokashi et al., “Efficient hybrid blind watermarking in DWT-DCT-SVD with dual biometric features for images,” Contrast Media & Molecular Imaging, vol. 2022, no. 1, p. 2918126, 2022.

[19] B. Bharathkumar, S. Logeswar, S. Sudharsun, and B. Karthikeyan, “An enhanced triple prime encryption approach for image encryption in LSB steganography,” in Proc. 2023 First Int. Conf. on Advances in Electrical, Electronics and Computational Intelligence (ICAEECI), pp. 1–6, IEEE, 2023.

[20] M. Takaoğlu, A. Özyavaş, N. Ajlouni, and F. Takaoğlu, “Highly secured hybrid image steganography with an improved key generation and exchange for one-time-pad encryption method,” Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, vol. 23, no. 1, pp. 101–114, 2023.

[21] S. L. Gomez-Coronel, E. Moya-Albor, J. Brieva, and A. Romero-Arellano, "A Robust and Secure Watermarking Approach Based on Hermite Transform and SVD-DCT," Applied Sciences, vol. 13, no. 14, p. 8430, 2023.

[22] S. Rahman, J. Uddin, M. Zakarya, H. Hussain, A. A. Khan, A. Ahmed, and M. Haleem, "A Comprehensive Study of Digital Image Steganographic Techniques," IEEE Access, vol. 11, pp. 6770–6791, 2023.

[23] X. Li, "Advanced Reversible Data Hiding Techniques in JPEG Images: Methods, Applications, and Future Perspectives," Applied and Computational Engineering, vol. 94, pp. 77–85, 2024.

[24] S. Rahman, J. Uddin, H. Hussain, S. Shah, A. Salam, F. Amin, I. de la Torre Díez, D. L. R. Vargas, and J. C. M. Espinosa, "A novel and efficient digital image steganography technique using least significant bit substitution," Scientific Reports, vol. 15, no. 1, p. 107, 2025.

[25] H. A. Ahmed, A. Mahfoud, and O. I. Al-Sanjary, "Comprehensive Review of Cryptography and Steganography Algorithms," Journal of Information Systems Engineering and Management, vol. 10, no. 29s, pp. 211–228, 2025.

[26] M. J. Dworkin, "Recommendation for block cipher modes of operation: Galois/Counter Mode (GCM) and GMAC," 2007.

[27] R. Koju and S. R. Joshi, "Comparative analysis of color image watermarking technique in RGB, YUV, and YCbCr color channels," Nepal Journal of Science and Technology, vol. 15, no. 2, pp. 133–140, 2014.

[28] M. R. PourArian and A. Hanani, "Blind steganography in color images by double wavelet transform and improved Arnold transform," Indonesian Journal of Electrical Engineering and Computer Science, vol. 3, no. 3, pp. 586–600, 2016.

[29] R. Dhar, R. Gupta, and K. L. Baishnab, "An analysis of CANNY and LAPLACIAN of GAUSSIAN image filters in regard to evaluating retinal image," in 2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE), pp. 1–6, IEEE, 2014.

[30] Z. Zurinahni, J. S. Teh, M. Alawida, and A. Alabdulatif, "Hybrid SVD-based image watermarking schemes: a review," IEEE Access, vol. 9, pp. 32931–32968, 2021.

[31] J. Sharma, K. Kumar, P. Jain, R. H. C. Alfilh, and H. Alkattan, "Enhancing intrusion detection systems with adaptive neuro-fuzzy inference systems," Mesopotamian Journal of CyberSecurity, vol. 5, no. 1, pp. 1–10, 2025.

[32] B. Singh and M. K. Sharma, "Watermarking technique for document images using discrete curvelet transform and discrete cosine transform," Multimedia Tools Appl., pp. 1–25, 2024.

[33] A. Tareef, K. Al-Tarawneh, and A. Sleit, "Block-Based Watermarking for Robust Authentication and Integration of GIS Data," Eng. Technol. Appl. Sci. Res., vol. 14, no. 5, pp. 16340–16345, 2024.

[34] M. G. Martini, "Measuring objective image and video quality: On the relationship between SSIM and PSNR for DCT-based compressed images," IEEE Trans. Instrum. Meas., 2025.

[35] S. Boopathiraja, V. Punitha, P. Kalavathi, and V. B. Surya Prasath, "Computational 2D and 3D medical image data compression models," Arch. Comput. Methods Eng., vol. 29, no. 2, pp. 975–1007, 2022.

[36] W. Kim, "A random focusing method with Jensen–Shannon divergence for improving deep neural network performance ensuring architecture consistency," Neural Process. Lett., vol. 56, no. 4, p. 199, 2024.

[37] X. Cao, W. Zhang, Q. Zhou, C. Zhu, and N. Ren, "Template Watermarking Algorithm for Remote Sensing Images Based on Semantic Segmentation and Ellipse-Fitting," Remote Sens., vol. 17, no. 3, p. 502, 2025.

[38] S. S. Rajput, B. Mondal, and F. Q. Warsi, "A robust watermarking scheme via optimization-based image reconstruction technique," Multimedia Tools Appl., vol. 82, no. 16, pp. 25039–25060, 2023.

[39] R. Ramos and B. Martins, "Using neural encoder-decoder models with continuous outputs for remote sensing image captioning," IEEE Access, vol. 10, pp. 24852–24863, 2022.

[40] G. Sumbul, S. Nayak, and B. Demir, "SD-RSIC: Summarization-Driven Deep Remote Sensing Image Captioning," IEEE Trans. Geosci. Remote Sens., vol. 59, no. 8, pp. 6922–6934, 2021.