Journal of Information Hiding and Privacy Protection

Tech Science Press

Doi:10.32604/jihpp.2025.060898

REVIEW

Unveiling the Hidden Pixels: A Comprehensive Exploration of Digital Image Steganography Schemes

Nagaraj V. Dharwadkar*

Department of Computer Science, Central University of Karnataka, Kadaganchi, 585367, India

*Corresponding Author: Nagaraj V. Dharwadkar. Email: dharwadkarn@cuk.ac.in

Received: 12 November 2024; Accepted: 23 January 2025; Published: 27 March 2025

ABSTRACT: Steganography, the art of concealing information within innocuous mediums, has been practiced for centuries and continues to evolve with advances in digital technology. In the modern era, steganography has become an essential complementary tool to cryptography, offering an additional layer of security, stealth, and deniability in digital communications. With the rise of cyber threats such as hacking, malware, and phishing, it is crucial to adopt methods that protect the confidentiality and integrity of data. This review focuses specifically on text-in-image steganography, exploring a range of techniques, including Least Significant Bit (LSB), Pixel Value Differencing (PVD), and Transform Domain methods, to evaluate their effectiveness in real-world applications. The analysis covers key parameters such as embedding capacity, computational complexity, and the interplay with data compression and cryptographic techniques. While significant progress has been made in improving the security and quality of images in steganographic systems, challenges remain. For instance, higher payloads can lead to reduced Peak Signal-to-Noise Ratio (PSNR), compromising image quality. Despite these limitations, recent advancements show promising results in balancing security with minimal distortion. This paper provides valuable insights into the strengths and weaknesses of current techniques, highlighting future research directions that may enhance both the robustness and efficiency of steganographic methods in digital security.

KEYWORDS: Image steganography; cryptography; data compression; spatial domain; transform domain; dual approach

1 Introduction

Digital communication is the cornerstone of modern interaction. It is the process of exchanging messages, data, and information over digital channels between people, organizations, and gadgets. Digital communication modes, including email, instant messaging, video calls, and social media platforms, have revolutionized the way we communicate and collaborate. Digital communication supports real-time collaboration remote work and establishes relations across geographical boundaries. Digital communication has become essential to both personal and professional life, enabling innovation, productivity, and social interaction [1]. However, with the advantages of digital communication also come significant threats and concerns. Among these, the most important issues are those of security and privacy. The digital terrain is at risk from many possible forms of attacks, attacks that range from malicious hacking and data breaches to unauthorized surveillance and interception. As we increasingly depend on digital communication, so does the requirement of taking the necessary steps to save our information assets from such threats increase [2].

Numerous traditional techniques have been put into practice to lessen the risks related to digital communication. Some of them are shown in Fig. 1. Digital cryptography is another method of encryption of messages to ensure their confidentiality and integrity [3]. The use of cryptography in the protection of sensitive communications has been as old as the need for secure transactions of information; the mathematical algorithms and protocols ensure the unavailability of the original message (plaintext) to unauthorized parties [4]. In addition to ensuring confidentiality, cryptography guarantees the integrity and authenticity of data through symmetric and public key cryptographic techniques. A limitation of cryptography is the issue of key management. Steganography addresses this issue by concealing the presence of the message itself [5]. The information age has taken threats to information security to an extremely sophisticated and pervasive level. Cybercriminals illegally access confidential data by taking advantage of network, system, and software flaws. Malware compromises data integrity and confidentiality by infecting systems with viruses, worms, and ransomware, among others. Other major threats involve phishing attacks by hackers who tend to masquerade as authentic entities to extort sensitive information from users [6].

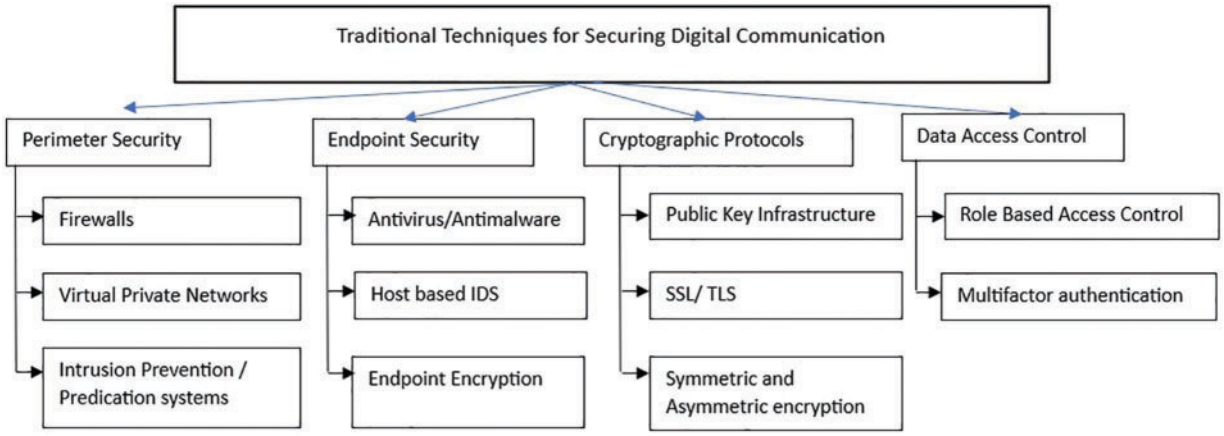

Figure 1: Traditional techniques to mitigate risks in digital communication

In light of the existing and emerging threats posed by cyber criminals, the necessity of higher information protection standards can be deemed inevitable. Companies and businesses operating in various industries are obliged to provide security measures and solutions to secure their intangible assets. Information security continues to remain relevant, not only as a way of protecting the content of information but also as one of the means of maintaining compliance with the requirements of legislation, as well as preserving customers' trust and ensuring the security of the organization's reputation [7].

Cryptography remains a core element of information security, employing encryption and decryption techniques to protect and analyse message traffic. It provides confidentiality, Integrity, and authenticity since it maps the sensitive information into an encrypted form termed cipher space which is intelligible only to the authorized parties. Nevertheless, based on the use of encryption, cryptography may actually alert the targeted party to the presence of encrypted communication and hence represents problems in other situations where secrecy and plausible denial are preferred. Furthermore, problems in the area of key management, and the analysis of cryptographic techniques also remain crucial concerns of cryptographic safety and security [3].

Steganography, derived from the Greek word for 'covered writing,' refers to the practice of embedding information within seemingly innocuous cover objects. Contrasted with cryptography, where messages are encrypted in order to obscure their understanding. Unlike cryptography, which focuses on obscuring

the content of the message, steganography emphasizes hiding the very presence of the message. There are various categories of steganography, including image, audio, video, and text. In any of the above-mentioned categories of steganography, a cover object is chosen as the carrier for the hidden message. Then using specific methods, specific secret data is inserted inside the cover object. On the receiver end, the secret information that was hidden is recovered from the cover object, completing the communication [8]. With the rising need for secure communication, steganography has gained much impetus as an addendum to traditional encryption techniques. While encryption scrambles the contents of a message, steganography embeds a hidden message such that its very presence is not suspected. There are several uses for this skill of sending information covertly, including copyright protection, digital forensics, cybersecurity, and covert communication [9]. Fig. 2 depicts some of the most common image steganography techniques. Audio and video steganography hide information in audio files or video frames, whereas text steganography hides information in text documents or messages. In all the categories, the purpose is to make the cover object look like the original to any naked eye observer but enable safe communication between the intended parties [10].

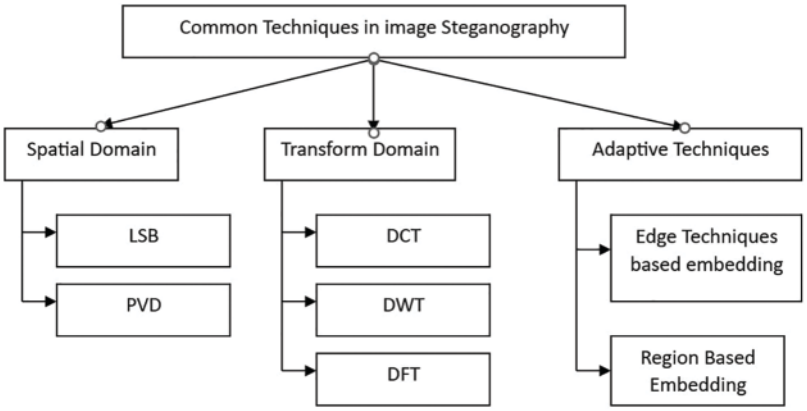

Figure 2: Common image steganography techniques

In image steganography, embedding algorithms conceal secret data within a cover image using techniques like Least Significant Bit (LSB) substitution, where the least significant bits of the image's pixels are altered. In the transform domain, techniques like the Discrete Cosine Transform Discrete Cosine Transform (DCT) embed information within the frequency components of the image. The extraction algorithms recover the hidden data by reading the modified bits or coefficients. The goal is to maximize the data capacity of the steganographic image while maintaining its imperceptibility. At the same time, the image must remain robust against processing operations without compromising these two aspects. In summary, these algorithms work together to securely hide and recover information from images, ensuring they remain undisturbed [11]. This flow is shown in Fig. 3 and Fig. 4.

The following are important terms necessary for a comprehensive understanding of steganography [12]:

1. Cover Image: The original image used for hiding the secret message.
2. Stego Image: The image with the secret message embedded in it.
3. Payload: The amount of secret message embedded within the cover image.
4. Embedding: The process of hiding data within the cover image.
5. Extraction: The process of retrieving the hidden data from the stego image.
6. Steganalysis: The detection and analysis of hidden messages within carrier media.
7. Cover Selection: The process of choosing an appropriate cover image for embedding the secret message.

8. Capacity: The maximum amount of information that can be embedded into a cover image without significantly altering its appearance.

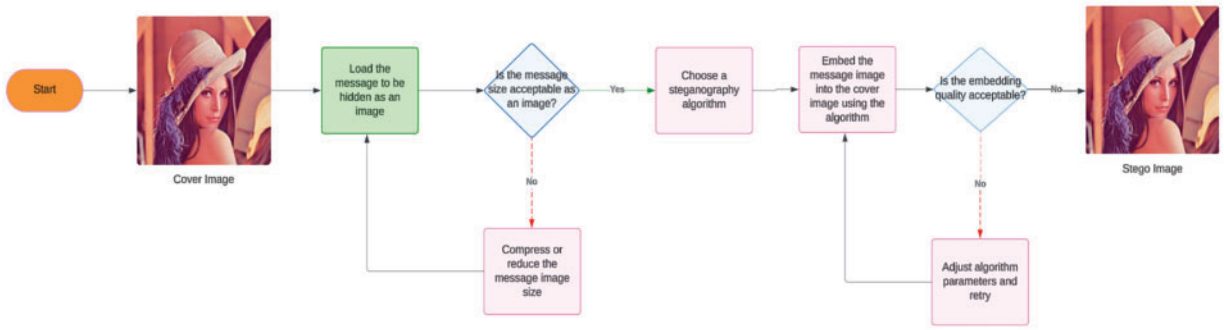

Figure 3: Flow of image steganography using embedding algorithm

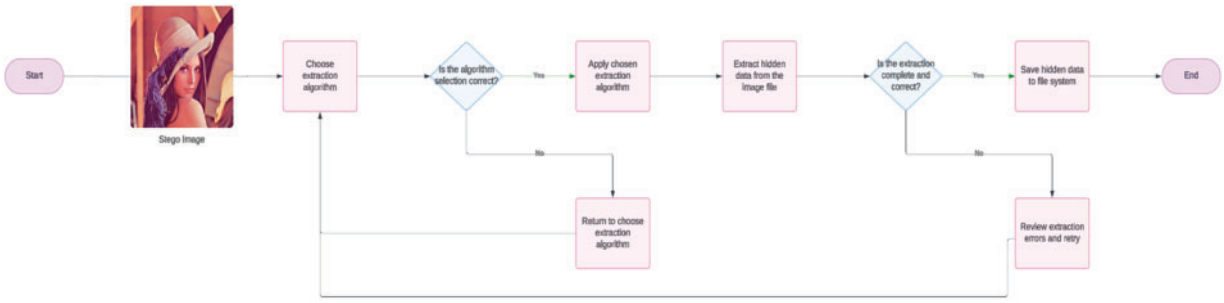

Figure 4: Flow of image steganography using extraction algorithm

Steganography is an effective complement to the limitations of traditional encryption in secure communication. Essentially, steganography differs from cryptography in that it hides the message rather than the contents of the message, since it inserts the message within digital data, which is often difficult to recognize. Steganography complements cryptography by concealing the message within other content, making it harder to detect.

The relationship between steganography, cryptography, and data compression is that they both work to protect information but go differently in methodology. Steganography conceals messages within other data, so the presence of such a message is hard to identify. Cryptography encodes messages for the safe transmission of data; data compression reduces data size for easy handling, as shown in Fig. 5. These techniques can be employed singularly or together to increase security. For example, a message may first be compressed to reduce its size, encrypted to ensure confidentiality, and then embedded into a cover image using steganography for covert transmission. This integration of compression and encryption with steganography may significantly enhance the effectiveness and versatility of secure communication [13–15]. With modern society advancing into a world where communication is increasingly shifting to the online environment, the combination of methods for protecting information has become even more relevant to cover new demands.

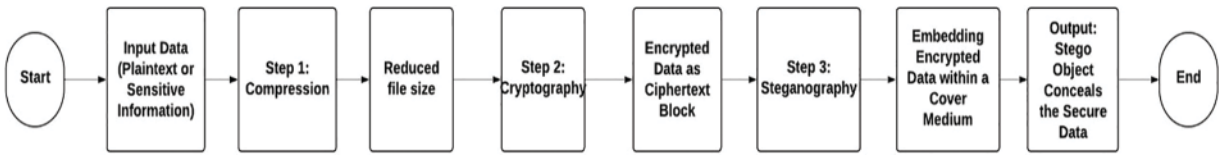

Figure 5: Flow of image steganography using extraction algorithm

Together, these techniques offer a comprehensive solution for protecting digital communication and personal information, ensuring confidentiality and trust even amidst evolving threats [16]. Due to the increasing landscape of cyber threats, a number of techniques related to information hiding have emerged as inherent modules within an information security paradigm. All these techniques aim to embed sensitive information within digital content in such a way that it remains undetected by unauthorized parties. One such technique is steganography, which embeds secret information inside cover media in ways imperceptible to unauthorized parties. Steganography embeds hidden information by altering pixels in images, samples in audio files, or frames in video streams in ways that are imperceptible to the human eye or ear [17].

This paper provides a comprehensive study of various steganography techniques, focusing on recent advancements in both spatial and transform domains. The aim of this study is to discuss different approaches and analyse their benefits and notable limitations. This will cover basic steganography methods as well as techniques that combine data compression and cryptography. The paper begins with an overview of current research in steganography. It then analyses various steganography techniques, evaluating their effectiveness and practical applications. The performance metrics section further assesses these techniques in terms of embedding capacity and computational complexity. Finally, the review concludes with the research findings and suggests directions for future research.

2 Spatial Domain Techniques

Spatial domain refers to the original representation of an image or signal in terms of its spatial coordinates; for example, an image is said to be represented as rows and columns of pixels, or samples in a signal. Operations that are performed directly in the spatial domain of an image. This includes basic operations like pixel value changes, filter application, or direct operations based on image or signal data in an arithmetic manner [18].

2.1 Least Significant Bit

The popular and understated original technique LSB gives birth to all the modifications and other techniques in image steganography; the alteration of LSB will only cause a slight change in color and, therefore, is mostly not visible to the human eye. Authors in [19] proposed the use of both LSB and Most Significant Bit (MSB) in color image steganography. The scheme uses a 24-bit color image with RGB values ranging from 0 to 255, where each pixel is represented by 3 bytes and one byte is used for embedding the LSB. Working is done in three steps: embedding and extraction using the secret key as the scheme, embedding it after the sinusoidal function is used to find the embedding locations, and finally, conditional embedding based on the size of the cover image. The use of both LSB and MSB embedding results in an increase in embedding capacity and robustness, but this increases the complexity of the method. Finally, the secret message is extracted using the same secret key and the LSB/MSB manipulation. Authors in [20] proposed the approach through bit rotation and inversion scoring to present a fresh approach to LSB extraction and embedding in grayscale images. This new approach based on bit rotation and inversion scoring these employ the classical method of image steganography by replacing the least significant bits of

grayscale pixels in the cover image. The size of the message is embedded in the LSBs of the first 16 grayscale pixels. The process of embedding involves grouping pixels of the cover image with calculated gaps and replacing their LSBs with bits of the secret message. During extraction, the size of the message is extracted first, and then the embedded message is retrieved. It takes into consideration the rotation and inversion indicators during extraction, enhancing the robustness against steganalysis. The method proposed by [21] is a bi-directional LSB steganography scheme that creates fixed-size blocks out of the cover image's LSB plane. Depending on how similar the cover and message bits are, the bits of each block are XOR'd with the matching message bits to determine whether the message will be encoded forward or backward. The performance evaluation demonstrates how effectively the information is hidden while preserving image quality, using metrics such as mean squared error and structural similarity indices in addition to its potential to minimize detectability by lowering the quantity of bits in the cover image that must be altered. However, this method may lack robustness in high-security scenarios, as its focus is more on minimizing the impact on the cover image rather than ensuring high security. In summary, spatial domain techniques for image steganography have been explored in various forms: some offer straightforward advantages in terms of ease of implementation but come with simple challenges related to detectability, while more advanced techniques such as combinations of LSB and MSB embedding, bit rotation and inversion scoring, and bi-directional LSB steganography offer better embedding capacity, robustness, and security, albeit at the cost of increased complexity and computational resources. This evolution highlights a trade-off between preserving image quality and enhancing the security of the embedded information an area of ongoing research.

2.2 Pixel Value Differencing

For I be the cover image represented as a matrix of pixel intensities, where $I_{i,j}$ represents the intensity value of the pixel at coordinates (i, j). The Pixel Value Differencing (PVD) algorithm calculates the difference between adjacent pixel intensities in the cover image. This can be represented as:

$$\Delta I(i, j) = |I(i, j) - I(i', j')| \tag{1}$$

where (i', j') represents the coordinates of a neighbouring pixel. The secret data is then embedded by the program by altering these pixel disparities. The modified pixel difference $\Delta I'(i, j)$ can be computed using a function:

$$\Delta I'(i, j) = f(\Delta I(i, j), secret\, data) \tag{2}$$

Finally, the stego image is generated by updating the pixel intensities based on the modified differences:

$$I'(i, j) = I(i, j) + \Delta I'(i, j) \tag{3}$$

where $I'(i, j)$ represents the intensity value of the pixel in the stego image. The selection of function f depends on the specific embedding approach and the characteristics of the secret data. PVD calculates the differences between adjacent pixel values, modifies these differences according to the secret data, and creates the stego image by adjusting pixel intensities [22].

In image steganography, the image is divided into multiple blocks, and pixel differences are altered to suit the need for embedding, as explained by [23]. The adaptive block-based PVD technique is proposed by [24]. In the presented PVD scheme, the image under consideration is transformed into non-overlapping blocks of size 3×3, and the difference matrix is computed based on the median of the pixel differences in each block. The algorithm adjusts the minimum and maximum difference range, placing the secret data in

regions of the image with high-intensity fluctuations. Confidentiality is enhanced by embedding data in areas with edges and intensity gradients, avoiding smooth or flat regions.

Texture images, which exhibit higher embedding capacity compared to normal images, are also considered in this method, further improving the algorithm. The proposed algorithm performs well, particularly at low embedding rates, yielding a higher Peak Signal-to-Noise Ratio (PSNR) compared to conventional PVD methods. Such enhanced PVD for color images is employed by [25] using a technique called Dynamic Pixel Value Differencing (DPVD), which not only embeds data in optimal directions but also in each channel. It splits the cover image into pixel blocks and computes horizontal, vertical, and diagonal PZ means per frame.

According to the PVD method, data bits are buried in each block depending on the maximum embedding limit for every channel direction. Extraction requires partitioning of the stego-image as in the embedding algorithm, computing the new differences, and obtaining the binary data of each color channel depending on the maximum embedding directionality. DPVD not only enhances embedding capacity but also increases the extraction difficulty by extending the embedding process across the color channels of the image.

As for the spatial domain, image steganography techniques present a wide variety of methods, each of which has unique advantages and drawbacks. Some techniques, such as LSB, are quite simple and can be implemented with low computational complexity; however, they are easily detected. The advanced techniques include basic PVD and MSB as well as incorporating S/N chaotic maps, which improve security and capacity and sometimes, but not necessarily, lead to higher complexity and computational costs. The development of steganography techniques is still actively focused on improving embedding efficiency, robustness, and resistance to steganalysis attacks.

2.3 Hybrid of LSB and PVD

The use of LSB in combination with PVD leverages the simplicity of LSB and the robustness of PVD. By integrating these techniques, the goal is to improve both the embedding capacity and the likelihood of avoiding detection. For instance, while single or dual LSB (Least Significant Bit) is a basic approach suggested by [26], there is the variant called Five Directional PVD with Modified LSB Substitution that divides the image into non-overlapping blocks. Each block is then transformed using k-bit Least Significant Bit substitution. This involves calculating the difference values and using binary bits from the secret data stream based on the range tables while optimizing pixel values to avoid boundary issues. In addition to the regular embedding process, extraction involves a reverse procedure, using additional directional computations to recover the embedded data. This technique has been enhanced by merging the advantages of PVD and LSB methods, improving its embedding accuracy and capacity. Splitting the image into a set of independent n × n blocks and using k-bit LSB substitution regulates pixel values within the block and avoids boundary problems.

Pradhan et al. [27] presented a study that combines Least Significant Bit (LSB) substitution, Pixel Value Differencing (PVD), and Exploiting Modification Direction (Earth Mover's Distance (EMD)) to optimize embedding capacity and imperceptibility. The authors utilized a hybrid approach where LSB enhanced simplicity, PVD improved payload capacity, and EMD ensured minimal distortion. The results showed a hiding capacity of up to 4 bpp and PSNR values exceeding 40 dB, indicating strong imperceptibility. However, Structural Similarity Index Measure (SSIM) and Quality Index (QI) were not discussed in detail, limiting the robustness assessment. While this methodology effectively balances capacity and imperceptibility, further exploration of its security against complex attacks is needed.

Authors in [28] introduced a new embedding technique in gray-scale images where pixel pairs are 1×2 blocks and compared their difference to determine how many hidden bits need to be included The quantization levels control the embedding process by categorizing pixel intensities into three ranges: low, medium, and high occurrences. In contrast, extraction is carried out by using the range table to determine the quantity of additional inactive bits and then extracting them in accordance with the LSBs. This method enables an elegant writing of the secret data and extraction without any other information; it equally provides a perfect hiding of text without causing much interference to the image quality. In this context, Authors in [29] proposed a steganography model utilizing a Duffing map to generate a random index vector (RIV) for data hiding. Chaotic sequences are employed to enhance embedding capacity without significantly affecting distortion. The RIV that excludes the true values along with the indices and they are not repeating within the image dimensions offers ultimate embedding. The process involves mapping the cover image and secret message to the image, generating chaotic random numbers, and acquiring exclusive indices before modifying the LSB of the concealed image as guided by the RIV. However, one must bear in mind that there is a collision probability associated with the indexes being generated, which means that the generated index can accidentally overwrite other data or even alter it.

Authors in [30] formulated steganography algorithm that uses PVD in combination with LSB on RGB covers to embed the secret data. As opposed to an ordered map which works through pixels in a sequential manner, hence the security is improved. There are two kinds of steps in preprocessing, namely RGB channel extraction and Integer Wavelet Transform (IWT). The process of embedding begins at the high-frequency sub bands and the process of postprocessing includes, reconstruction of image and inverse wavelet transform. Extraction remains similar to embedding, whereby it applies the same chaotic map alongside a secret key. It meets requirements for shallow hiding and does not compromise the quality of the image, but may be vulnerable to detection techniques designed for non-sequential hiding patterns. Authors in [31] combined LSB substitution with error correction coding, using Hamming codes to enhance robustness and ensure reliable data extraction in noisy conditions. The hybrid approach emphasizes maintaining visual imperceptibility, achieving a PSNR of over 35 dB and SSIM values above 0.90, demonstrating minimal perceptual degradation. While the integration of error correction strengthens resistance to noise, the method's payload capacity is limited compared to techniques like PVD or EMD. This makes it ideal for secure communication in environments prone to noise, though further optimization is required for applications demanding higher embedding capacities, particularly in hybrid spatial steganography systems.

Tables 1–3 summarize various spatial domain techniques, highlighting their respective advantages and limitations. The techniques include the least significant bit embedding, hybrid techniques, generation of random sequences, bi-directional techniques of coding, block-based techniques, and chaotic algorithms. Advantages include high security with an increase in the efficiency of embedding and minimal distortion of images. Limitations include computational complexity, vulnerability in attacks, and possible degradations of image quality. Each technique has unique features that enhance the embedding capacity or resistance to steganalysis, but at the cost of one or the other; thus, much care should be taken according to specific application requirements.

Table 1: Comparison of different steganography techniques, their benefits and limitations (part 1)

Paper	Specific technique employed	Benefits	Limitations
[19]	LSB and MSB based image steganography using color images (RGB)	1. Enhanced security through MSB utilization. 2. Retention of image quality with high PSNR and low MSE. 3. Message embedding in imperceptible areas. 4. Security augmented by pixel location obfuscation.	1. Potential decrease in PSNRPSNR with larger payloads. 2. Complexity compared to traditional LSB-based methods. 3. Limited by image format compatibility.
[20]	Hybrid LSB and MSB em- bedding technique	1. Increased security 2. Retention of image quality 3. Concealment of message location using sinusoidal curve points. 4. Consistent histogram distribution preserving visual appearance.	1. Reliance on a secret key for embedding and extraction, requiring secure key management. 2. Complexity in determining optimal embedding locations due to sinusoidal curve points. 3. Vulnerability to attacks targeting LSB and MSB manipulation. 4. Limited applicability to grayscale images.
[21]	Bidirectional coding	1. Minimization of amendments to cover image bits. 2. Reduction of error between original and stego-image. 3. Better structural similarity index compared to traditional LSB. 4. Adaptability to different block sizes for encoding.	1. Limited embedding capacity in LSB bit planes. 2. Dependency on predefined block size and parameters. 3. Inability to decode direction without additional information. 4. Applicability primarily to grayscale images.
[24]	Block based Pixel Value Differencing (PVD)	1. Improved embedding quality. 2. Texture masking effect. 3. Increased security.	1. Dependency on image characteristics. 2. Pivot pixel selection sensitivity. 3. Embedding capacity constraints.

Table 2: Comparison of different steganography techniques, their benefits and limitations (part 2)

Paper	Specific technique employed	Benefits	Limitations
[25]	Directional Pixel Value Differencing (DPVD)	1. Optimal embedding directions. 2. Channel-specific embedding. 3. Enhanced embedding capacity. 4. Robust extraction. 5. Improved performance.	1. Computational complexity. 2. Sensitivity to image content. 3. Detection vulnerability 4. Potential distortion.
[28]	Combining LSB (Least Significant Bit) substitution with PVD	1. High embedding capacity compared to existing methods. 2. Improved visual imperceptibility compared to some previous approaches. 3. Resistance against steganalysis detection attacks.	1. Error blocks in the embedding process can affect the recovery of secret data. 2. Visual imperceptibility may decrease as the number of embedded bits increases. 3. In the extraction process, there might be a need for additional information in some cases.
[29]	Chaotic steganography algorithm based on Duffing map	1. High hidden information capacity. 2. Data security with chaotic systems. 3. Minimal image distortion. 4. Additional security layer with chaotic map.	1. Computational complexity. 2. Sensitivity to parameter changes. 3. Limited data hiding capacity. 4. Dependency on chaotic map secrecy.
[30]	Combines (PVD) and (LSB) techniques for steganography in color images	1. Vulnerability to statistical attacks. 2. Possible visual image degradation with high embedding capacities. 3. Sensitivity to initial conditions and control parameters of the chaotic map.	1. Utilization of PVD and Libor embedding secret data into RGB cover images. 2. Nonsequential embedding process through the use of chaotic maps. 3. Increased security and randomness due to the use of complex chaotic maps.

(Continued)

Table 2 (continued)

Paper	Specific technique employed	Benefits	Limitations
[27]	Combines LSB substitution, PVD, and EMD for data embedding	1. High embedding capacity (up to 4 bits per pixel (bpp)). 2. Strong imperceptibility (PSNR > 40 dB). 3. Hybrid approach balances simplicity, capacity, and minimal distortion.	1. Limited discussion on SSIM and QI, impacting robustness assessment. 2. Lack of detailed exploration of security against complex attacks.

Table 3: Comparison of different steganography techniques, their benefits and limitations (part 3)

Paper	Specific technique employed	Benefits	Limitations
[27]	Combines LSB substitution, PVD, and EMD for data embedding	1. High embedding capacity (up to 4 bits per pixel (bpp)). 2. Strong imperceptibility (PSNR > 40 dB). 3. Hybrid approach balances simplicity, capacity, and minimal distortion.	1. Limited discussion on SSIM and QI, impacting robustness assessment. 2. Lack of detailed exploration of security against complex attacks.
[31]	LSB substitution combined with Hamming codes for error correction	1. Increased robustness against noise. 2. High visual imperceptibility (PSNR > 35 dB, SSIM > 0.90). 3. Reliable data extraction in noisy conditions.	1. Limited payload capacity compared to PVD or EMD. 2. Requires optimization for applications needing higher embedding capacity.

3 Transform Domain Techniques

The transform domain refers to an alternative representation of image data, obtained by applying mathematical transformations such as the Discrete Fourier Transform (DFT), DCT, Discrete Wavelet Transform (DWT), and others. Transforming the image into a different domain can reveal certain features or properties of the data that may be useful for embedding secret information or for enhancing certain aspects of the steganographic process, such as imperceptibility or robustness against attacks. It manipulates transformed coefficients instead of pixel values, offering new opportunities for data hiding and manipulation [32].

Most commonly used Technique DCT works as per follows:

Given a finite sequence of N data points x[n] for 0n < N, the DCT of order N is defined as:

$$X[k] = \sum_{n=0}^{N-1} x(n) \cos\cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right) for\ 0 \le k \le N \tag{4}$$

where $X[k]$ is the transformed coefficient at index k and $x[n]$ are the original signal. This formula represents the forward DCT, which transforms the signal from the spatial domain to the frequency domain.

The inverse DCT (IDCT), which transforms the signal back from the frequency domain to the spatial domain, is given by:

$$X[n] = \sum_{n=0}^{N-1} x(n) \cos\cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right) for\ 0 \le n \le N \tag{5}$$

Various normalization factors may be applied to the DCT formulation depending on the specific variant (e.g., Type-II DCT used in JPEG compression).

According to the study by [33], proper covering selection based on local variance allows for the imperceptibility of the watermark. Following this, the DCT to the YCbCr components is used to determine relevant scales. The message bits to be conveyed secretly are embedded under curvelet coefficients. A key is used to select the blocks. The extraction process involves converting the stego RGB images to YCbCr, computing the DWT of the Cb and Cr components, and extracting the hidden bits using the key. The process of embedding the data into the curvelet coefficients preserves a high level of security, as keeping it out of the reach of unauthorized users affects the neighborhood of the transformed shape of the coefficients. The accuracy of the algorithm depends on parameter settings such as α, β, and δ, which may need to be adjusted to suit different situations.

To improve the steganographic capacity of JPEG images, authors in [34] introduced an improved method that utilizes Block Entropy Transformation (BET) to enhance JPEG steganography by spatially embedding entropy in the DCT domain. It employs spatial distortion measures to encode the embedding entropy for each DCT coefficient that is used in JPEG images. The distortion measure for quantized DCT coefficients is derived using inverse mapping of optimum parameters for minimal distortion embedding. BET is employed for initial embedding, and Gaussian filtering is applied to smooth the distortion in the way described below. Additionally, this method combines both spatial and JPEG domains to create a more secure steganography technique, which minimizes distortion, as demonstrated in the experimental results.

To minimize embedding errors and produce stego-images that closely resemble normal images, undetectable to the naked eye, authors in [35] developed a technique that integrates DT-CWT with data embedding for steganography. Initially, the RGB cover image and the secret image are pre-processed. The secret image is transformed into its gradient and blurred versions through Gaussian smoothing. Next, DT-CWT is applied to the cover image, producing coefficient sub-bands that increase the embedding capacity. These sub-bands contain secret patches, which are embedded using intensity mapping. Finally, in step 6, the coefficients are inverted back into the spatial domain for transmission. During retrieval, the stego-image undergoes the DT-CWT process, after which the patches, embedded with exclusive keys, are extracted and transformed back into the secret image.

Authors in [36] suggested an approach to reduce distortion in the resulting stego-image to provide near-perfect representation of the cover image while holding the messages. This approach utilizes the DCT, One-Time Pad (OTP), and Pseudo-Random Noise Sequence (PN-Sequence) to securely embed and extract messages within grayscale images. In the encryption and embedding process, the cover image is divided into multiple blocks, from which the DC coefficients are extracted using the DCT. The second message, a secret binary image, is encrypted with OTP and embedded into the DC matrix using PN-Sequence binary numbers, ensuring that the message remains intact and accessible only to the authors. The decryption and extraction processes closely mirror the embedding process. The DCT and PN-Sequence are used to extract the hidden message, while OTP decryption is applied to decode the resulting secret image.

For JPEG image steganography, authors in [37] proposed a two-module approach. This approach involves encrypting the secret message, embedding it into the carrier signal, and decoding the same signal to retrieve the encrypted message. The encoding module involves dividing the image into smaller 8 × 8 blocks, applying the DCT, quantization, embedding the secret message using Singular Value Decomposition (SVD), and finally performing entropy encoding. The extraction module of the steganography system is responsible for retrieving the hidden message from the stego-image by reversing the above processes. This approach is based on JPEG compression, where the message is embedded during the distortion process, which helps in the transmission and storage of secret information, thereby enhancing the overall effectiveness of the process. The mathematical formulations of the forward DWT and its inverse (IDWT)

Forward DWT: Let I(x, y) be the pixel intensity of the image at coordinates (x, y). The forward DWT decomposes the image into approximation (low-frequency) coefficients LL and detail (high-frequency) coefficients LH, HL, and HH. The approximation coefficients LL are obtained by convolving the image with a low-pass filter h and downsampling by 2 in both dimensions:

$$LL(x,y) = \sum_m \sum_n h(m)\, h(n)\, I(2x-m, 2y-n) \tag{6}$$

The detail coefficients *LH*, *HL*, and *HH* are obtained similarly by convolving the image with high-pass filters g_1, g_2, and g_3 (or their flipped versions) and down sampling:

$$LH(x,y) = \sum_m \sum_n g_1(m)\, h(n)\, I(2x-m, 2y-n) \tag{7}$$

$$HL(x,y) = \sum_m \sum_n h(m)\, g_2(n)\, I(2x-m, 2y-n) \tag{8}$$

$$HH(x,y) = \sum_m \sum_n g_3(m)\, g_4(n)\, I(2x-m, 2y-n) \tag{9}$$

where h, g_1, g_2, and g_3 are the filter coefficients.

Inverse DWT: The inverse DWT reconstructs the original image from the approximation and detail coefficients. It involves up sampling and filtering operations using synthesis filters $\tilde{h}$, $\tilde{g_1}$, $\tilde{g_2}$, and $\tilde{g_3}$. The reconstructed image $\hat{I}(x,y)$ is obtained by:

$$\begin{aligned} \hat{I}(x,y) = &\sum_m \sum_n \tilde{h}(m)\, \tilde{h}(n)\, LL\left(\frac{x-m}{2}, \frac{y-n}{2}\right) + \sum_m \sum_n \tilde{g_1}(m)\, \tilde{h}(n)\, LH\left(\frac{x-m}{2}, \frac{y-m}{2}\right) \\ &+ \sum_m \sum_n \tilde{h}(m)\, (\tilde{g_2})(n)\, HL\left(\frac{x-m}{2}, \frac{y-n}{2}\right) \\ &+ \sum_m \sum_n g_3(m)\, \tilde{g}(n)\, HH\left(\frac{x-m}{2}, \frac{y-n}{2}\right) \end{aligned} \tag{10}$$

where *LL*, *LH*, *HL*, and *HH* are the approximation and detail coefficients, and $\tilde{h}$, $\tilde{g}_1$, $\tilde{g}_2$ and $\tilde{g}_3$ are the synthesis filter coefficients.

Authors in [38] described a method that uses double wavelet transforms in a steganographic approach, enabling the insertion of secret text into the cover image. First, the cover image is decomposed into wavelet coefficients by applying 2D DWT, and the secret text is converted to its ASCII codes and then transformed into its wavelet coefficients with 1D DWT. According to the sign of these HH coefficients, the pixels of the cover image and text pixels are combined to create stego-pixels. Finally, the stego-image is formed by performing an inverse wavelet transformation of the concatenated quarters of waves. This embedded

technique improves the construction of a two-dimensional digital watermark by using a dual wavelet transform approach in both the spatial and frequency domains, offering higher security and robustness compared to single transform methods.

In recent work, authors in [39] suggested a steganography algorithm that employs Quaternion Fast Fourier Discrete Transform (QFFDT) to hide textual messages in cover images for maximum security and stability. They minimize an embedding function and maximize an extraction function. In this process, the cover image is altered using the quaternion Fourier transform and small ratios to embed the secret message. During extraction, the hidden message is retrieved from the stego-image using cropping operations and the inverse quaternion fast Fourier transform, minimizing data degradation. This method utilizes the ability of QFFDT to deal with quaternion-valued signals and offers the best approach to hiding textual messages within cover images without much distortion.

Authors in [40] introduced a steganographic method based on Singular Value Decomposition (SVD) and 2D DWT for embedding and extracting messages. SVD and 2D DWT are used to embed the process based on the message image and the cover image, respectively. The S matrix extracted through SVD is well concealed within the HH matrix of DWT, while the U and V matrices act as keys. In this process of feature extraction, the DWT of the stego-image is calculated to produce sub-bands, from where the S matrix is derived from the HH matrix and the U and V matrices are extracted to reconstruct the message image through the inverse SVD. The proposed multi-layered approach enhances resilience while maintaining a high PSNR. It also protects digital media from certain attacks, such as (Additive White Gaussian Noise (AWGN)). Authors in [41] introduced a method that employs wavelet domain transformation and adaptive weights to increase the robustness of data regarding data loss.

This process involves decomposing the original image into blocks and applying the second-generation 2D DWT to each block. An adaptive algorithm assigns weights to the wavelet coefficients, and blocks whose total 2D weights are below a preset threshold are transmitted. The coefficients are then converted into binary form, and the secret data is embedded in the Least Significant Bit (LSB) section. The extraction of secret data at the receiver end is only possible if both block partitioning and all block indices are transmitted securely. MSE, PSNR, and SSIM were used to compare the proposed method with existing methods, and visual results were also assessed. Although the adaptive algorithm and wavelet transformation offer optimal data embedding and extraction processes suitable for real-time applications, the high computational demands of these techniques can raise concerns regarding their impact on computational capacity.

This process involves decomposing the original image into blocks and applying the second-generation 2D DWT to each block using an adaptive algorithm. From them blocks with lower total weight are preferred being selected for the secret data embedding into their LSBs. Decoded blocks are returned to decimal representation; then the two-dimensional Inverse Discrete Cosine Transform is provided to get the stego image. It is evident that the block size influences performance metrics and accuracy, without compromising image quality.

Tables 4–6 show the strengths and weaknesses of techniques that use approaches in the transform domain. It provides a summary of transform domain techniques used in steganography, along with their advantages and drawbacks, as identified in various research articles. Advantages include expansion of the message capacity, improvement of message security, and ability to maintain picture quality; disadvantages include susceptibility to specific forms of attack, high computation costs for encryption and decryption procedures, and variation in the measure of security based on parameters such as the block size of the image. As will be discussed, each algorithm performs well in different scenarios, depending on factors such as flexibility to work with different image formats or resistance to certain types of attacks. However, factors like available computational power and acceptable levels of distortion are crucial in real-world applications.

Specifically, transform domain techniques play a vital role in advancing the capabilities of steganography. They offer the advantages of higher message-hiding capacity, enhanced security, and better image quality preservation, while also providing the flexibility needed to improve steganographic methods. Compared to standard techniques, each transform domain method has its own strengths and weaknesses. In this context, it is essential to recognize these distinctions to avoid misunderstandings and ensure the best course of action is taken in any given situation.

Table 4: Benefits and limitations of transform domain techniques (part 1)

Paper	Specific technique employed	Benefits	Limitations
[19]	LSB and MSB based image steganography using color images (RGB)	1. Enhanced security through MSB utilization. 2. Retention of image quality with high PSNR and low MSE. 3. Message embedding in imperceptible areas. 4. Security augmented by pixel location obfuscation.	1. Potential decrease in PSNR with larger payloads. 2. Complexity compared to traditional LSB-based methods. 3. Limited by image format compatibility.
[33]	Discrete Curvelet Transform applied to YCbCr selected cover	1. Utilizes cover statistics for enhanced security. 2. Multiscale transform for improved robustness. 3. Secret message embedded in middle frequencies for better imperceptibility.	1. Vulnerable to attacks affecting middle frequency coefficients.
[34]	Domain transformation of block embedding entropy from spatial to DCT domain	1. Incorporates statistics from both spatial and DCT domains. 2. Increases security by transforming spatial distortion measures into DCT domain.	1. Dependency on initial embedding schemes for spatial domain embedding.

Table 5: Benefits and limitations of transform domain techniques (part 2)

Paper	Specific technique employed	Benefits	Limitations
[35]	Dual-Tree Complex Wavelet Transform (DT-CWT) for embedding	1. Higher payload capacity due to DT-CWT's higher number of sub bands 2. Improved imperceptibility by reducing embedding errors 3. Enhanced security through adaptive intensity transformation and secret key usage	1. Higher decoding errors at higher DT-CWT transform levels and larger secret patch sizes 2. Complexity increases with smaller patch sizes, impacting computation resources
[36]	Edge detection filters (Laplacian, Prewitt, Sobel, Canny)	1. Enhanced security through edge-based hiding of secret information 2. Improved performance metrics (SNR, PSNR) with Prewitt and Canny edge detectors 3. Reliable embedding and compression using DCT	1. Reduction in image size due to compression 2. Potential side effects of compression on image quality may require further investigation 3. Limited to image files; extension to audio and video files for increased embedding capacity may be needed
[37]	DCT-SVD Steganography	1. High insertion capacity 2. Robustness 3. Preservation of image quality	1. Dependency on JPEG format 2. Computationally intensive 3. Sensitivity to compression
[38]	Double wavelet transforms	1. High imperceptibility 2. Multilevel security 3. Message length flexibility	1. Reduction in PSNR with increased message length 2. Complexity increases with higher DWT levels
[42]	Haar-DWT (HARR-DWT)	1. Enhanced data security 2. Minimal visible change in image 3. Improved handling of negative values	1. Vulnerable to statistical analysis 2. Susceptible to visual inspection 3. Limited capacity compared to deep learning-based techniques

Table 6: Benefits and limitations of transform domain techniques (part 3)

Paper	Specific technique employed	Benefits	Limitations
[39]	Quaternion Discrete Fourier Transform (QDFT)	1. Enhanced data security through quaternion domain 2. Utilization of maximum image capacity for text embedding 3. Concealed transmission of secret message within cover image.	1. Sensitivity to Variation in and Ratios 2. Potential Distortion of Transmitted Image with High Values 3. Dependency on Specific Image Formats (e.g., Tiff)
[40]	Singular Value Decomposition (SVD),Wavelet Transform (DWT)	1. (PSNR) for improved image quality 2. Robustness Against (AWGN) Attacks 3. Preservation of image quality during data hiding	1. Susceptibility to High Variance Noise Levels 2. Dependency on Message Image Size for Imperceptibility Effect 3. Potential Overhead in Computational Complexity for SVD and DWT Calculations
[41]	2D Discrete Wavelet Transform (2D DWT)	1. Adaptive selection of coefficients minimizes alterations to original image 2. High (PSNR) and (SSIM) 3. Compatibility with both gray-scale and RGB images	1. Sensitivity to Block Size and Message Size Selection 2. Potential increase in processing time with smaller block sizes 3. Dependency on secure transmission of block index for data extraction
[43]	Two-Dimensional Discrete Cosine Transform (2D DCT)	1. Adaptive selection of coefficients minimizes distortions in original image 2. High (PSNR) and (SSIM) 3. Immunity Against first order attacks like chi-square	1. Sensitivity to block size and window dimensions selection 2. Increased processing resources required for smaller window dimensions

4 Techniques Using Data Compression

This section discusses the techniques that use various algorithms to compress data before embedding it into cover images or text to reduce the size of the secret message while maintaining security. Data compression plays a pivotal role in steganography by reducing the size of the payload, which enhances embedding efficiency and minimizes the detectability of hidden data. It not only optimizes the use of storage space in cover media but also adds an additional layer of security by transforming the message into a compact format before embedding [13–15]. Integrating data compression into steganographic systems improves the embedding process's efficiency and reduces the detectability of the hidden message. This synergy between data compression and steganography provides a robust solution for secure and efficient communication

in scenarios requiring both data protection and resource optimization [44,45]. To begin, let's define key terminologies necessary for understanding these techniques [46]:

- Lossy Compression: Reduces the amount of data by eliminating unnecessary information. However, it results in a loss of some data.
- Lossless Compression: Reduces the amount of data without losing any information to ensure that the hidden data remains intact during compression and extraction.
- Entropy Coding: Shorter codes are assigned to more frequent data values and longer codes to less frequent ones using entropy coding based on their probability of occurrence.
- Run-Length Encoding (RLE): A count and a single value are substituted for sequential identical data values in RLE, a basic type of lossless compression.

Authors in [47] incorporated the Deflate compression algorithm with the LSB approach. Deflate compression uses LZ77 and Huffman coding techniques. LZ77 uses a dictionary-based mechanism where character matches are recorded using parameters such as offset, length, and codeword. Huffman coding produces a tree from character frequencies and represents characters as a binary code. LSB message embedding is done after the secret message is compressed. In manipulating the color channels of the cover image, LSB embedding is used in either red, green, or blue. In this way, integrating the Deflate compression algorithm with the LSB approach gives advantages in steganography. In the technique proposed by [48], the secret data is embedded after multilayer encoding, and False Positive Error (FPE) is applied using Huffman coding. This technique combines cryptography and compression techniques to tackle the problems of imperceptibility and security. The secret message undergoes multilayer encoding using FPE, which reduces its size while enhancing security. Thereafter, Huffman Coding is applied to it for further compression. The secret message is thus compressed through the Huffman tree so constructed.

To selectively embed information in high-texture regions, authors in [49] proposed the ADEDGEMME2 method, which utilizes matrix encoding and region selection to improve data hiding. It uses the MME2 method for adaptive embedding to enable multiple embedding solutions while maintaining pixel block complexity. Region selection relies on the complexity of pixel blocks to select areas where the texture is high so that distortion is minimized. The approach estimates complexity through the use of pixel differences, whereby blocks with complexity above 1.0 are chosen for embedding. While this method improves security against various attacks, its reliance on pixel block complexity for region selection may limit its applicability to certain types of images, particularly those with uniform textures.

Authors in [50] proposed a new technique called HEWTEA-IS. This method applies Haar DWT for decomposition of the image into frequency bands. For the selection of pixels, it uses a White-Tailed Eagle Algorithm (WTEA). Huffman encoding provides lossless data compression. During extraction, DWT transforms the stego images back into spatial representation. WTEA locates the hidden information, and Huffman decoding will produce the secret message. The chaotic map-based WTEA optimizes pixel selection, thereby enhancing the security of the method. By using the WTEA and Huffman encoding, the method provides robustness against detection. Using Haar (DWT) and taking care in choosing the frequency bands, the method ensures secrecy while maintaining image quality.

Method proposed in [45] utilizes the Goldbach G0 code for text compression. It first sorts the characters based on how frequently they appear and in what order. Then, to each character, its binary equivalent is associated with the codeword from the first two prime numbers from the computation of 2(n+3). Then, the codeword is substituted for each character of the text. After that, LSB hides the text within the cover image. This technique embeds the compressed text into the cover image using LSB, arranging characters according to their frequency of occurrence. As a result, the data is efficiently hidden with the least amount

of processing overhead. The 2DRLE method, which uses lossless text image compression, was proposed by [51] to generate a compressed image with a smaller file size compared to the original. Three stages will comprise the 2DRLE method: row scanning, column scanning, and entropy coding. Row scanning involves the storage of consecutive repeated rows of the image as single rows; column scanning will then be applied to the resulting image, identifying and storing consecutive repeating columns. The resulting symbol set from both row and column scanning is encoded using Huffman coding, further reducing the image size by representing the most frequent symbols with shorter codes. This method significantly reduces the image size by compressing white space repetition.

Tables 7 and 8 outline the advantages and drawbacks of data compression techniques integrated with transform-domain steganography methods. The steganography techniques include techniques such as Huffman compression, DCT, and Deflate compression. Benefits are lossless compression, enhanced security, enhanced image quality, and increased hiding capacity. The deficiencies arising from these techniques include potential loss of data during compression, dependency on image quality, and complexity in managing the encryption key. Each of the approaches offers specific advantages, such as efficient embedding or additional security layers, but practical implementation has to take into consideration challenges like sensitivity of parameters and data integrity to realize optimized data storage, efficiency in transmission, and robust security in actual scenarios.

Table 7: Benefits and limitations of various data compression techniques integrated with steganographic techniques (part 1)

Paper	Specific technique employed	Benefits	Limitations
[47]	Deflate Compression Algorithm (Combining LZ77 and Huffman Coding) with Least Significant Bit (LSB) embedding	1. Significant reduction in text size through deflate compression 2. Higher performance in terms of MSE and PSNR compared to baseline 3. Efficient combination of compression and embedding techniques	1. Potential loss of data during compression process 2. Dependency on image quality for effective LSB embedding
[48]	Multilayer encoding with Format Preserving Encryption (FPE) and Huffman coding	1. High security ratio 2. Increased hiding capacity 3. Improved imperceptibility	1. Key management complexity 2. Vulnerability to cryptanalysis 3. Dependence on encoding techniques
[49]	ADEDGE_MME image steganography	1. Exploitation of high-textured image regions 2. Adaptive embedding process using multi-bit layers 3. Complexity preservation in stego images	1. Potential decrease in embedding capacity with very high texture images 2. Dependency on shared keys for data extraction

(Continued)

Table 7 (continued)

Paper	Specific technique employed	Benefits	Limitations
[50]	DWT with Huffman encoding and WTEA)	1. Secrecy preservation without compromising image quality 2. Robustness against unauthorized access 3. Additional layer of security provided by Huffman encoding. 4. Optimized embedding of secret bits using WTEA	1. Sensitivity to parameter tuning in WTEA 2. Dependency on image quality and resolution 3. Potential degradation of image fidelity with high embedding rates

Table 8: Benefits and limitations of various data compression techniques integrated with steganographic techniques (part 2)

Paper	Specific technique employed	Benefits	Limitations
[45]	Goldbach G0 code with LSB embedding	1. Enhanced security 2. Improved image quality 3. Efficient storage	1. Sensitivity to image manipulation 2. Limited capacity 3. Dependency on prime numbers
[51]	2DRLE method	1. Double layer of data security 2. Compression for efficient data embedding 3. LSB Embedding for concealment	1. Potential loss of data integrity 2. Sensitivity to image modifications

5 Techniques Using Cryptography with Data Compression

Techniques that use cryptography first encrypt the secret message using cryptographic algorithms to ensure data security. Then, they compress the encrypted message to reduce data size for efficient transmission and storage within cover images or texts. This dual layered approach enhances both security and efficiency: encryption ensures confidentiality, while compression reduces the data footprint, making it more difficult to detect and intercept during transmission. For instance, symmetric encryption algorithms such as AES (Advanced Encryption Standard) or DES (Data Encryption Standard) are commonly used to encrypt messages before compression. The encrypted data, being more randomized, also benefits from entropy-based compression techniques, which further reduce its size. These methods are particularly useful in scenarios where both security and efficient use of cover medium capacity are crucial, such as secure multimedia sharing or covert communications in bandwidth-constrained environments [14,44,45]. Here are some commonly used terminologies for understanding techniques in this domain.

- Encryption: The process of transforming readable data, or plaintext, into ciphertext is called encryption.
- Decryption: The process where ciphertext is converted back into plaintext.
- Symmetric Encryption: A single key is used for both encryption and decryption in symmetric encryption.
- Asymmetric Encryption: A public key is used for encryption, and a private key is used for decryption in asymmetric encryption.

Authors in [52] proposed method for text steganography, which makes use of Unicode characters and encryption techniques for concealing a secret message in a cover text. It involves the encoding of the secret message by utilizing the AES encryption algorithm and embedding it within the cover text by using Unicode characters. The embedded message is then compressed using the Huffman algorithm to reduce the stego-text size, achieving high compression efficiency. This approach guarantees both security and data compression efficiency regarding concealing information in the text documents. The algorithm proposed by [53] combines RSA encryption with various data compression techniques, such as Huffman coding, RLE, or DWT, in steganography. The secret message is first encrypted using RSA during the embedding process and then inserted into the LSB of the cover image. The stego-image is then compressed using the RLE, DWT, or Huffman coding methods. The stego-image is decompressed, the message is extracted from the LSBs, and RSA is used to decode it. RSA-based encryption provides the line of defence for the secure transmission of the message. The role of Huffman coding, RLE, or DWT is to reduce the size of the stego-image and optimize the storage and transmission. While compression reduces the file size, it may also cause some information loss, potentially affecting image quality. To share the symmetric key securely, the system proposed by [54] uses both cryptography and steganography in tandem. Through AES encryption with symmetric key sharing, messages are encrypted and hidden in cover images, thereby evading visual detection. SCrypt hashing algorithm enhances security on shared keys against dictionary attacks. Compressing the messages, AES encrypts in counter mode CBC, encoding and embedding in cover images using LSB. The message is recovered through the extraction of stego-objects and password validation via the SCrypt hashing algorithm. Integration of AES encryption and SCrypt hashing enhances the confidentiality and integrity of messages. Authors in [55] proposed a technique combining a variant of the Collatz Conjecture with Diffie-Hellman Key Exchange to establish a shared secret key and generate unique random locations for LSB-based image steganography. By modifying the Collatz Conjecture, unique random numbers are generated iteratively from a given secret key. These generated random numbers determine the pixels for hosting the secret message bits in a way that is both secure and robust. Encryption is performed by XORing the secret message with the random numbers represented in binary. The Diffie-Hellman Key Exchange provides a mechanism for generating the keys without actually sending the key over the internet; hence, the possibility of its interception is reduced to a minimum. Successful implementation would require that the shared secret key generated through the Diffie-Hellman Key Exchange be kept secure to avoid access by unauthorized users and thus ensure confidentiality. Authors in [56] proposed system that combines DWT compression, AES encryption, and LSB steganography for advanced data security. In the embedding phase, the secret image is first compressed using DWT and then encrypted using AES prior to its embedding into the cover image using LSB. In the extraction phase, the cipher text extracted from the stego-image is decrypted using AES and decompressed using DWT to retrieve the secret image. Huffman coding and achromatic components are utilized in this approach to enhance data encoding and representation. DWT compression may deteriorate the image quality, especially with increased compression ratios. Authors in [57] proposed the two-stage encryption, one-stage steganography approach for data security enhancement, wherein the integration of steganography and cryptography is done. First, the important text is divided into two parts, encrypted separately by the Caesar Cipher and Vigenère Cipher. Thereafter, the ciphertext is converted into Morse

code to further encrypt the ciphertext. Finally, the ciphertext written in Morse code is embedded inside an image with the help of the LSB technique to hide the encrypted message within the cover image. Multi-layering is the best approach for the encryption of the data and transmitting it safely, and it reduces the file size through compression techniques like Morse coding and LSB embedding. The encryption methodology proposed by [58] consists of two major stages: modified Caesar Cipher encryption and Card Deck Shuffle Rearrangement. The first part consists of a modified version of the Caesar Cipher algorithm that encrypts image pixels with variable shifts determined by keys extracted from the master key for added security and to avoid easy decryption. The second part of the process is the pixel rearrangement using a transposition Cipher adapted to the riffle shuffle technique. The two-fold encryption strengthens data security and makes decryption more difficult. Modifying the Caesar Cipher with variable shifts and using a transposition cipher for pixel rearrangement significantly enhance data security, making decryption much more difficult for attackers. The effectiveness of the encryption scheme depends on the proper setting of parameters, such as key sizes and the number of sub-decks, which must be carefully tuned for optimal performance.

The proposed encryption algorithm of [59] presents a novel approach using frequency domain compression and lightweight chaos. It uses dynamic key generation with plaintext correlation, applying chaotic sequences for encryption. Dynamic key generation, time domain shifting, DCT coding, quantification, coefficient extraction, compression coding, coefficient scrambling, and diffusion encryption are the main steps included in this algorithm. The algorithm transforms the given plaintext images to the corresponding cipher text images by passing through a series of domain transformations, including frequency domain mapping and chaotic sequence-based encryption. This work uses dynamic keys and chaotic sequences, which enhance information security, making the encrypted information difficult for unauthorized users to interpret. The technique proposed by [60] provides an efficient encryption and decryption process and is fit for real-time applications with secure data transferring. It combines AES crypto-algorithm with Dynamic XOR for image steganography. AES handles the secure encryption and decryption, while Dynamic XOR conceals the message in the image. During encryption, the message is first converted to binary bits, and each pixel value is XORed with a dynamic key derived from the message.

Subsequently, the resulting image is encrypted using AES. Reversing this process yields the decrypted image, from which the hidden message can be extracted the dynamic XOR hides the message in image without any distortion of visual quality; thus, from the visual point of view, it would be hard to detect for third parties. The handling of keys, particularly for Dynamic XOR, requires meticulous care to prevent unauthorized access and ensure data integrity.

As per Tables 9 and 10, cryptographic and data compression methods combined with transform domain steganography to offer advantages in security, high hiding capacity, and data transmission efficiency but are limited by factors like computational complexity, susceptibility to different attacks, capacity constraints, and image distortion. RSA, AES, Huffman coding, and LSB encoding are some of the techniques that improve data confidentiality and robustness; however, some factors need to be taken into consideration, such as key management, algorithm dependency, and the computational overhead the solution brings with it. This integration of cryptography and data compression creates a robust framework for secure communication. By encrypting the message before compression, the system achieves layered protection, ensuring that even if the compression is reverse-engineered, the data remains secure due to the encryption. This dual technique is particularly beneficial for applications requiring high security and optimal storage or transmission efficiency, such as secure financial transactions and military communications.

Table 9: Benefits and limitations of cryptographic and data compression methods integrated with steganography techniques (part 1)

Paper	Year of publication	Specific technique employed	Benefits	Limitations
[52]	2021	AES encryption with Huffman compression	1. Improved security 2. High payload capacity 3. Cognitive Transparency Efficient Data Hiding	1. Complexity 2. Detection risk 3. Capacity constraint 4. Dependency on unicode support
[53]	2021	RSA and Huffman coding, RLE, or DWT combination	1. Secure transmission 2. Compression efficiency 3. Concealed message embedding 4. High-quality stego-image preservation	1. Vulnerability to attacks 2. Limited embedding capacity 3. Potential image distortion 4. Algorithm dependency
[54]	2020	1. Password validation with SCrypt algorithm 2. AES encryption with CBC mode	1. Strong security 2. Password authentication 3. Robustness 4. Data concealment 5. Efficient communication	1. Limited capacity 2. Detection possibility 3. Performance overhead 4. Dependency on image format
[55]	2022	1. Diffie-Hellman key exchange 2. Modified collatz conjecture 3. LSB-based image steganography	1. Shared secret key establishment 2. Unique random number generation 3. Secret message encryption 4. Imperceptible embedding	1. Limited embedding capacity 2. Complexity in key exchange 3. Sensitivity to collatz conjecture parameters

Table 10: Benefits and limitations of cryptographic and data compression methods integrated with steganography techniques (part 2)

Paper	Year of publication	Specific technique employed	Benefits	Limitations
[56]	2022	1. DWT 2. Advanced Encryption Standard (AES)	1. Improved data security and capacity 2. Utilizes hybrid layers of security for enhanced protection 3. Maintains good quality of stego-images	1. Susceptible to detection if LSB is not distributed evenly 2. Potential distortion of cover image due to LSB embedding 3. Requires careful selection of encryption key for AES to ensure security

(Continued)

Table 10 (continued)

Paper	Year of publication	Specific technique employed	Benefits	Limitations
[57]	2023	1. Caesar cipher 2.Vigenere cipher 3. Morse code 4. Least Significant Bit (LSB) Technique	1. Enhanced security 2. Increased robustness 3. Data concealment 4. Visual imperceptibility	1. Key management 2. Increased complexity 3. Susceptibility to attacks 4. Limited capacity
[58]	2023	Modified caesar cipher encryption and card deck shuffle rearrangement	1. Sensitivity to key changes 2. Complexity of implementation 3. Vulnerability to cryptanalysis	1. Enhanced security 2. Noise resistance 3. Lossless recovery
[59]	2023	Chaotic sequences and block permutation for encryption	1. Computational complexity 2. Vulnerability to key attacks 3. Sensitivity to initial conditions	1. Enhanced security 2. Compression efficient 3. Nonlinearity
[60]	2024	Dynamic 8-bit XOR combined with AES crypto algorithm	1. Enhanced security 2. Efficient data transfer 3. Imperceptible changes to image	1. Increased computational complexity 2. Vulnerability to cryptanalysis 3. Potential loss of image quality

6 Summarized Results of above Discussed Techniques

The Tables 11–13 summarize findings of various image steganography techniques. The metrics include the size of the cover image, payload (bits embedded), (PSNR), Mean Squared Error (MSE), and Embedding Rate. Authors in [19] used various image formats, including JPEG, PNG, and BMP, for images such as Mandrill, Lena, Baboon, Boy, Peppers, and Monarch. This study reports high PSNR values (around 70–73 dB), suggesting excellent image quality after embedding, with very low MSE values (0.0032–0.0057) and a low embedding rate (0.006–0.007). Authors in [20] focus on smaller images (32×32 to 121×121), achieving PSNR values between 55.58 and 67.24 dB and very high embedding rates (8 bits per pixel). These results suggest significant data embedding capacity, though quality diminishes as MSE with image size. Authors in [21] report extremely low PSNR values (0.48–0.5 dB) and very high MSE (up to 234714.501), indicating significant quality degradation for images such as Lena and Cat. [24] and [25] both deal with high payloads (over 1 million bits), achieving moderate PSNR (46.69–53.99 dB) and MSE values (around 1.046–5.617), with embedding rates between 4.46 and 5.25. Authors in [26] achieved high payloads (over 2 million bits) while maintaining PSNR values between 35.15 and 40.34 dB and moderate MSE, demonstrating a balance between capacity and quality. [28] and [30] show consistent performance with PSNR values around 31.94 to 54.15 dB and varied MSE

values, reflecting different embedding capacities and image qualities. [37] consistently embed 257,600 bits, with PSNR values ranging from 38.69 to 54.98 dB, reflecting varying image quality across different images. [53] evaluate several images with moderate payloads, achieving PSNR values around 40 dB and MSE values near 2.54, demonstrating balanced performance. Finally, authors in [47] reported very high PSNR values (75.77–76.03 dB) and very low MSE (0.0016–0.0017), suggesting excellent quality with an exceptionally high embedding rate (55.775). This comprehensive summary offers insights into the trade-offs between embedding capacity, image quality, and payload across different steganography techniques.

Table 11: Summarized results of discussed techniques (part 1)

Paper	Cover image	Size of image	Payload (bits)	PSNR (dB)	MSE	Embedding rate (bpp)
[19]	Mandi.JFIF	768 × 508	2426	72.023	0.0044	0.0062182
	Lena.PNG	512 × 512	1667	72.242	0.0038	0.0063591
	Baboon.Bmp	500 × 480	1738	70.534	0.0057	0.,
	Boy.JPEG	768 × 512	2426	72.897	0.0033	0.0061696
	Peppers.PNG	512 × 512	1748	71.656	0.0044	0.0066681
	Monarch.JFIF	768 × 508	2370	73.032	0.0032	0.0060747
[20]	–	32 × 32	8192	67.24	0.000193	8
	–	45 × 45	16,384	64.19	0.000772	8.0908642
	–	64 × 64	32,768	61.17	0.003129	8
	–	90 × 90	65,536	58.18	0.012316	8.0908642
	–	114 × 114	104,864	56.11	0.031828	8.0689443
	–	121 × 121	117,968	55.58	0.040511	8.0573731
[21]	Lena	512 × 512	512	0.48	234714.5	0.0019531
	–	512 × 512	131,072	0.5	233636.1	0.5
	Cat	512 × 512	394	0.48	234714.5	0.001503
	–	512 × 512	25,208	0.49	234174.7	0.0961609
[24]	–	512 × 512	1,224,351	53.99	1.04602	4.6705284
	–	512 × 512	1,377,534	46.69	5.617459	5.2548752
	–	512 × 512	1,223,540	53.94	1.058132	4.6674347
	–	512 × 512	1,169,652	51.97	1.665482	4.4618683
	–	512 × 512	1,256,278	51.3	1.9433	4.7923203
	–	512 × 512	1,282,361	51.64	1.796966	4.891819
[26]	Tiffany	512 × 512	2,366,437	37.2	49.95051	9.0272408
	Peppers	512 × 512	2,371,837	35.15	80.08292	9.0478401
	Jet	512 × 512	2,377,031	39.52	29.2779	9.0676537
	Boat	512 × 512	2,405,009	35.32	77.00872	9.1743813
	House	512 × 512	2,399,229	37.58	45.76568	9.1523323
	Pot	512 × 512	2,359,271	40.34	24.24041	8.9999046
	Average	512 × 512	2,394,086	37.49	46.72399	9.1327133
[28]	Lena	512 × 512	1,057,962	33.49	117.3654	4.0358047
	Pepper	512 × 512	1,057,060	33.12	127.8027	4.0323639
	Baboon	512 × 512	1,070,892	31.94	167.7027	4.0851288

(Continued)

Table 11 (continued)

Paper	Cover image	Size of image	Payload (bits)	PSNR (dB)	MSE	Embedding rate (bpp)
	Boat	512 × 512	1,068,110	32.09	162.0093	4.0745163
	Airplane	512 × 512	1,063,578	32.23	156.87	4.0572281
	Man	512 × 512	1,062,088	32.95	132.9046	4.0515442

Table 12: Summarized results of discussed techniques (part 2)

Paper	Cover image	Size of image	Payload (bits)	PSNR (dB)	MSE	Embedding rate (bpp)
[25]	Lena	512 × 512	1,224,351	53.99	1.046	4.6705284
	Baboon	512 × 512	1,377,534	46.69	5.6175	5.2548752
	Airplane	512 × 512	1,223,540	53.94	1.0581	4.6674347
	Peppers	512 × 512	1,169,652	51.97	1.6655	4.4618683
	Car	512 × 512	1,256,278	51.3	1.9433	4.7923203
	Boat	512 × 512	1,282,361	51.64	1.797	4.891819
	Baboon	512 × 512	74,047	39.305	30.764	0.2824669
	Peppers	512 × 512	74,047	39.15	31.882	0.2824669
	Jet	512 × 512	74,047	38.696	35.395	0.2824669
	Animal	512 × 512	1,060,748	32.21	157.59	4.0464325
	Baboon	512 × 512	1,054,327	31.74	175.61	4.0219383
	Boat	512 × 512	1,051,124	32.84	136.31	4.0097198
	City	512 × 512	1,049,284	32.06	163.13	4.0027008
	Gatbawi	512 × 512	1,057,086	30.66	225.19	4.0324631
	House	512 × 512	1,051,474	32.73	139.81	4.011055
	Iseland	512 × 512	1,052,513	32.84	136.31	4.0150185
	Lena	512 × 512	1,049,742	33.21	125.18	4.0044479
	Lotus	512 × 512	1,051,584	33.6	114.43	4.0114746
	Man	512 × 512	1,052,264	32.72	140.13	4.0140686
	Pepper	512 × 512	1,050,571	33.55	115.76	4.0076103
	Average	512 × 512	1,052,641	32.61	143.73	4.0155067
	Baboon	512 × 512	2,502,616	33.76	110.29	9.5467224
[30]	Lena	512 × 512	77,244	54.1489	1.0084	0.2946625
	Baboon	512 × 512	77,244	49.9544	2.6491	0.2946625
	Pepper	512 × 512	77,244	54.0779	1.0251	0.2946625
	Jet	512 × 512	77,244	53.7585	1.1033	0.2946625
[47]	Airplane	512 × 512	4700	76.03	0.0016	55.775319
	Lisa	512 × 512	4700	75.81	0.0017	55.775319
	Peppers	512 × 512	4700	75.77	0.0017	55.775319
	Baboon	512 × 512	4700	75.8	0.0017	55.775319

Table 13: Summarized results of discussed techniques (part 3)

Paper	Cover image	Size of image	Payload (bits)	PSNR (dB)	MSE	Embedding rate (bpp)
[37]	Lena	512 × 512	257,600	42.63	3.4574	0.982666
	Pepper	512 × 512	257,600	40.14	6.1459	0.982666
	Boat	512 × 512	257,600	52.94	0.33	0.982666
	Barbara	512 × 512	257,600	42.1	4.0008	0.982666
	Golhill	512 × 512	257,600	54.98	0.2	0.982666
	Zelda	512 × 512	257,600	43.65	2.8059	0.982666
	Mandrill	512 × 512	257,600	38.69	0.5812	0.982666
	Boy	512 × 512	25,000	40.105	2.44	1.51
	Girl	512 × 512	30,178	41.015	2.21	1.35
	Apple	512 × 512	33,145	40.61	3.2	1.47
	Bear	512 × 512	43,123	40.4	2.96	1.41
	Lena	512 × 512	22,652	40.02	2.95	1.06
	Average	512 × 512	32,063	40.31	2.72	1.32

7 Conclusion

This review presents a comprehensive overview of the evolving field of image steganography, exploring a range of techniques across both spatial and transform domains. Traditional methods like LSB embedding are simple but prone to detection, while advanced techniques, including hybrid and transform domain approaches, offer enhanced security and efficiency at the cost of increased computational complexity. The integration of steganography with data compression and cryptographic methods further bolsters security and transmission efficiency, albeit with trade-offs such as computational overhead and potential image distortion. Achieving the right balance between these factors is crucial for practical applications. Despite notable advancements, challenges like computational complexity, vulnerability to attacks, and image quality degradation persist, emphasizing the need for ongoing innovation in the field.

Future Scope:

The future of image steganography lies in addressing its current limitations while leveraging emerging technologies. Key directions include:

- Optimization of Hybrid Methods: Developing hybrid techniques that combine the strengths of different approaches while mitigating their weaknesses.
- Application of AI and ML: Incorporating machine learning and artificial intelligence to enhance steganographic systems' security, adaptability, and efficiency.
- Standardization: Creating standardized protocols and tools for seamless integration across platforms to facilitate broader adoption.
- Advancements in Data Compression and Cryptography: Exploring innovative combinations of compression and cryptographic techniques to improve security and transmission efficiency.
- Real-World Application: Focusing on practical implementations in areas like digital forensics, secure communication, and copyright protection, ensuring robust solutions for increasingly digital interactions.

By addressing these opportunities, researchers and practitioners can advance the field of image steganography to meet the growing demands of secure and efficient digital communication.

Acknowledgement: The author would like to express sincere gratitude to Mr. Harsh S Deshmukh, Dr. S. U. Mane, Computer Science Engineering Department, Rajarambapu Institute of Technology, Rajaramnagar for their support in completing the literature survey.

Funding Statement: The author declares that no funding was received for the preparation or completion of this review paper.

Availability of Data and Materials: All the papers referenced in this review are publicly available through various academic databases, including IEEE Xplore, Google Scholar, ResearchGate, ScienceDirect, and Springer. The materials referenced in the review can be accessed from these platforms.

Ethics Approval: None.

Conflicts of Interest: The author declares no conflicts of interest to report regarding the present study.

Acronyms

AWGN	Additive White Gaussian Noise
DCT	Discrete Cosine Transform
DPVD	Dynamic Pixel Value Differencing
EMD	Earth Mover's Distance
IWT	Integer Wavelet Transform
MSB	Most Significant Bit
PVD	Pixel Value Differencing
SSIM	Structural Similarity Index Measure
BPP	Bits Per Pixel
DFT	Discrete Fourier Transform
DWT	Discrete Wavelet Transform
FPR	False Positive Error
LSB	Least Significant Bit
PSNR	Peak Signal-to-Noise Ratio
QI	Quality Index
WTEA	White-Tailed Eagle Algorithm

References

1. Saravanan V, Ramachandran M, Soundharaj S. Exploring various digital communication and its classification. Renew Nonrenewable Energy. 2022;1:52–7. doi:10.46632/rne/1/1/9.
2. Digital technology's threats to human security. In: 2022 special report on human security; 2022. p. 65–75. doi:10.18356/9789210014007c006.
3. Yadav R. Analysis of cryptography in information technology. Int J Sci Res Eng Manag. 2023;7(3). doi:10.55041/IJSREM18379.
4. Chandra G, Verma SB, Yadav AK. Cryptography techniques for information security. In: Computational intelligent security in wireless communications; 2022. p. 191–200. doi:10.1201/9781003323426-11.
5. Thakre N, Junghare S, Sakhre P, Khawse D. Steganography a technique to hide information within an image. J Cyber Secur, Priv Issues Chall. 2023;2:14–9. doi:10.46610/JCSPIC.2023.v02i01.003.
6. Aslan M, Aktuğ SS, Ozkan-Okay M, Yılmaz AA, Akin E. A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. Electronics. 2023;12(6):1333. doi:10.3390/electronics12061333.
7. Winsley BB, Muthukannan M. Information security. In: Advances in library and information science. IGI Global. 2020. doi:10.4018/978-1-7998-1482-5.ch017.
8. Kumar A, Pooja K. Steganography-a data hiding technique. Int J Comput Appl. 2010;9(7):19–23. doi:10.5120/1398-1887.

9. Singla D, Verma N, Patni S. A review on spatial and transform domain-based image steganography. In: Advances in multimedia and interactive technologies. IGI Global. 2023. doi:10.4018/978-1-6684-6864-7.ch010.
10. Febryan A, Purboyo T, Saputra R. Steganography methods on text, audio, image and video: a survey. Int J Appl Eng Res. 2017 Jan;12:10485–90.
11. Kaur R, Singh B. A robust and imperceptible n-Ary based image steganography in DCT domain for secure communication. Multimed Tools Appl. 2024;83:20357–86. doi:10.1007/s11042-023-16330-9.
12. Yahya A. Introduction to steganography. In: Steganography techniques for digital images; 2018 Jun 13. doi:10.1007/978-3-319-78597-4_1.
13. Sidar S, Chandel G, Garg D. Hybrid technique of data security by ECS (Encryption, Compression, Steganography). Int J Adv Res Ideas Innov Technol. 2020;6:753–8.
14. Sethi P, Kapoor V. A proposed novel architecture for information hiding in image steganography by using genetic algorithm and cryptography. Procedia Comput Sci. 2016;87:61–6. doi:10.1016/j.procs.2016.05.127.
15. Mishra R, Mishra A, Bhanodiya P. An edge based image steganography with compression and encryption. In: 2015 International Conference on Computer, Communication and Control (IC4); 2015; Indore, India. p. 1–4. doi:10.1109/IC4.2015.7375510.
16. AL-Shaaby AA, AlKharobi T. Cryptography and steganography: new approach. Trans Netw Commun. 2017;5(6). doi:10.14738/tnc.56.3914.
17. Vasava D, Doshi N. Study and analysis of network steganography methods. Vol. 311, In: Choudrie J, Mahalle P, Perumal T, Joshi A, editors. ICT with intelligent applications. Smart innovation, systems and technologies. Singapore: Springer; 2023. doi:10.1007/978-981-19-3571-8_9.
18. Sharma N, Batra U. A review on spatial domain technique based on image steganography. In: 2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). Gurgaon, India: IEEE; 2017. p. 24–7. doi:10.1109/IC3TSN.2017.8284444.
19. Mahdi AS. An improved method for combine (LSB and MSB) based on color image RGB. Eng Technol J. 2021;39:231–42. doi:10.30684/etj.v39i1B.1574.
20. Subong RA, Fajardo AC, Kim YJ. Adaptive bit rotation and inversion scoring: a novel approach to LSB image steganography. In: 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management (HNICEM); 2018 Nov 29–Dec 2. Baguio City, Philippines: IEEE; 2018. p. 1–6. doi:10.1109/HNICEM.2018.8666228
21. Al-Momin MMSA, Abed IA, Leftah HA. A new approach for enhancing LSB steganography using bidirectional coding scheme. Int J Electr Comput Eng. 2019;9:5286–94. doi:10.11591/ijece.v9i6.pp5286-5294.
22. Swain G. Advanced digital image steganography Using LSB, PVD, and EMD: emerging research and opportunities. IGI Global. 2019. doi:10.4018/978-1-5225-7516-0.
23. Wu D, Tsai W. A steganographic method for images by pixel-value differencing. Pattern Recognit Lett. 2003;24:1613–26. doi:10.1016/S0167-8655(02)00402-6.
24. Hosam O, Ben Halima N. Adaptive block-based pixel value differencing steganography. Secur Commun Netw. 2016;9:5036–50. doi:10.1002/sec.1676.
25. Abdel Hameed M, Aly S, Hassaballah M. An efficient data hiding method based on adaptive directional pixel value differencing (ADPVD). Multimed Tools Appl. 2017;77:14705–23. doi:10.1007/s11042-017-5056-4.
26. Swain G. High capacity image steganography using modified LSB substitution and PVD against pixel difference histogram analysis. Secur Commun Netw. 2018:1–14. doi:10.1155/2018/1505896.
27. Pradhan KRS, Swain G. Digital image steganography using LSB substitution, PVD, and EMD. Math Probl Eng. 2018:1804953. doi:10.1155/2018/1804953.
28. Hussain M, Riaz Q, Saleem S, Ghafoor A, Jung KH. Enhanced adaptive data hiding method using LSB and pixel value differencing. Multimed Tools Appl. 2021;80:20381–401. doi:10.1007/s11042-021-10652-2.
29. Abd AS, Hussein EAR. Design secure multi-level communication system based on duffing chaotic map and steganography. Indones J Electr Eng Comput Sci. 2022;25:238–46. doi:10.11591/ijeecs.v25.i1.pp238-246.
30. Yassin N. Data hiding technique for color images using pixel value differencing and chaotic map. Jordan J Comput Inf Technol. 2022;8(3):242–55. doi:10.5455/jjcit.71-1642508824.

31. Kosuru SNVJD, Pradhan A, Basith KA, Sonar R, Swain G. Digital image steganography with error correction on extracted data. IEEE Access. 2023;11:80945–57. doi:10.1109/ACCESS.2023.3300918.
32. Gnanitha G, Swetha A, Teja GS, Vasavi DS, Sirisha BL. Review on image steganography transform domain techniques. In: Intelligent manufacturing systems in Industry 4.0. Singapore: Springer; 2023. p. 501–12. doi:10.1007/978-981-99-1665-8_43.
33. Gharavi H, Rajaei B. A robust steganography algorithm based on curvelet transform. In: Electrical Engineering (ICEE), Iranian Conference on; 2018 May 8–10; Mashhad, Iran: IEEE; 2018. p. 1624–8. doi:10.1109/icee.2018.8472443.
34. Hu X, Ni J, Shi YQ. Efficient JPEG steganography using domain transformation of embedding entropy. IEEE Signal Process Lett. 2018;25:773–7. doi:10.1109/LSP.2018.2818674.
35. Kadhim IJ, Premaratne P, Vial PJ. Secure image steganography using dual-tree complex wavelet transform block matching. In: 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA); 2018 Mar 29–31. Coimbatore, India: IEEE; 2018. p. 41–7. doi:10.1109/iceca.2018.8474616.
36. Ayub N, Selwal A. An improved image steganography technique using edge based data hiding in DCT domain. J Interdiscipl Math. 2020;23:357–66. doi:10.1080/09720502.2020.1731949.
37. Tchakounté F, Kamdem P, Kamgang J, Tchapgnouo H, Atemkeng M. An efficient DCT-SVD steganographic approach applied to JPEG images. EAI Endorsed Trans Ind Netw Intell Syst. 2020;7:166365. doi:10.4108/eai.28-9-2020.166365.
38. Oudah KM, Abed NA, Khudhair SR, Kaleefah MS. Improvement of image steganography using discrete wavelet transform. Eng Technol J. 2020;38:83–7. doi:10.30684/etj.v38i1A.266.
39. ElSharkawy MI. Using quaternion fourier transform in steganography systems. Int J Commun Netw Inform Security. 2022;10(2). doi:10.17762/ijcnis.v10i2.3266.
40. Singh J, Singla M. Image steganography technique based on singular value decomposition and discrete wavelet transform. Int J Electric Electronic Res. 2022;10:122–5. doi:10.37391/IJEER.
41. Alobaidi T, Mikhael W. An adaptive steganography insertion technique based on wavelet transform. J Eng Appl Sci. 2023;70(1):144. doi:10.1186/s44147-023-00300-x.
42. BrahmaNaidu N, RamaKrishna K, Diyyala K, Sai Swaroop M, Sandeep K. Image steganography using modified DWT technique. Int J Food Nutr Sci. 2023;11(12). doi:10.48047/ijfans/v11/i12/212.
43. Alobaidi T, Mikhael W. An adaptive steganography insertion technique based on cosine transform. Iraqi J Electric Electron Eng. 2024;20:45–58. doi:10.37917/ijeee.
44. Wahab OFA, Khalaf A, Hussein A, Hamed H. Hiding data using efficient combination of RSA cryptography and compression steganography techniques. IEEE Access. 2021;9:31805–15. doi:10.1109/ACCESS.2021.3060317.
45. T.Arroyo JC. LSB image steganography with data compression technique using goldbach G0 code algorithm. Int J Emerg Trends Eng Res. 2020;8:3259–64. doi:10.30534/ijeter/2020/62872020.
46. Jayasankar U, Thirumal V, Dhavachelvan P. A survey on data compression techniques: from the perspective of data quality, coding schemes, data type and applications. J Saud Univ Comput Inform Sciences/Mağalaẗ Ğam'aẗ Al-Malīk Saud: Ùlm Al-Ḥasib Wa Al-Ma'lumat. 2021;33:1–20. doi:10.1016/j.jksuci.2018.05.006.
47. Tayyeh HK, Ahmed AL-Jumaili AS. A combination of least significant bit and deflate compression for image steganography. Int J Electr Comput Eng. 2022;12(1):358–64. doi:10.11591/ijece.v12i1.pp358-364.
48. Majeed MA, Sulaiman R, Shukur Z. New text steganography technique based on multilayer encoding with format-preserving encryption and huffman coding. Int J Adv Comput Sci Appl. 2022;13. doi:10.14569/issn.2156-5570.
49. Nguyen TD, Le HQ. A secure image steganography based on modified matrix encoding using the adaptive region selection technique. Multimed Tools Appl. 2022;81:25251–81. doi:10.1007/s11042-022-12677-7.
50. Alkhliwi S. Huffman encoding with white tailed eagle algorithm-based image steganography technique. Eng, Technol Appl Sci Res. 2023;13:10453–9. doi:10.48084/etasr.5501.
51. Nuha HH. Lossless text image compression using two dimensional run length encoding. Jurnal Online Informatika. 2020;4:75–8. doi:10.15575/join.v4i2.330.
52. Ali RH, Kadhim JM. Text-based steganography using huffman compression and AES encryption algorithm. Iraqi J Sci. 2021:4110–20. doi:10.24996/ijs.2021.62.11.31.

53. AbdelWahab OF, Hussein AI, Hamed HF, Kelash HM, Khalaf AA. Efficient combination of RSA cryptography, lossy, and lossless compression steganography techniques to hide data. Procedia Computer Sci. 2021;182:5–12. doi:10.1016/j.procs.2021.02.002.
54. Oo BB, Aung MT. Enhancing secure digital communication media using cryptographic steganography techniques. In: 2020 International Conference on Advanced Information Technologies (ICAIT); 2020 Nov 4–5. Yangon, Myanmar: IEEE; 2020. p. 1–6. doi:10.1109/icait51105.2020.9261790.
55. Molato AD, Calanda FB, Sison AM, Medina RP. LSB-based random embedding image steganography technique using modified collatz conjecture. In: 2022 7th International Conference on Signal and Image Processing (ICSIP); 2022 Jul 20–22; Suzhou, China: IEEE; 2022. p. 367–71. doi:10.1109/ICSIP55141.2022.9886754.
56. Awadh WA, Alasady AS, Hamoud AK. Hybrid information security system via combination of compression, cryptography, and image steganography. Int J Electr Comput Eng. 2022;12:6574–84. doi:10.11591/ijece.v12i6.pp6574-6584.
57. Majid Msallam M, Aldoghan F. Multistage encryption for text using steganography and cryptography. J Tech. 2023;5:38–43. doi:10.51173/jt.v5i1.1087.
58. Veera D, Mangrulkar R, Bhadane C, Bhowmick K, Chavan P. Modified caesar cipher and card deck shuffle rearrangement algorithm for image encryption. J Inf Telecommun. 2024;8(2):280–300. doi:10.1080/24751839.2023.2285549.
59. Wen H, Huang Y, Lin Y. High-quality color image compression-encryption using chaos and block permutation. J King Saud Univ-Comput Inf Sci. 2023;35(8):101660. doi:10.1016/j.jksuci.2023.101660.
60. Madhu D, Vasuhi S, Samydurai A. Dynamic 8-bit XOR algorithm with AES crypto algorithm for image steganography. Signal Image Video Process. 2024;18(1):429–45. doi:10.1007/s11760-024-03165-6.