

Review

A Review on Image Steganography Techniques: Evolution from Classical to Adaptive Methods

Shikha Chaudhary ¹, Gunjan Gupta ², Vikash Kumar Mishra ³, Vipin Balyan ^{2,*}
and Pramod Kumar Soni ^{4,*}

¹ Department of Information Technology, Manipal University Jaipur, Jaipur 303007, India; shikha.chaudhary@jaipur.manipal.edu

² Department of Electrical, Electronics, and Computer Engineering, Cape Peninsula University of Technology, Cape Town P.O. Box 1906, South Africa; guptag@cput.ac.za

³ School of Computer Science, University of Petroleum and Engineering Studies (UPES), Dehradun 248007, India; mail2dr.vikash@gmail.com

⁴ Department of Computer Applications, Manipal University Jaipur, Jaipur 303007, India

* Correspondence: balyanv@cput.ac.za (V.B.); pramod.soni@jaipur.manipal.edu (P.K.S.)

Abstract

Image steganography is an information-hiding technique, aiming to achieve confidentiality and data privacy while transmitting the data in a digital environment. Over the last two decades, steganography has evolved from classical spatial domain embedding to intelligent and adaptive steganographic systems capable of balancing imperceptibility, embedding capacity, robustness and security. This paper presents a review by categorizing the existing techniques into spatial domain-based, transform domain-based, hybrid and adaptive intelligent techniques. The review follows the PRISMA approach to make the selection process transparent for the inclusion and exclusion of papers in the study. Initially, the reviews include the spatial domain-based methods focusing on higher embedding capacity and simple embedding strategy, followed by transform-domain based techniques, including discrete cosine transform, discrete wavelet transform, and other multi-resolution wavelet transforms aiming to enhance robustness and imperceptibility by embedding the data into frequency coefficients. This paper further explores the methods that combine these techniques with other recent trends to develop adaptive and hybrid techniques. These techniques mainly integrate chaotic theory to enhance the security of secret data before embedding and optimization algorithms such as genetic algorithm, particle swarm optimization, Firefly, etc., for adaptive embedding to achieve an improved tradeoff. Finally, intelligent and adaptive techniques based on deep learning models such as convolutional neural networks, autoencoders, and generative adversarial networks are examined, highlighting their ability to learn intelligent embedding strategies and resist modern steganalysis. A comparative analysis is presented, including the technique, strengths, and limitations, together with the discussion of performance evaluation metrics and vulnerability analysis under image processing attacks. The review highlights the current trends and outlines the future direction to develop next-generation secure image steganographic systems.

Keywords: information security; image steganography; adaptive steganography deep learning; chaotic map; spatial domain; transform domain; industry; innovation and infrastructure



Academic Editor: Santiago Marco

Received: 24 April 2026

Revised: 11 July 2026

Accepted: 14 July 2026

Published: 5 August 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article

distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Data communication over the Internet has greatly enhanced the ease of digital communication. However, ensuring the security of data transmitted over public networks remains a significant challenge in maintaining data authentication and privacy [1]. Therefore, it raises the need for information security [2]. Information security is a discipline that deals with protecting data from malicious attacks and making it secure from intruders. It primarily consists of two branches of study: cryptography and data hiding [3] as shown in Figure 1. These branches have garnered the attention of many researchers over several decades to explore new methods to secure the data. Cryptography ensures data confidentiality by transforming the message into an unreadable form using mathematically secure algorithms; however, the presence of encrypted communication remains apparent [4]. Data hiding, or steganography, addresses a different security requirement by concealing the existence of the secret information [5]. Rather than replacing cryptography, steganography serves as a complementary approach, particularly in scenarios where covert communication is essential. Steganography is the way of concealing sensitive data inside a seemingly innocuous carrier medium without drawing the attention of undesired recipients, which distinguishes it from related methods such as cryptography and watermarking [6]. Watermarking is a technique to provide copyright protection to identify the proprietor; the watermark can be visible or invisible, depending on the application. Whereas steganography hides the presence of secret data, this characteristic protects the message. The comparison among steganography, cryptography, and watermarking is presented in Table 1 [7].

Table 1. Comparison among steganography, watermarking and cryptography.

Factors	Cryptography	Watermarking	Steganography	Hybrid Systems
Definition	Secret writing to make the information unintelligible	Embedding a message in an original signal	Concealing the information without revealing the noticeable changes	Integration of cryptography and steganography in a unified framework
Objective	To make the data unreadable	To protect the data from copyright protection	To hide the existence of data	To achieve multilayered security by protecting the data and its existence
Techniques	Public key cryptography, transposition, substitution	Blind, non-blind, fragile watermarking	Spatial and frequency domain-based techniques	Hybrid (cryptography technique + watermarking/steganography technique)
Strength	Alter the data with a key and algorithm	Embed the message within the cover	Hides the message without alteration	Hybrid method employs a layered approach
Key	Necessary	Optional	Optional	Necessary
Imperceptibility	High	High	High	High
Robustness	Yes/ High	Yes/Low–Moderate	Yes/Low–Moderate	Yes/High
Visibility	Always	May be	Never	Never
Output	Cipher text	Watermarked file	Stego file	Stego file
Type of Attack	Cryptanalysis	Statistical attacks	Steganalysis	Cryptanalysis/Steganalysis

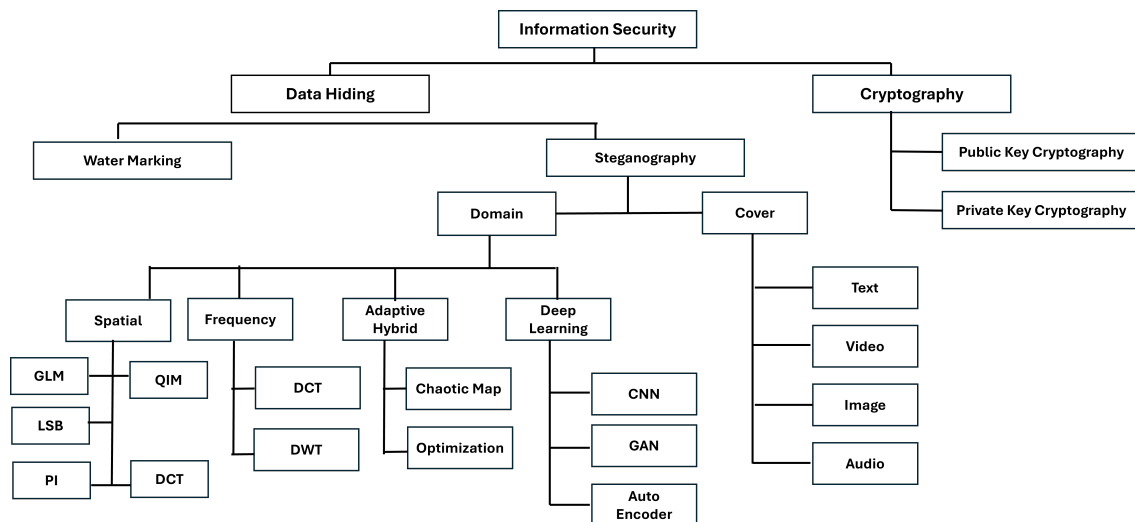


Figure 1. Hierarchical diagram of information security.

The term “steganography” is derived from the Greek words “stegano”, i.e., “hidden”, and “graphia”, i.e., “writing”. The history of steganography dates to ancient Greece, where secret messages were written on the shaved heads of slaves and then allowed to grow back, making them invisible until they were shaved again. In modern times, steganography is primarily used by the military, intelligence agencies, and healthcare for the secure communication of secret information [8]. Steganography is primarily used to achieve secrecy and confidentiality in communication. It enables the sender and the intended recipient to exchange sensitive information without drawing attention to the communication. The basic process of steganography involves (i) the selection of cover media, which can include image, text, audio, and video files; (ii) embedding the secret data involves the changing of certain properties of cover media; (iii) extraction: the recipient of the steganographic message extracts the concealed information from the carrier as shown in Figure 2. Typical steganography methods include text-based steganography, in which techniques such as invisible ink, microdots, and random and statistical generation are used to hide a message within the text of a file [9]. Network steganography is the practice of hiding data inside the protocols themselves, for example, by hiding it in header information or in fields that are left empty. In video and audio steganography, data are concealed within video and audio files, respectively; in image steganography, the secret data are hidden inside an image known as a cover image. Steganography and steganalysis form an adversarial pair of techniques that evolve together, where progress in one domain drives corresponding advancements in the other. The former targets to conceal the existence of secret information within digital images, whereas steganalysis seeks to reveal such hidden content by detecting embedding-induced anomalies.

Steganography techniques are classified into two primary domains: the spatial and frequency domains [10]. The secret message is embedded into cover image pixels in the spatial domain-based techniques; however, in the latter approach, the secret data are stored in the frequency coefficients of the cover image after being converted into the frequency domain [11], which makes it more robust. The most common frequency domain techniques are fast Fourier transform (FFT) [12], discrete wavelet transform (DWT), discrete cosine transform (DCT), and curvelet transforms [13]. A wavelet is a mathematical process for analyzing and synthesizing a signal in the time and frequency domains.

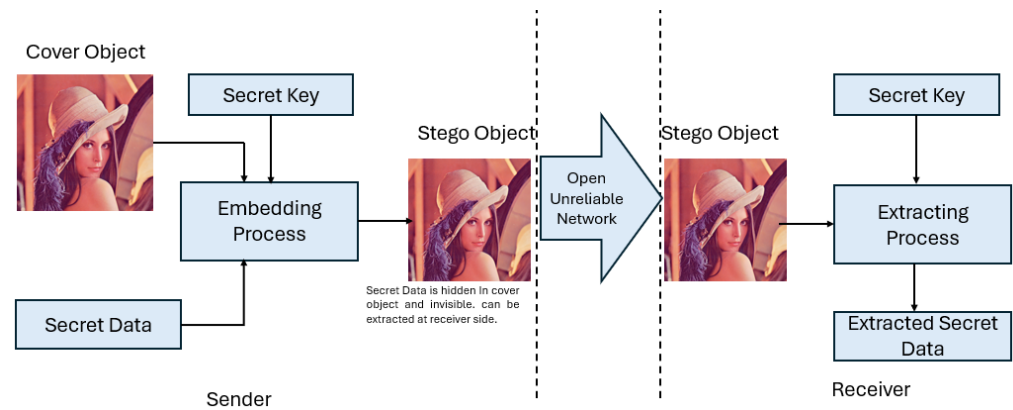


Figure 2. The basic framework of a steganography system.

Since the transmission of multimedia content via the Internet is expanding quickly, multimedia content must be protected against unauthorized use in the modern digital world. Over the past several years, researchers have presented a range of different solutions for protecting multimedia data. Steganography is one of these kinds of techniques. Many of the steganography techniques now in use offer solutions with adequate security and imperceptibility. The ability of image steganography to improve covert communication, protect multimedia content, and support academic and technical growth is what motivates this field of study. It is a vibrant field of research with enormous potential for enhancing information security and tackling the new problems of the digital era due to ongoing improvements in image steganography techniques.

While several review articles on image steganography exist, each has focused on a specific dimension of the field, like metaheuristic optimization-based approaches [14], neural network-based information hiding [15], and performance improvement strategies [16]. However, this survey provides a comprehensive synthesis that traces the methodological evolution from classical spatial domain techniques to modern adaptive approaches. Our review presents this progression as a unified narrative: we systematically analyze how early spatial-domain methods laid the foundation for transform-domain techniques, and how these, in turn, inspired adaptive strategies that respond dynamically based on characteristics of the image and field of application. The review highlights connections that are often overlooked, clarifies the strengths and limitations of each stage, and identifies emerging trends that point toward future research directions. This evolutionary perspective distinguishes this paper from prior surveys and ensures that it contributes new knowledge to the field. The main contribution of the paper is as follows:

1. To discuss the design challenges in the field of image steganography, which facilitate the exploration and design of new steganography techniques.
2. Progression from classical spatial-domain techniques to modern adaptive approaches, highlighting how each stage builds upon the previous one, helping researchers understand why the field moved toward intelligence, rather than just how a specific tool works.
3. Discussing the adaptivity achieved through three modern pillars: chaos theory for security, metaheuristic optimization for parameter tuning, and neural networks for automated feature extraction.
4. This paper explores the multiple performance evaluation criteria of existing image steganography techniques, which include aspects like imperceptibility in terms of PSNR, SSIM, and NCC; robustness, embedding capacity, execution time and computational cost. This multidimensional exploration provides a unified view of work rather than a scattered one.

2. Design Challenges in Image Steganography Techniques

The image steganography research field presents various challenges and directions for designing new steganography techniques. These challenges arise due to the rapid evolution in technology and the need for robust and secure steganography methods [17]. Researchers have a wide scope to propose a new technique that handles one or more challenges.

2.1. Analysis of Image and Embedding Domain

The representation and analysis of the image can be accomplished using the spatial and frequency domains. Subsequently, the secret data can also be embedded according to these domains. Many methods have been used to hide the data using different spatial domain methods and frequency domain methods based on various wavelets. However, there is still scope to investigate novel approaches to image representation and data embedding.

2.2. Color Models

Various color models are available to represent the color image, like RGB, YCbCr, HSV, CMY, etc. Researchers have used these to improve the performance of the technique. The image can be decomposed on the different color planes, and the embedding can also be completed on these color planes to increase the imperceptibility of the technique. The performance of the approach can be enhanced by identifying a suitable color plane to conceal data.

2.3. Tradeoff Among Imperceptibility, Robustness and Embedding Capacity

To design and analyze a steganography technique, the requirements associated are imperceptibility, embedding capacity, robustness, and complexity [18]. Imperceptibility denotes that the visual quality of the stego image should be similar to the original cover image after manipulating the pixel or coefficients. Despite minor degradation in the quality, it should not be noticeable to humans. Small changes in the image or low embedding improve the imperceptibility but make it vulnerable to attacks, while the strong embedding survives the attacks, making the technique more robust. The robustness is the ability of hidden data to remain intact after going through various attacks. So if one wants to increase the robustness, the visual quality of the image may be degraded because of a higher embedding factor or vice versa. Embedding capacity represents the amount of secret data hidden inside the cover image; a higher amount of data leads to poor quality of the stego image or vice versa. Similarly, hiding more payload, like 1 bit in every pixel, means that the data may be lost after attacks, and to survive the attack, if the data are repeated every 3 pixels, then the capacity will be less. The objective of a researcher is to develop a technique that can maximize all these parameters, which are almost reciprocal to each other. This tradeoff arises because increasing payload modifies more pixel/transform coefficients, thereby reducing imperceptibility, while robustness often requires redundancy, further constraining capacity. Figure 3 [19] presents the tradeoff between these three characteristics. However, the significance of each attribute depends upon the application. There is scope in designing an algorithm that can have a balance among all.

2.4. Security

The development of scrambling and encryption methods by ensuring the security of hidden information that works in conjunction with the image steganography technique is an ongoing challenge. The researcher can use the existing chaotic map and encryption methods to encrypt the secret data before concealing it. Therefore, there is a scope for developing a new encryption technique to hide data before embedding.

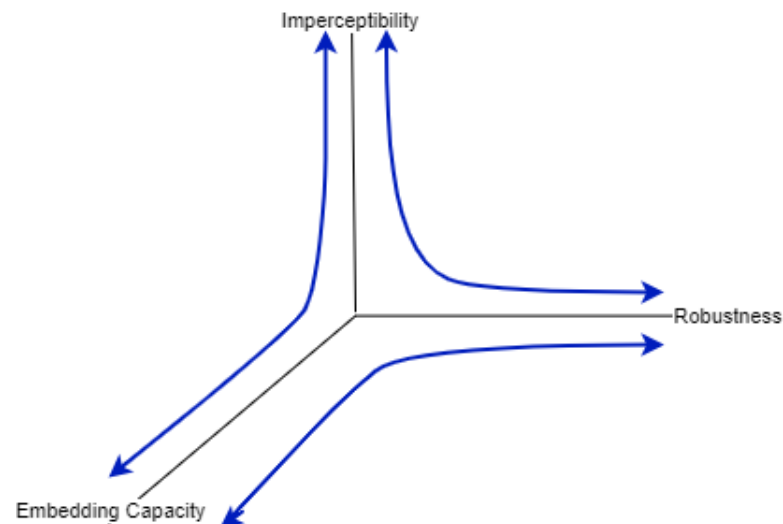


Figure 3. Imperceptibility, robustness, embedding capacity tradeoff [19].

2.5. Embedding Factor

The embedding factor is a significant determinant in the field of steganography. The efficacy of the factor has an impact on both the imperceptibility and robustness of the approach. In general, researchers have used a static value for the embedding factor that is consistent across all images. This approach opens a new way to find out the dynamic value of an embedding factor on the basis of the cover and secret image.

2.6. Robustness Against Image Processing Attacks

It is imperative that concealed information maintains its integrity despite undergoing basic image processing operations such as cropping, scaling, rotation, filtration, compression, etc. Ensuring the robustness of steganographic techniques in such scenarios is a significant challenge. From this perspective, there remains ample opportunity for advancements.

3. Evaluation Metrics

To analyze the performance of image steganography techniques, it is important to consider how much the carrier image is affected by the hidden message and how well the hidden message is concealed. Here, some important metrics like imperceptibility, robustness, and complexity are taken into account while evaluating the security and performance of image steganography techniques in this paper. These parameters are summarized as follows:

3.1. Imperceptibility

It presents the degree of undetectability by the observer. This factor assesses the level of difference or distortion caused in the cover image by concealing the secret message; the more imperceptible the hidden message, the higher the quality of the stego image. A list of parameters is discussed as follows, which measures the quality of the stego image [7].

1. **Peak Signal-to-Noise Ratio (PSNR):** PSNR is a metric to assess the alteration between the original cover image and the stego image (image generated by inserting the secret message in the cover image). It measures the degree of distortion or noise added during the process of embedding. Higher values indicate good imperceptibility, that is, higher similarity between the original and the stego image. It is calculated using mean

square error (MSE), where C_I is an $m \times n$ size original grayscale cover image and its noisy approximation is I_I (stego image) [20]; then, MSE is defined as

$$MSE = \frac{1}{mn} \sum_{p=1}^m \sum_{q=1}^n [C_I(p, q) - I_I(p, q)]^2 \quad (1)$$

Here, $C_I(p, q)$ and $I_I(p, q)$ are the pixel values of the cover image and stego image at p^{th} row and q^{th} column. For color images, the MSE can be computed by averaging the squared error across all three color channels (red, green, and blue). The total error is summed over each channel and normalized by $3mn$, which is the total number of pixels and channels. Now, PSNR in dB can be defined as

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

Here, Max_C is the maximum feasible pixel value for a grayscale image, which is 255. A higher value of PSNR and a lower value of MSE indicate a good imperceptibility.

2. Structural Similarity Index (SSIM): SSIM evaluates how structurally comparable the original image and stego image appear to be. It takes into account the contrast, luminance, and structure of the image. The higher value of SSIM indicates better perceptual similarity and better imperceptibility. The range of SSIM is $[0, 1]$, where 1 signifies the highest similarity and 0 signifies the difference between cover and stego images.

$$SSIM(C, S) = \frac{(2\mu_C\mu_S + L1)(2\sigma_{CS} + L2)}{(\mu_C^2 + \mu_S^2 + L1)(\sigma_C^2 + \sigma_S^2 + L2)} \quad (3)$$

Here, μ_C , μ_S are the mean, σ_C , σ_S are the standard deviation, σ_{CS} is the cross-covariance of the cover and stego image of size $m \times n$, and $L1$ and $L2$ are constants. $L1$ and $L2$ are used to avoid the division by zero; they are calculated using $L1 = (0.01 \times 255)^2 \approx 6.5025$, $L2 = (0.03 \times 255)^2 \approx 58.5225$ for each grayscale image. For color images, SSIM is computed independently for each RGB channel and averaged to calculate the final value.

3. Structural Content (SC): SC measures the preservation of the structural content in the steganographic image. It evaluates how well the essential visual features, edges, and textures are retained after the embedding process. An SC score of one or near one indicates better imperceptibility and preservation of important visual characteristics.

$$SC = \frac{\sum_{p=1}^m \sum_{q=1}^n C_I(p, q)^2}{\sum_{p=1}^m \sum_{q=1}^n I_I(p, q)^2} \quad (4)$$

4. Normalized Cross-Correlation (NCC): NCC evaluates the degree of resemblance between two images. The value of NCC lies in $[0-1]$. The value of NCC near to 1 show a higher correlation or similarity between two images.

$$NCC = \frac{\sum_{p=1}^m \sum_{q=1}^n C_I(p, q) * I_I(p, q)}{\sum_{p=1}^m \sum_{q=1}^n C_I(p, q)^2} \quad (5)$$

5. Average Difference: It evaluates the average difference in pixel values of the cover and generated stego image after concealing the secret message. The average difference (AD) evaluated, having a value of zero or less, shows the good quality of the image.

$$AD = \frac{1}{mn} \sum_{p=1}^m \sum_{q=1}^n C_I(p, q) - I_I(p, q) \quad (6)$$

3.2. Robustness

The robustness of a steganography technique talks about how the hidden message stays intact even when the image goes through conventional image processing operations like rotation and scaling. To evaluate the robustness of steganography algorithms, image processing operations like sharpening, histogram equalization, gamma correction, speckle noise, Gaussian noise, compression, scaling, cropping, and rotation are performed. To achieve better robustness, the secret message should be embedded into a significant area of the cover image, which is not affected by common image processing operations. It can also be measured by the Bit Error Rate (BER), which is the ratio of incorrectly extracted bits to the total number of embedded bits.

$$BER = \frac{\sum_{i=1}^N (b_i \oplus \hat{b}_i)}{N} \quad (7)$$

Here, b_i is the original embedded bit, $\hat{b}_i$ is the extracted bit, and N is the total number of embedded bits.

3.3. Embedding Capacity

It refers to the number of hidden bits in each pixel. These embedding bits per pixel (bpp) should be high while keeping visual quality and other evaluation parameters in mind, and it is referred to as the payload or embedding capacity. In the actual measurement of the embedding payload, P and Q represent the height and width of the image.

$$EC = \frac{\text{Number of bits embedded}}{P \times Q} \quad (8)$$

3.4. Computational Cost

The complexity of a steganography technique primarily depends on the embedding techniques and the image representation domain. The time complexity of a steganography algorithm also depends on the application where it has been applied. For real-time applications, the complexity of the algorithm must be as low as possible so that it will take less time to insert and reveal the secret message.

3.5. Datasets Used in Image Steganography Research

To evaluate the performance of image steganography techniques, the datasets used are ImageNet [21], BossBase [22], COCO [23], and CelebA [24]. Among these, BOSSBase (Break Our Steganographic System) is one of the most widely used datasets for evaluating spatial-domain steganography and steganalysis algorithms. BOSSBase 1.0.1 comprises 10,000 grayscale images at a resolution of 512×512 pixels of uncompressed portable gray map images of nature and objects as shown in Figure 4, which are captured with different cameras and imaging conditions. Due to its high image quality and diversity, it serves as a reliable benchmark for performance evaluation. ImageNet is a large-scale image dataset containing more than 1 million images of 1000 categories to evaluate the model used for computer vision for object recognition. It has its tiny dataset, which is publicly available, containing the size of 64×64 images of 200 categories. The COCO dataset is also widely used in computer vision for object detection and segmentation. It contains 330,000 images of 80+ categories such as people, animals, vehicles, and everyday items. CelebA is a large-scale, high-resolution facial image dataset of 10,000+ unique celebrity images and is widely used in face recognition and generative modeling. The dataset includes different expressions, poses and lighting conditions.



Figure 4. BossBase dataset sample images.

4. Literature Review

To send sensitive information to the destination, the image steganography technique must be able to securely transmit the data without affecting the quality of the stego image and embedded data. The image steganography methods can be designed and categorized with a focus on transform domain techniques, finding out the embedding position, embedding factor, encryption to provide an added layer of security, color or grayscale image, etc.

The secret data can be embedded directly by substituting the secret information at a particular location in a cover image. These are known as spatial domain techniques, like the least significant bit substitution approach, which is a traditional approach [25]. Techniques within the transform domain, including fast Fourier transform, discrete wavelet transform, discrete cosine transform, integer wavelet transform, and others are applied to conceal data in the transform coefficient of the cover image.

The detailed survey of the existing image steganography methods has been divided and performed as per the following criteria:

1. Steganography techniques based on embedding method: Spatial [5] and transform [11] are the two embedding domains in image steganography. The first one uses pixels for hiding the information, whereas the latter one uses different wavelet transforms to hide the secret data.
2. Steganography techniques based on adaptive and hybrid methods:
 - (a) Scrambling method: Before hiding the secret information inside a cover image, an additional layer of security is provided by encrypting or scrambling the data to be hidden [26]. In the review, steganography techniques based on different scrambling techniques and chaotic map-based techniques are explored.
 - (b) Selection of embedding factor (alpha): The embedding factor has significance as it extracts the precise quantity of secret information that needs to be concealed in the cover image. Later, it reflects the stego and extracted secret image's quality. The optimization algorithms can be used to select the most suitable value for the embedding factor. In the review, the different nature-inspired optimization algorithm-based steganography techniques are reviewed.
 - (c) Embedding Position: Optimization methods to find the best position to hide data according to the cover image.

To conduct the survey based on the research questions, the research papers are identified from different data sources, specifically the "Scopus" using an initial screening for the

term “image steganography”, yielding thousands of records. These are then screened based on the title and year of publication from the last 10 years. The number of records after the initial screening is more than 1000. Screening of the records containing duplicate removal is also on the basis of their title and publication year. For further screening, the query is applied to the keywords of the paper containing the words chaotic map, DCT, DWT, spatial domain, adaptive, optimization, deep learning, CNN, GAN and autoencoder with AND and OR operators. After applying this, 331 records were identified that were relevant to the field of research. To check the eligibility of the identified and screened papers based on their abstract, the total number of identified records now remained at 211. To exclude the papers further, the exclusion criteria included (i) lack of a standard evaluation metric like PSNR, MSE or EC; (ii) studies that are purely review papers; (iii) studies focusing mainly on watermarking, cryptography, or steganalysis; (iv) redundant publication; and (v) non-availability of the full text of the paper. After performing the full assessment, 100 records were included in the study. The review process is represented using the flowchart shown in Figure 5.

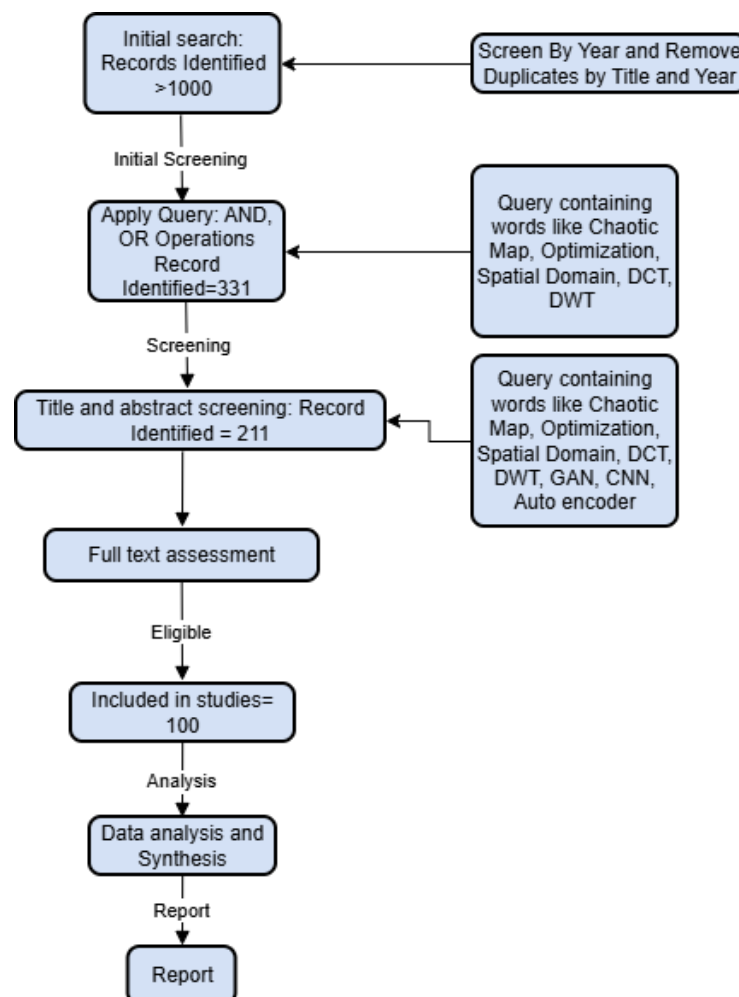


Figure 5. Flowchart representing the systematic review process using PRISMA approach.

4.1. Spatial Domain-Based Steganography Techniques

Spatial domain steganography presents the earliest class of data hiding techniques, where embedding is performed directly on pixel intensity values. These methods prioritize simplicity and high embedding capacity but often suffer from limited robustness against

image processing attacks. The mathematical equation of spatial domain-based image steganography is represented as follows:

$$I_I(p, q) = C_I(p, q) + \alpha \times S_I(p, q) \quad (9)$$

Here, I_I is the stego image, C_I is the cover image, S_I is a secret message, and α is an embedding or strengthening factor which controls the modification of the cover image. The small value of α indicates the weak embedding by modifying a smaller number of pixels, which means high imperceptibility and low robustness. In contrast, a higher value of α , such as ≥ 0.5 , represents strong embedding, which leads to low imperceptibility and is more robust against attacks.

4.1.1. Least Significant Bit Substitution Method

The most common method of spatial domain steganography is to insert the secret message in the least significant bits of the cover media, like an image or audio. This technique is known as LSB (least significant bit) steganography, which functions by substituting the lower bits of the cover file with the bits of the secret message [27], as shown in Figure 6. The primary goal of earlier LSB steganography was to build a system that could carry more payload by altering pixels in the cover image [28]. The LSB approach assumes that changing a small number of pixel values will not result in observable differences. For an 8-bit grayscale image, each pixel value can be represented as:

$$C_I(p, q) = b_7b_6b_5b_4b_3b_2b_1b_0$$

b_0 is the least significant bit, and in LSB, this is substituted by the secret message bit m as

$$I_I(p, q) = (C_I \& 11111110) | m$$

This introduces minimal changes to the cover image pixel value, which makes the modification highly imperceptible as per the human visual system.

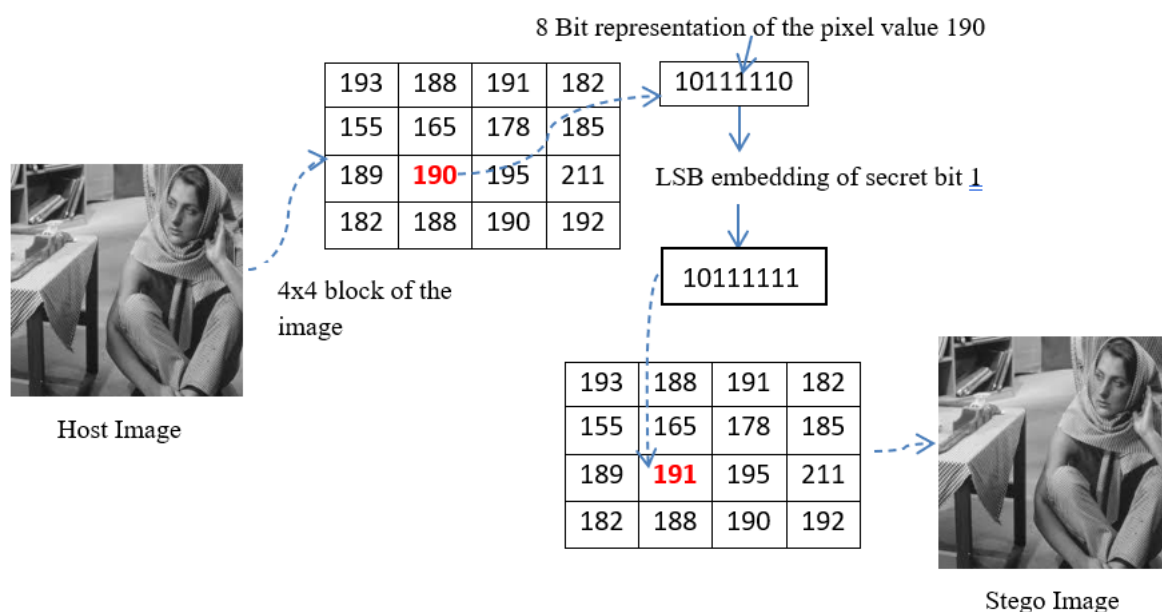


Figure 6. Basic block diagram of LSB-based steganography technique.

The advancements in LSB-based steganography focus on adaptive embedding strategies rather than uniform bit replacement. The LSB-based technique significantly enhances the imperceptibility of the stego-image with minimal additional computing cost by employing an optimal pixel adjustment process (OPAP) on the image received after applying the

traditional LSB substitution approach [29]. In [30], confidential information is encrypted and hidden among four non-overlapping, equal-sized sections of the cover image using random walks. In [31], the cover image is divided into two non-overlapping pixel blocks. The parity-bit pixel value difference and enhanced rightmost digit replacement methods are chosen based on the pixel value difference in each block. This lowers the possibility of regular/singular (RS) attacks. In [32], the payload is divided into multiple matrices that are placed into the cover image using the X-OR operation rather than using encryption. The secret image splits up into a frequency, error, and sign matrix to find the payload. These matrices are concealed in various cover images by the use of the X-OR operation in the least computation time. Gray-level modification (GLM) is a technique that is closely related to LSB [33]. Unlike basic LSB substitution, which alters the least significant bits, GLM modifies the actual gray value of pixels.

LSB-based steganography offers high embedding capacity and low visual distortion, but less security for the embedded payload. To address this limitation, recent advancements increasingly integrate cryptographic encryption techniques with LSB-based embedding, creating dual-layer security frameworks that protect both the existence and the content of the hidden data. A steganography method is needed that can fuse cryptography to ensure that sensitive data remain hidden even if the existence of a covert channel is compromised [34]. A stego key-directed adaptive LSB replacement with several encryption levels as a secure embedding approach is developed [35]. Before embedding, sensitive data are encrypted with a multi-level encryption algorithm, while the stego key is encrypted with a simpler technique. Additionally, the encrypted data are embedded using an adaptive LSB embedding method that takes into account the red channel, the encryption algorithm, and the secret data. In [36], a public-key crypto-system based on error-correcting codes is used to encrypt text using the Goppa code. The cipher-text will then be incorporated into a cover image by applying the LSB approach to protect data from attackers. A hybrid encryption algorithm using the blowfish and AES algorithm is developed to encrypt the secret image before embedding [37]. An adaptive technique is presented that chooses the best pattern in order to reduce the error ratio before embedding the data [38]. The adaptive technique, which is based on the two-bit and least significant bit (LSB) pattern in the cover image, can maximize the efficiency of the inverted LSB substitution approach. Before embedding, the message and container image bits are evaluated, and the error ratio is determined using the inverted LSB replacement technique for a variety of potential patterns. The message is embedded using the pattern with the lowest error rate.

In conclusion, LSB-based steganography has evolved significantly from simple bit substitution to an adaptive data-hiding mechanism. Through the integration of adaptive embedding techniques, encryption methods, and optimization techniques, modern LSB-based approaches effectively address traditional limitations related to security, imperceptibility, and detectability.

4.1.2. Pixel Value Differencing Method

Pixel value differencing (PVD) is a technique used in the spatial domain to hide secret data within the image by analyzing the difference between the values of neighboring pixels [39]. The modification in the high-contrast area allows more data to be inserted in such regions without noticeable distortion. In this approach, the cover image is segmented into a non-overlapping pair of pixels, and the absolute difference between each pair decides how many bits of secret data can be embedded. Larger differences permit the embedding of more bits, while smaller differences allow fewer bits to preserve image quality. The modified pixel values are adjusted to maintain the new difference while minimizing visual distortion after embedding. PVD is popular due to its ability to attain a good balance between

imperceptibility and embedding capacity [40]. In summary, PVD-based steganography has emerged as an effective and adaptive data-hiding approach by exploiting local pixel variations to balance the embedding capacity and visual imperceptibility.

4.1.3. Exploiting Modification Direction Method

Exploiting modification direction (EMD) is a spatial domain steganography approach that embeds a secret message in an image by modifying pixel values in a controlled and minimal manner, thereby reducing visual distortion [41,42]. Unlike simple LSB techniques, EMD uses modulo operations and the concept of modification direction to embed data in a group of pixels (typically a group of $n+1$ pixels to embed n -ary digits). By restricting pixel changes to minimal and predefined directions, EMD achieves improved imperceptibility.

The EMD-based embedding process uses some of the basic steps as mentioned below:

Pixel Grouping: The cover image is partitioned into non-overlapping groups G of n pixels, which are denoted as $G = (x_1, x_2, \dots, x_n)$.

Secret Digit Conversion: The secret binary data are converted into a sequence of digits S in the $(2n + 1)$ -ary numeral system.

Extraction Function: An extraction function f is defined to map each pixel group to a unique value:

$$f(x_1, x_2, \dots, x_n) = \left(\sum_{i=1}^n x_i \cdot i \right) \pmod{2n + 1} \quad (10)$$

where x_i represents the grayscale value of the i -th pixel.

Modification Rule using Modulo: Let d be the difference between the secret digit s and the current function value:

$$d = (S - f(x_1, x_2, \dots, x_n)) \pmod{2n + 1} \quad (11)$$

The stego-group $(x'_1, x'_2, \dots, x'_n)$ is generated based on d :

- If $d = 0$, then $(x'_1, \dots, x'_n) = (x_1, \dots, x_n)$.
- If $d \leq n$, the d -th pixel is incremented: $x'_d = x_d + 1$.
- If $d > n$, the $(2n + 1 - d)$ -th pixel is decremented: $x'_{2n+1-d} = x_{2n+1-d} - 1$.

The recovery of the secret digit s is computationally efficient and blind, meaning it does not require the original cover image. The receiver calculates the extraction function for each stego-group:

$$s = f(x'_1, x'_2, \dots, x'_n) = \left(\sum_{i=1}^n x'_i \cdot i \right) \pmod{2n + 1} \quad (12)$$

The EMD method chooses a single “direction” to change the pixels for hiding the message. Because of the way the extraction function is structured, any secret digit in a $(2n + 1)$ -ary system can be represented by modifying, at most, one pixel in the group by a value of only $+1$ or -1 . This “one-pixel-change” rule is what keeps the visual quality high and maintains an exceptionally high peak signal-to-noise ratio (PSNR). Ultimately, EMD provides a sophisticated framework for balancing payload capacity and imperceptibility.

Table 2 represents the review in a summarized way; it shows that LSB-based steganography techniques are easy to implement and imperceptible to human eyes, as LSB bit alteration does not much affect the image quality and has a relatively high embedding capacity. Despite these merits, LSB-based steganography shows some limitations: it is vulnerable to certain attacks and detection methods, and excessive alterations to the LSB bits can affect the image quality. The limitations broaden the scope of the development of transform-domain steganography methods.

Table 2. Comparative study of spatial domain-based steganography methods.

Ref.	Method	Performance	Robustness	Attacks Performed	Findings/Gap
2004 [29]	LSB method using OPAP	PSNR: 51.141	Low	Visual, statistical	Good imperceptibility, high Computational complexity, and vulnerable to statistical attacks
2012 [30]	LSB method with DES encryption of secret data before embedding	Avg. PSNR: 41.419 Avg. MSE: 12.1676	High	Histogram, chi-square	Estimation of the security based on computational time
2013 [32]	Secret message is divided into three matrices to submerge in the cover image using LSB method	Computation cost: $O(n^2)$	Moderate	Statistical analysis, key sensitivity analysis	Robust but less imperceptible
2015 [34]	Arnold transform scrambles the secret message before embedding in cover using LSB substitution	PSNR: 37.2732	High	Key sensitivity Analysis	Histogram of the cover and stego reveals the presence of the hidden information to the attackers
2016 [43]	LSB method: 5th and 6th bits are chosen for hiding	PSNR: 51.1797 EC: 2.6 bpp	High	Histogram attacks	Bit differencing increases embedding capacity but higher distortion risk
2017 [31]	PVD method	PSNR: 40.66 EC: 2.0632 bpp	High	Key sensitivity analysis	Tradeoff between embedding rate and robustness remains
2017 [35]	Adaptive LSB replacement approach to embed encrypted data in the host image	PSNR: 45.0309, NCC: 0.952924, SSIM: 0.955984	High	RS steganalysis	Reasonable payload with less computational cost
2018 [40]	Combine LSB substitution, PVD, and EMD	PSNR: 41.07 EC: 2.14 bpp	Moderate	RS analysis	Combined approach increasing robustness
2019 [28]	Image is enlarged to distribute the embedded pixel	PSNR: 49.34, EC: 2.0 bpp, histogram analysis	Less	Histogram	Good imperceptible but not robust
2021 [37]	Data are encrypted before embedding using hybrid encryption	PSNR: 85.11	High	Visual and statistical attack	Good, imperceptible, and robust
2022 [38]	Adaptive LSB method: The message and container image bits are evaluated to find the error ratio and embedded at the lowest error rate.	PSNR: 52.4983, SSIM: 0.9977	High	RS analysis	Dependency on the number of bits inverted makes the imperceptibility unstable

4.1.4. Vulnerability Assessment of Spatial Domain-Based Image Steganography

As we have already discussed, spatial domain-based steganography directly manipulates the cover image pixels to hide any data, which alters first-order and higher-order statistical properties such as pixel histograms and correlation structures. This statistical disruption makes the spatial domain-based image steganography vulnerable to steganalysis. The attacker may attack the stego image and detect these statistical changes by analyzing the pixel intensity $P_{Cover}(p, q) \neq P_{Stego}(p, q)$ using histogram analysis, RS analysis, and chi-square attack.

The histogram represents the probability distribution of pixel intensities. Ideally, the probability distribution of the cover and stego should be similar for secure embedding,

but the embedding schemes used in the LSB approach make it detectable. Similarly, the spatial embedding also affects the correlation between the neighboring pixels by embedding the data directly. This noise breaks the strong correlation between the pixels, enabling detection through advanced steganalysis techniques.

The vulnerability of the spatial domain, specifically the LSB-based image steganography technique, was evaluated after applying different image processing attacks. The bit error rate (BER) measures the proportion of incorrectly extracted bits from the stego image after the attack. A higher BER indicates that the embedded secret data have been significantly altered, reflecting the weakness of the steganography method against that specific attack. From Figure 7 and Table 3, it can be analyzed that when Gaussian noise was introduced with the variance of 0.001 and 0.005, the BER reached 0.5076 and 0.5146, respectively, indicating that more than half of the hidden bits were corrupted during the extraction process. Gaussian noise randomly alters pixel intensities across the image, and since LSB-steganography stores information in the least significant bits of pixel values, even minor pixel variations can flip these bits. As a result, the recovered message becomes highly distorted, demonstrating that the LSB technique is extremely sensitive to Gaussian noise. In contrast, the BER after applying salt and pepper BER was only 0.005, and the attack contains the noise density 0.005 and 0.01. This very low error rate suggests that the embedded data remained largely intact. Salt and pepper noise typically affects a relatively small number of pixels by setting them to extreme intensity values. Because only a limited portion of pixels is modified, the majority of the embedded LSB bits remain unchanged, allowing the hidden information to be recovered with minimal error. However, the BER value after applying median filtering is 0.4182 and 0.4806 with the $[3 \times 3]$ and $[5 \times 5]$ shapes of kernels, which indicates that a large portion of the bits have been altered. Median filtering works by replacing each pixel with the median value of its neighboring pixels in order to remove noise. A similar trend was observed in JPEG compression attacks, where the BER increased to 0.4986 and 0.4892 on the quality factors 50 and 60, respectively. JPEG compression operates in the frequency domain and employs quantization, which alters pixel values during reconstruction.

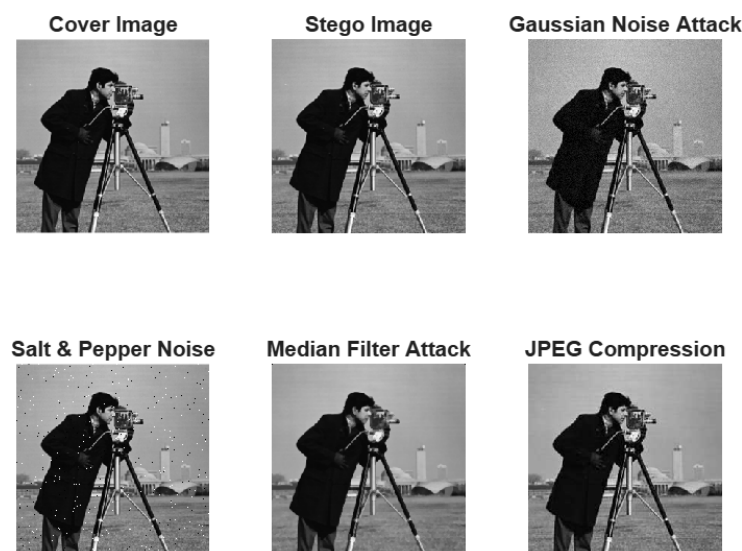


Figure 7. Vulnerability analysis of LSB steganography after applying major image processing attacks.

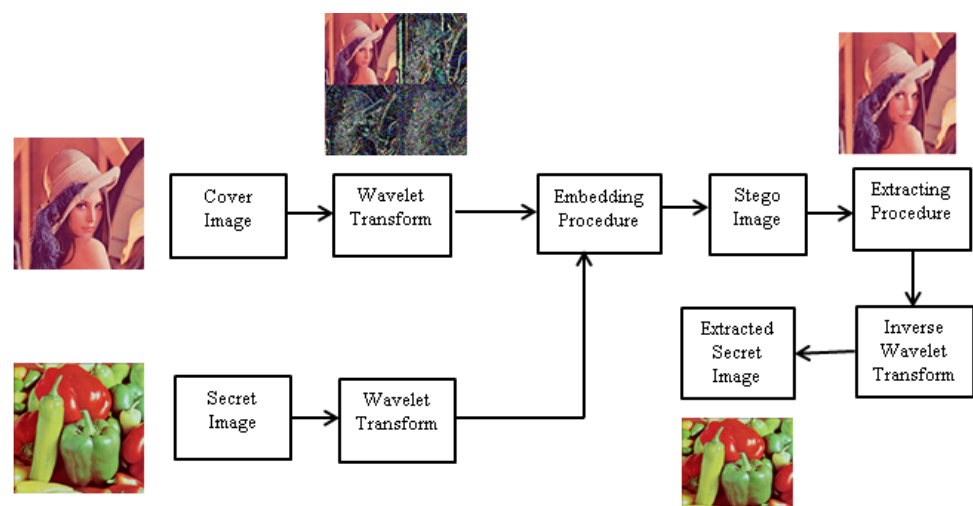
Table 3. Vulnerability analysis of LSB-based steganography.

Metric	Parameter	Value
Gaussian Noise	Variance = 0.001	0.507600
	Variance = 0.005	0.514600
Salt and Pepper Noise	Noise Density = 0.005	0.002600
	Noise Density = 0.01	0.003000
Median Filter	Kernel Shape = $[3 \times 3]$	0.418200
	Kernel Shape = $[5 \times 5]$	0.480600
JPEG Compression	Quality Factor = 50	0.498600
	Quality Factor = 60	0.489200

Overall, the results indicate that LSB-based image steganography is not robust to several common image-processing operations. While it shows some resistance to low-density salt and pepper noise, it becomes highly vulnerable under Gaussian noise, median filtering, and JPEG compression. These findings emphasize the limitations of basic spatial-domain embedding methods and suggest the need for more robust approaches, such as transform-domain or hybrid steganographic techniques.

4.2. Transform Domain-Based Steganography Techniques

While spatial domain steganography offers advantages such as high imperceptibility, high payload capacity, and low computational complexity, it also suffers from several limitations that can be detected using modern steganalysis techniques. To address the limitations, researchers shifted to transform domain-based techniques. Transform domain-based image steganography is an approach where data are concealed in an image by modifying its transform coefficients. There are different transform domains like fast Fourier transform (FFT), integer wavelet transform (IWT), discrete cosine transform (DCT), and discrete wavelet transform (DWT). The highly efficient and popular transform domain-based steganography techniques, DCT and DWT, have been reviewed in this paper. The basic model of frequency domain-based steganography is presented in Figure 8.

**Figure 8.** Basic block diagram of frequency domain-based steganography technique.

4.2.1. Discrete Cosine Transform (DCT)-Based Steganography Techniques

Discrete cosine transform presents an image as a sum of cosine functions oscillating at varied frequencies. The low-frequency components correspond to the smooth and uniform regions of the image, while high-frequency components show edges and fine details. DCT-

based image steganography modifies the DCT coefficients to conceal the secret information. The image is divided into its corresponding low, middle, and high frequency bands. Here, low-frequency bands provide the most important visual information, while high-frequency bands carries subtle details. The mid-level frequency component is generally chosen to hide the data, as it does not show a noticeable difference in pixels.

To embed the data, these frequency bands are used by applying different embedding strategies. In [44], the middle-frequency portion of the quantized DCT coefficients is employed for concealing the secret message. The secret image is encrypted using a secret key before embedding. The entropy coding scheme is applied later to obtain the compressed quantized DCT blocks. In [45], the modified quantization table is used to obtain the quantized DCT coefficients of the DCT-transformed cover image. To conceal the secret information, each DCT block contains some sets, and the value of each set contains some sequence of zeros, which can be utilized to conceal the secret information by keeping a minimum value of zeros to two. The method is resistant to visual and statistical attacks.

In [46], the method estimates the number of bits that can be concealed inside every DCT component using a capacity table. The JPEG quantization table and the human visual system (HVS) are applied to calculate the capacity table. The LSB approach is applied to hide messages in quantized DCT coefficients of the host image. The method has an embedding capacity of almost 20% of the compressed image size with negligible effects on image quality. Table 4 represents the review carried out with the findings of the study using the DCT wavelet.

DCT is a reversible mathematical transform and does not introduce information loss. However, when used in JPEG compression, the subsequent quantization of DCT coefficients introduces information loss, making the technique more vulnerable to attacks and limiting payload capacity. Researchers have developed various adaptive or hybrid techniques by combining DCT with other modern techniques to enhance robustness and the embedding capacity. These techniques are reviewed in Section 4.3. It also widens the scope of exploring other transforms to develop new steganography techniques to overcome these limitations.

Table 4. Comparative study of DCT-based steganography techniques.

Ref.	Algorithmic Approach	Performance Metric	Robustness	Attacks Performed	Findings/Gap
2002 [44]	Quantized DCT coefficients of the cover image are used for data hiding	PSNR: 34.84	Moderate	Key sensitivity analysis; JPEG compression	Modified JPEG quantization tables for embedding; acceptable image quality; better capacity; not secure against modern attacks
2004 [46]	JPEG quantization table and HVS are used to design the capacity table to embed data	PSNR: 40.84, EC: 20%	Low (robust against JPEG attack)	JPEG compression	Good visual quality for JPEG images but not robust against all image processing attacks
2007 [45]	Data are hidden in 8×8 DCT blocks in medium frequency components having two or more successive zeros	PSNR: 40.49, EC: 0.24 bpp	High	Visual and statistical attack	Provides reversibility with high embedding capacity

4.2.2. DWT-Based Steganography Techniques

Discrete wavelet transform divides a signal or image into a collection of wavelet coefficients, each representing a particular band of frequencies. The signal is decomposed by running it through a series of low-pass and high-pass filters and then downsampling the results. The discrete wavelet transform of an image divides the image into four frequency bands, HH, HL, LH, LL bands, and allows for analysis at multiple resolutions. The DWT

possesses the perfect reconstruction property, which enables a precise reconstruction of the original image from the wavelet coefficients. The wavelet coefficients are upsampled, filtered, and then summarized to produce the reconstruction.

DWT transforms host and secret images into four sub-bands, where LL and HL sub-bands of the host and secret are further split into 4×4 blocks in [47]. These blocks are used to conceal secret data by calculating the root mean square error between the host and secret image blocks. The block with the minimum error is selected to hide the secret message. The method is a blind steganography approach; it does not require an original host image at the receiver side to extract the hidden data. In [48], there is a steganography approach where the DWT is carried out on the cover and secret image, and each image is subdivided into 4×4 blocks. The difference of each image block is computed to find the optimal position to hide the secret data by means of three secret keys. In [49], the cover image is converted into a frequency domain using the 2D-Daubechies DWT transform of the R plane of the host image, and its transform coefficients are changed to conceal the secret image. The robustness of the approach is investigated on various image processing attacks. In [50], the hybrid edge detector identifies sharp places in the image where data can be concealed without being easily spotted by the human visual system. When compared to previous steganographic techniques, this methodology achieves a greater PSNR for the same amount of bits inserted at each pixel. In [51], for the cover and secret image, SVD is applied on the LL sub-band of the secret image and three stego images are generated by blending U, S, and V matrices with the cover image separately. In [52], a Daubechies DWT-based image steganography is presented where db8 is used on the cover and secret image, which is followed by an Arnold transform on the secret image for scrambling; thereafter, post-processing is performed using the genetic algorithm on the stego image to obtain a better visual appearance. In [53], secret data are hidden inside the SVD of the R, G, and B color planes of the LL sub-band of the cover image. The singular value matrix of the host and secret image is blended to generate the stego image. The orthogonal matrices are used as a key to regenerate the image at the receiver end. Table 5 represents the review carried out with the findings of the study using discrete wavelet transform.

Frequency domain-based steganography has some advantages, such as better robustness, higher embedding capacity, and enhanced security. However, it comes with some costs, including computational cost, difficulty of implementation, and vulnerability to attacks through alterations in specific frequency components. These limitations widen the scope of developing new steganography techniques using some adaptive and hybrid approaches.

Table 5. DWT-based image steganography techniques.

Ref.	Algorithmic Approach	Performance Metrics	Robustness	Attacks Performed	Findings/Gap
2008 [47]	Discrete wavelet transform is used to embed secret data into the cover image	PSNR: 53.04	Moderate	JPEG compression, cropping, Gaussian blur, median filtering	Cover image with similar characteristics is selected to improve imperceptibility
2012 [49]	2D Daubechies-DWT is applied on the R-plane of the cover image for data hiding	PSNR: 74.169	Moderate	Sharpening, Gaussian noise, salt and pepper noise, blurring	Optimization enhances robustness without compromising visual quality

Table 5. Cont.

Ref.	Algorithmic Approach	Performance Metrics	Robustness	Attacks Performed	Findings/Gap
2012 [50]	Hybrid edge detector-based embedding with adaptive payload estimation	PSNR: 46.88; EC: 0.93 bpp	NA	NA	Edge-guided embedding achieves high payload with minimal distortion but ignores inter-pixel relationships
2018 [48]	DWT is applied to both cover and secret images, subdivided into fixed 4×4 blocks for optimal embedding	PSNR: 44.84; SSIM: 0.9646	High	Sharpening, histogram equalization, gamma correction, Gaussian noise, rotation	Fixed-size sub-block division increases computational cost and may produce isolated artifacts
2020 [51]	DWT is applied on the SVD of the LL band of the secret image, generating three stego images by blending U, S, and V matrices	PSNR: 74.2345	High	JPEG compression, noise	Combined DWT-SVD framework enhances robustness and security significantly
2020 [52]	Daubechies DWT with Arnold transform is used for encryption, while a genetic algorithm performs denoising	PSNR: 43.82; NCC: 0.9903; Histogram analysis	Moderate	Filtering, noise	Smoothing operation improves imperceptibility and resistance to attacks
2022 [53]	DWT followed by SVD on cover and secret images; singular matrices are blended using alpha blending on colored image	PSNR: 63.70; SSIM: 0.9853; EC: 0.402 bpp ET: 1.83 s	High	JPEG compression, scaling, cropping	Fractal-based cover image improves robustness and ensures secure embedding

4.2.3. Other Transform-Domain Based Image Steganography Techniques

Integer wavelet transform (IWT) maps integer-valued image pixels to integer-valued wavelet coefficients using the lifting scheme, thereby eliminating the rounding errors associated with the conventional discrete wavelet transform (DWT). It preserves the exact pixel values after inverse transformation, making it suitable for reversible image processing applications. The author in [54] embeds secret data in the high-frequency edge coefficients, and it classifies the coefficients according to the position of the MSB. In [55], the sub-bands are classified into edge and non-edge coefficients after the IWT transform of the cover image. The edge coefficients are chosen to conceal the secret data along with a genetic algorithm for finding the position. SVM is used first to separate the region of interest (ROI) from the non-region of interest (NROI) in the medical image. Then, IWT is applied to embed secret information within the NROI part of the medical image [56].

The contourlet transform (CT) is a multiresolution and multidirectional transform that represents an image to capture the smooth contours and geometric structures efficiently [57]. It employs a Laplacian pyramid (LP) for multiscale decomposition and a directional filter bank (DFB) for directional decomposition, enabling it to represent edges and contours with high directional sensitivity and anisotropy. This combination offers superior image representation, making the contourlet transform particularly suitable for applications such as watermarking and steganography. In [58], a high contrast image is chosen for embedding the secret data from the dataset. The sub-band after the contourlet transform of the cover image is divided into 8×8 blocks, and the mean energy and block energy are computed. The block that has maximum energy is chosen to embed the secret data. Three different image steganographic methods using contourlet transform are developed using SVD, QR factorization, and non-negative matrix factorization to embed secret data in the respective

coefficients [59]. Each secret image is encrypted using the geometric mean, and data are concealed in the contourlet coefficients [60].

Curvelet transform (CvT) is a multiscale, multidirectional, and anisotropic image representation method that overcomes the limitations of wavelet transform in representing curved edges and contours [61]. It represents image features using basis functions with varying scales, orientations, and locations, following the parabolic scaling law. This enables a sparse and accurate representation of smooth curves and edges while preserving directional information. Its excellent edge-preserving ability, high directional selectivity, and efficient sparse representation make it suitable for image steganography. In [62], secret data are scrambled using Arnold transform and embedded in non-uniform regions of high-frequency curvelet coefficients. The cover image is selected using fuzzy logic based on image complexity and characteristics. In [63], high-detail regions are selected as cover images using predefined thresholds. Secret bits are embedded into each selected block by modifying coefficient values using an embedding strength parameter. The comparison among the techniques is presented in Table 6.

Table 6. Other transform-domain-based image steganography techniques.

Ref.	Algorithmic Approach	Performance Metrics	Robustness	Attacks Performed	Findings/Gap
2017 [54]	IWT-based embedding on MSB of edge coefficients	PSNR: 57.9 (avg.)	-	-	Does not evaluate robustness against common attacks, limited evaluation metrics
2023 [55]	IWT with genetic algorithm for embedding position selection	PSNR: 56.41 SSIM: 0.9986	Moderate	RS attack	Single-objective function (PSNR) is chosen that may not produce the best tradeoff among imperceptibility, robustness, and security
2023 [56]	IWT with SVM	PSNR: 64 SSIM: 0.9600 EC: 0.44 bpp	Moderate	Salt and pepper noise, cropping, filtering, Gaussian noise, JPEG compression, histogram equalization, and rotation	High computational complexity and may underperform on large datasets
2013 [58]	CT of high-contrast image for data embedding	PSNR: 45.02 SSIM: 0.9500	-	-	Algorithm is not evaluated against stego attacks
2019 [59]	CT with SVD, QR, NMF	PSNR: 57 NCC:1	Moderate	Noise, JPEG compression, histogram equalization, and rotation	Good imperceptibility, moderate execution and extraction time
2026 [60]	CT with geometric mean for encryption	PSNR: 45.5 SSIM: 0.9837 NCC: 0.9951	Moderate	Salt and pepper noise, cropping, filtering, Gaussian noise, JPEG compression, histogram equalization, and rotation	No adaptive embedding strategy, a fixed embedding strength ($\alpha = 0.15$) is used for all images regardless of image characteristics
2017 [62]	CvT with fuzzy-based cover image selection	PSNR: 52.09 NCC: 0.9999 SSIM: 0.9998	Modearte	Noise, cropping, histogram equalization, JPEG compression and steganalysis attacks	High computational cost

Table 6. *Cont.*

Ref.	Algorithmic Approach	Performance Metrics	Robustness	Attacks Performed	Findings/Gap
2018 [63]	CvT with cover image selection using threshold	PSNR: 42.89	Moderate	JPEG compression, filtering	Fixed embedding length, limited image processing attacks to evaluate robustness and limited evaluation metrics to evaluate the overall performance

4.2.4. Vulnerability Assessment

To analyze the vulnerability of transform domain-based image steganography, we implemented the basic two-level DWT decomposition on the grayscale cover image “Camera-man” using the Daubechies (db4) wavelet, and the LH2 sub-band is chosen for embedding the secret image “Airplane”. The experiment was conducted on MATLAB R2025b.

The analysis from the Table 7 reveals significant sensitivity to common image processing attacks, as reflected by the obtained BER values. Under a Gaussian noise attack with the variance values 0.001 and 0.005, the method performs poorly because Gaussian noise alters almost every pixel in the image, which in turn disturbs the DWT coefficients. Since the embedding depends on precise coefficient values, even small changes can change the embedded bits, leading to a high error rate. In comparison, the method performs better on salt and pepper noise with noise density of 0.005 and 0.01, where noise affects only a small percentage of pixels, yielding moderate robustness. For median filtering on $[3 \times 3]$ and $[5 \times 5]$ kernel windows, the method again shows vulnerability because filtering operations smooth the image and modify neighboring pixels, thereby altering the DWT coefficients in which the data are embedded and increasing extraction errors. Similarly, JPEG compression attack on quality factors 50 and 60 introduces loss through quantization by particularly affecting high and mid-frequency components. Since the embedding is performed in the LH2 sub-band, which lies in the mid-frequency range, the compression process distorts these coefficients and leads to a loss of hidden information. The effect of the image processing attacks is presented in Figure 9. Overall, the results indicate that while the method demonstrates moderate resilience to sparse noise, it is highly vulnerable to continuous noise, filtering, and compression. This vulnerability arises due to the sensitivity of transform-domain coefficients, the use of single-coefficient embedding, and the absence of redundancy or error correction mechanisms, highlighting the need for more robust embedding strategies.

Table 7. Vulnerability analysis of DWT-based steganography.

Metric	Parameter	BER
Gaussian Noise	Variance = 0.001	0.403320
	Variance = 0.005	0.502241
Salt and Pepper Noise	Noise Density = 0.005	0.144531
	Noise Density = 0.01	0.188477
Median Filter	Kernel Shape = $[3 \times 3]$	0.285645
	Kernel Shape = $[5 \times 5]$	0.490723
JPEG Compression	Quality Factor = 50	0.243652
	Quality Factor = 60	0.128906

4.3. Adaptive and Hybrid Steganography Techniques

The traditional steganography techniques have their own strengths and weaknesses. When they are used alone, they have some limitations, such as low robustness, low embedding capacity, and vulnerability to attacks. To mitigate these challenges and to strengthen the performance, two or more approaches can be combined to develop a hybrid approach. The adaptive methods adjust the embedding technique to hide the data based on cover media and also adjust the payload on the basis of the requirement and available capacity.

4.3.1. Chaotic Map-Based Steganography Techniques

In order to provide an additional level of security to a hidden secret image, cryptography has been used with steganography to develop a hybrid approach. Many steganography techniques have been developed by applying encryption techniques like RSA, the Blow-fish algorithm, and AES to encode the hidden data. These techniques work well for text encryption but are ineffective for image encryption due to the repetitive pixel information in the image. This problem is fixed by chaos-based encryption methods, which generate uniformly dispersed random keys that encode the image [64]. A chaotic map is a mathematical function that generates a sequence that is sensitive to the initial condition and shows pseudo-random but deterministic behavior. This behavior encourages researchers to apply a chaotic map in image scrambling and key generation to provide additional security [65]. This section presents the review of existing chaotic image steganography, its findings, and its limitations.

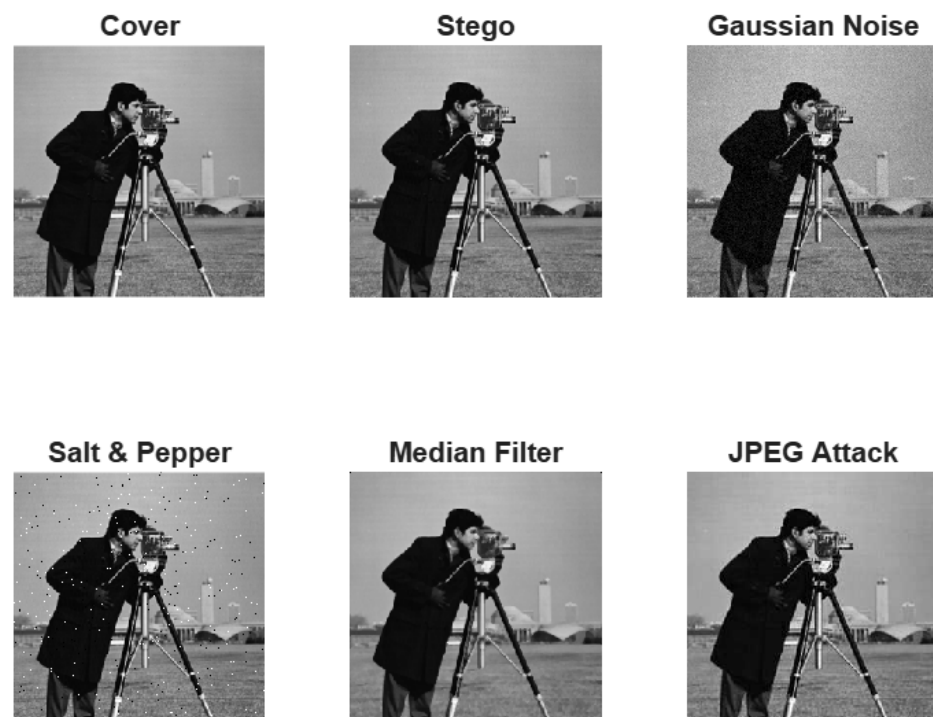


Figure 9. Vulnerability analysis of DWT-based image steganography after applying major image processing attacks.

Chaotic Maps

The chaotic maps are categorized based on their characteristics and dimensions. This review focuses on the chaotic map-based steganography techniques. The well-known chaotic maps which are applied to scramble the data or to generate the key sequence are discussed as follows:

(1) 1D Logistic Map: The one-dimensional discrete map was proposed by Robert in 1976 [66]. The formula to calculate or map the value to a new value is as follows:

$$x_{i+1} = \mu x_i (1 - x_i) \quad (13)$$

Here, parameter μ controls the behavior of the system. The value ranges in $1 \leq \mu < 3$ converge to a stable fixed point, $3 \leq \mu < 3.57$ shows period doubling $(2, 4, 8, \dots)$, $3.57 \leq \mu \leq 4$ shows chaotic behavior.

(2) Hénon Map: In this 2D chaotic map proposed by Hénon in 1964 [67], the x, y plane is mapped using the following equation:

$$\begin{aligned} x_{i+1} &= 1 - p \cdot x_i - y_i \\ y_{i+1} &= q \cdot x_i \end{aligned} \quad (14)$$

Here, parameters p and q control the behavior of the map; the map shows chaotic behavior for $p = 1.4$ and $q = 0.3$. The system behaves unpredictably and is complex with a small change in the value of the initial conditions.

(3) Lorenz Map: This 3-D chaotic map developed by Edward Lorenz in 1963 [68] is presented as follows:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= x(\rho - z) - y \\ \frac{dz}{dt} &= xy - \beta z \end{aligned} \quad (15)$$

Here, the system behaves chaotically for the value of $\sigma = 10, \rho = 28, \beta = 8/3$. Due to its large key space, it is widely used in image encryption.

A chaos-based steganography algorithm that employs Canny's edge detection to locate the edge pixels of the host image and conceal data within the chosen pixels [69]. Canny's approach is noise-resistant and can detect actual weak edges. A cat map is applied to scramble the hidden image before embedding. The data are embedded in edge regions using a matrix-encoding technique and an LSB-matching technique. Data are scrambled using the chaotic map, and the data are concealed in the cover image by the pixel pair-matching technique [70]. In [71], the host image is subdivided into low and high-frequency regions; the LSB of the secret image is inserted in the lower region and the MSB in the higher-frequency regions of the host image. The pixel positions in higher and lower frequency regions are determined using a chaotic map. The modified logistic map is implemented to identify the pixel position to insert data in the cover image [72]. The DCT-transformed host image and the high-frequency part are scanned in a zig-zag way, and the pixel position for data embedding is calculated using the piecewise chaotic map [73]. The message is encrypted using RSA, and a Duffing chaotic map is being applied to generate the chaotic sequence of the image to be embedded inside the DCT coefficients of the host image [74]. A 3D chaotic map-based image encryption technique is suggested to shuffle pixels of the secret image using a secret key before concealing [75]. A hybrid chaotic map encryption technique, a composition of the Duffing and Hénon maps, is designed to conceal secret data in the graph wavelet transformed cover image to enhance robustness and security [76]. The use of the graph wavelet and the hybrid map increased the time complexity of the technique. A two-step encryption method is designed to encrypt the secret message. In the first step, the original text is scrambled using an Artificial Neural Network (ANN) structure. The second step further secures the scrambled text by using a combination of a tent-logistic hybrid map and advanced XOR operations for encryption. To strengthen the randomness

and unpredictability of where the encrypted data are embedded in the image, Langton's ant cellular automaton is used to generate the coordinates for embedding [77]. Table 8 represents the review carried out with the findings of the study using chaotic maps.

Table 8. Comparative study of adaptive and hybrid steganography techniques using chaotic maps.

Ref.	Method	Performance	Robustness	Attacks Performed	Findings/Gap
2013 [69]	Cat map for image scrambling of secret image and LSB method for data hiding	PSNR: 71.94, EC: 2.0 bpp, Exe.time: 1.44 s.	Moderate	steganalysis attacks	Data hiding capacity is reduced due to edge-based embedding as selective pixels are chosen for substitution; high computational cost.
2013 [70]	Chaotic map for secret image encryption and pixel pair matching for embedding	PSNR: 61, Histogram Analysis	Moderate	Noise, JPEG Compression	Efficient embedding, secure under mild noise.
2014 [71]	LSB of secret data is embedded in the lower region and MSB in higher frequency regions of the cover image. Three chaotic map sequences are used to find the pixel position in higher and lower frequency regions	PSNR: 55.41	Moderate	Filtering	Vulnerable under distorted images
2017 [72]	Modified 3D logistic map is proposed for pixel selection	PSNR: 54.387, SSIM: 0.9984	High	Compression, Filtering	Secret key can be used to make the technique more secure, security against major attack has not been investigated
2017 [73]	Chaotic embedding in DCT coefficient	PSNR: 36.3762 Correlation: 0.8582 Entropy: 7.6840	High	Statistical Attacks	Low imperceptibility for some images
2017 [74]	Two algorithms are proposed, using RSA and Duffing chaotic map to encrypt the message before embedding into DCT coefficients	PSNR: 21.5606, Time: 3.012 s	Moderate	Compression	Parallel implementation enhances security but reduces imperceptibility
2020 [75]	Pixel shuffling and 3D chaotic map to scramble the secret data	Entropy: 7.2636, NCC: 0.9700, Time: 0.01316 s	High	Key Sensitivity	Imperceptible and secure against brute force
2022 [76]	Hybrid chaotic map using Henon and Duffing maps to scramble the secret image and graph wavelet transform to hide data	PSNR: 44.82, NCC: 0.9902	High	Compression, Rotation, Scaling, Filtering, Noise	Imperceptability and robustness is increased with high computational complexity
2025 [77]	Two levels of security using ANN and a hybrid chaotic map are applied	PSNR: 52.07, SSIM: 0.9994, NPCR: 99.97, UACI: 32.2094	High	Steganalysis	Enhances imperceptibility and robustness but may increase the time complexity

Security Implications and Cryptanalytic Challenges of Low-Dimensional Chaotic Maps

However, 1D chaotic maps are simple and computationally efficient, but they are vulnerable to attacks [78]. Low-dimensional chaotic maps suffer from structural flaws that make them unsuitable for applying directly to high-security steganography without modification. The flaws include the following:

(1) Small Key Space: Due to limited floating-point precision (typically 64-bit double precision), as if the floating-point number is 10^{-18} , the actual number of key spaces will be finite, i.e., 2^{42} . Modern cryptographic standards require a key space of at least 2^{128} to resist brute-force attacks. The 1D maps fall significantly short, allowing attackers to exhaustively search the parameter space in hours.

(2) Chosen Plain-text Attack: The other issue with a 1D map is that it is vulnerable to a chosen plain-text attack, as there is a very simple mapping of the data to a new location [79]. If an image is scrambled using the 1D map, it would be easy for an attacker to detect the mapping to unscramble the data. Maps like the Logistic map ($x_{n+1} = \mu x_n(1 - x_n)$) only exhibit “full” chaos for very specific values of μ (e.g., $\mu \approx 4.0$).

(3) Non-Uniform Chaotic range: The non-uniform chaotic sequence creates the clumping effect, where most of the time, numbers will be generated within a specific range, which does not allow the pixels to spread unevenly. For an attacker, it is easy to detect the scrambled data at a particular range of pixels through visual analysis and histogram analysis. A secure approach has a flat histogram rather than a vulnerable one, which indicates that every pixel value (0–255) appears with nearly the same frequency.

(4) Dynamic Degradation: Mathematically, the chaotic map generates a highly random sequence with infinite precision. While implementing digitally with finite computational precision, its dynamic properties are affected due to the different computational precision of the hardware device and rounding errors. This degradation will hamper the security of chaos-based steganographic systems due to the reduction in randomness and sensitivity of the generated sequence, which makes embedding locations more predictable.

To overcome the challenges of 1D maps, hyper-chaotic maps are used for image encryption due to their larger key space and greater sensitivity to initial conditions, as shown in Table 9.

Table 9. Comparative analysis of chaotic maps in steganography.

Map Type	Examples	Key Space	Computational Complexity	Dynamic Degradation Susceptibility	Security Vulnerabilities
1D Chaotic	Logistic, Tent, Sine Maps	Low (Effective space $< 2^{100}$)	Low	High	Susceptible to key space reconstruction, sensitive to brute force attack
2D Chaotic	Hénon, Arnold Cat	Medium ($> 2^{128}$)	Low–Moderate	Moderate-High	Better than 1D but finite-precision degradation, and periodicity may expose embedding patterns
3D Chaotic	Lorenz, Chen	Medium ($> 2^{180}$)	High	Moderate	Stronger resistance to brute-force and statistical attacks
4D/Hyper-chaotic	4D Rössler, Chen Systems	High ($> 2^{256}$)	Very High	Low	Highly resistant to differential cryptanalysis and AI-based prediction

The suitability of low-dimensional chaotic maps for cryptographic applications is also analyzed using phase space trajectory analysis and Lyapunov exponents. The phase space trajectory analysis illustrates how system states evolve over time and whether the trajectories densely cover the phase space, which is an essential property in encryption techniques to ensure high diffusion and confusion. The Lyapunov exponent (LE) presents a quantitative measure of the system’s sensitivity to initial conditions. The Lyapunov exponent of a discrete chaotic map $x_{n+1} = f(x_n)$ is as follows:

$$\lambda = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \ln |f'(x_i)| \quad (16)$$

A positive value of λ shows the chaotic behavior of the system, and the larger value of λ shows the system is more chaotic. It also exhibits faster divergence, which enhances resistance against statistical and cryptanalytic attacks. If a system has more than one positive value of Lyapunov exponent, then the system becomes hyperchaotic. Conversely, low-dimensional chaotic maps often suffer from limited phase space coverage and weaker Lyapunov exponent spectrum, leading to reduced randomness and potential dynamical degradation in digital implementations.

The Lyapunov exponent (LE) spectrum of the logistic map, as shown in Figure 10, shows how the system behaves as the value of μ increases. For the value $\mu < 3.57$, the system is stable, whereas for the greater value of μ , the Lyapunov exponent becomes positive and shows the chaotic behavior of the system. It also represents a window where the LE becomes negative, which represents the non-uniform behavior of the map and makes it inappropriate for cryptographic applications. The LE of the tent map is more uniform and positive for the value of $\mu \geq 1$, which shows it is more suitable for cryptographic applications as compared to 1D maps.

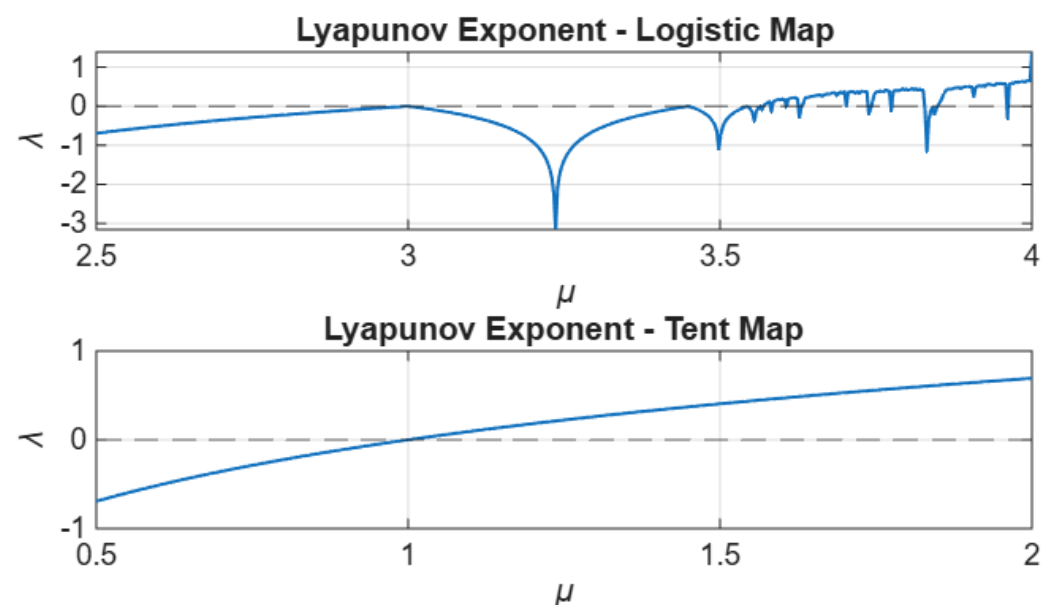


Figure 10. Lyapunov exponent spectrum of logistic chaotic map and tent map.

Chaotic map-based image steganography has some merits, such as better robustness, better security due to its sensitivity to initial conditions, and the uniform distribution of hidden information inside a cover image. In addition to these merits, it also has some limitations, such as low payload capacity, high complexity for multidimensional chaotic maps, a dynamic degradation of digital chaos and computational overhead. To overcome these limitations, researchers have come up with new solutions to design adaptive methods by combining/applying optimization algorithms and neural networks for adaptive embedding.

4.3.2. Optimization-Based Steganography Techniques

The performance of steganography techniques also depends on the pixel position selection and embedding factor. To locate the optimal position for data hiding and the appropriate value of the embedding factor, various optimization techniques are explored in the literature employing various types of optimization techniques. The classification of the optimization algorithms is presented in Figure 11. The evolutionary optimization is inspired by the principles of natural selection and genetic evolution, such as selection, crossover, and mutation, to iteratively evolve a population of candidate solutions toward an optimal result. The swarm intelligence optimization mimics the collective behavior of social

organisms such as birds, or ants, where individual agents share information and cooperate to converge on optimal solutions within a search space. The physics-based optimization is inspired by the laws of physics, such as thermodynamics or gravity, to guide the search process toward optimal solutions. However, the swarm-based optimization in the field of image steganography research dominates evolutionary and physics-based algorithms due to the larger search space, faster convergence, and stronger exploration and exploitation. In this section, a review of some of these optimization-based steganography techniques is presented.

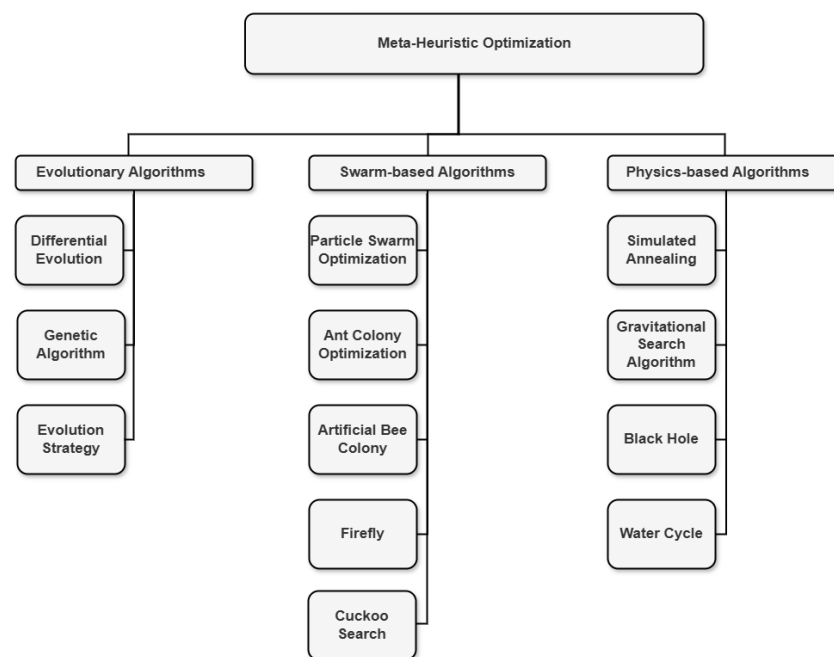


Figure 11. Classification of optimization techniques.

Evolutionary Algorithm Based

Evolutionary optimization-based image steganography has emerged as an effective research direction. As the embedding process can be formulated as a complex optimization problem, evolutionary algorithms provide an efficient mechanism for discovering near-optimal embedding solutions. The algorithm is inspired by the principles of biological evolution. It iteratively evolves candidate solutions through operations such as selection, crossover, and mutation, enabling an adaptive optimization of embedding locations, coefficient selection, and embedding parameters. In a modified genetic algorithm-based steganography approach, the initial population is produced by a heuristic approach and fitness function using eight-connected PSNR based on neighborhood pixels in 3×3 block [80]. Genetic algorithm (GA) and particle swarm optimization (PSO) are applied to find the optimal coefficients for hiding the secret “Quick Response” (QR) coded data [81]. This method enhances the PSNR and provides good embedding capacity.

Swarm Intelligence Based

Swarm intelligence is inspired by the collective behavior of natural swarms, such as bird flocking, fish schooling, and bee foraging. These algorithms employ population-based search strategies to identify optimal embedding locations. Unlike conventional steganographic techniques that rely on fixed embedding rules, swarm intelligence algorithms dynamically adapt to the characteristics of the cover image, resulting in balancing the imperceptibility, embedding capacity and robustness. The particle swarm optimization-based technique can select the pixels to insert the secret audio signal in an image [82].

The pixel selection is based on a fitness function, which is a cost function. For evaluating fitness, the cost function evaluates the pixel's edge, entropy, and intensity. In [83], PSO is employed for identifying pixels to conceal a message in the spatial domain to attain a high embedding capacity. The PSO is applied to select an appropriate pixel in the cover image for the hiding of secret bits. The MSBs of the secret image are extracted to hide. Initially, the cover image is scanned using a raster scan; these pixels are scanned from the first row to the last row and left to right. Later, the PSO method is employed to obtain the starting point for the scanning and scanning order in each block to find out the best position to hide the data. One steganography approach embeds hidden messages in images using particle swarm optimization (PSO) and integer wavelet transform (IWT) [84]. It seeks to minimize differences between modified coefficient values and original values, assuring excellent security and accurate retrieval of the secret message. The method was evaluated against other PSO-based techniques and popular steganography techniques, and it surpassed them in terms of imperceptibility, payload capacity, statistical analysis, security, and robustness against image processing attacks. In [85], an adaptive method is presented to conceal secret information in low-frequency coefficient bands of the DWT-transformed cover image. The algorithm determines the ideal location to conceal the secret message. The message is compressed with the help of the "Huffman encoding" method before embedding into the low-frequency coefficients.

Other swarm optimization techniques are explored to design efficient steganography techniques. In [86], cuckoo search (CS) is applied to identify the optimal substitution matrix to embed the confidential message. The steps include encrypting the confidential message, preprocessing the host image, embedding the message with a CS algorithm, and creating an embedded image are all steps in the procedure. The CS method assists in determining the best substitution matrix for transforming the secret messages. The approach exceeds other existing methods with respect to image quality, security level, and embedding capability. A bat algorithm-based steganography scheme has also been proposed [87]. The initial pixel position is found using the logistic map and later, the bat algorithm is employed to discover the optimal position to embed the secret data into the cover. A steganography technique using an improved fruit fly optimization (FOIS) algorithm, which identifies the optimal locations of pixels of the host image, is presented [88]. The host image is broken into 8×8 blocks, and the number of blocks for subsequent processing is determined by a permutation combination. The secret information is encoded in every block by optimal pixel selection and the discrete cosine transform coefficients' LSB. To transmit the data securely, a secret communication channel is developed by using the suggested steganography algorithm and the Rivest Cipher (RC4) crypto-system. A coyote algorithm is applied for key selection, and a new design technique to find the pixel to hide the secret data is applied [89]. The enhanced crow algorithm is used to find the optimal embedding position along with LSB to hide the secret data, which enhances the stego image quality and decreases the computational cost compared to other similar techniques [90]. The comparative analysis of the reviewed techniques is shown in Table 10.

From the study, it is observed that adaptive techniques maximize embedding capacity without sacrificing the quality of the media, increase robustness, and improve security. However, the complexity of implementation poses significant challenges. Hybrid techniques build complex layers of deception that greatly increase the difficulty of detection and extraction and achieve higher data embedding capacity. However, it increases computational cost and complexity. These challenges broaden the scope of the development of new steganography techniques.

Table 10. Comparative study of adaptive and hybrid steganography using optimization techniques.

Ref.	Optimization Strategy	Method	Performance	Robustness	Attacks Performed	Findings /Gap
2015 [80]	Evolutionary	Modified genetic algorithm with heuristic method for initialization of population	PSNR: 38.466808, Histogram Analysis	NA	NA	Fitness function may be improved for better PSNR.
2017 [81]	Hybrid (Evolutionary and Swarm Intelligence)	QR coded secret data are concealed using GA and PSO algorithms	PSNR: 52.56, EC: 2.39 bpp	High	Statistical attacks	High computational cost.
2017 [82]	Swarm Intelligence	Particle swarm optimization for pixel selection to hide audio message in an image	PSNR: 47.6	High	RS steganalysis attacks	Extracted message contains noise.
2019 [83]	Swarm Intelligence	Pixel selection using PSO by partitioning the cover image into blocks	PSNR: 56.60, EC: 4.0 bpp	Moderate	Gaussian, Salt and pepper noise	Imperceptible and secure
2020 [86]	Swarm Intelligence	Cuckoo algorithm to identify best substitution matrix to embed a secret message	PSNR: 37.92, EC: 1.125 bpp	High	Cryptanalysis, noise attack	Dual-layer protection
2020 [87]	Hybrid (Evolutionary and Swarm Intelligence)	GA and bat algorithm for pixel selection and logistic map for initial population	PSNR: 47.540	Moderate	Steganalysis	Only PSNR is taken for analyzing the technique
2020 [88]	Swarm Intelligence	Improved fruit fly to determine pixel position for embedding	PSNR: 63.4216, SSIM: 0.99718, Processing Time: 11.52 s	High	Compression, noise	Hybrid optimization enhances robustness
2020 [84]	Swarm Intelligence	PSO to find optimal pixel position for embedding and optimal pixel adjustment process for better visual quality	PSNR: 41.15, SSIM: 0.9977	High	RS attack, chi-square attack, Image processing attacks	Higher computation time than existing similar approaches
2021 [85]	Swarm Intelligence	Secret data are encoded using Huffman encoding and concealed in low-frequency bands of cover image using PSO	PSNR: 78.65, SSIM: 0.999, BER: 0.035	High	Compression	Better PSNR for the smaller size of secret text data, not evaluated the performance on varying sizes of secret images
2022 [91]	Swarm Intelligence	ACO along with chaotic sequence to find the best pixel position	PSNR: 83.4021, SSIM: 0.9985 and histogram analysis	Moderate	Filtering, steganalysis	High computational overhead and slow and premature convergence
2022 [92]	Swarm Intelligence	Pixel selection using CSO algorithm and double chaotic map-based encryption	PSNR: 56.66, SSIM: 0.9984, QI: 0.9990, CT: 0.8445	High	Image processing attacks	High computational time

Table 10. Cont.

Ref.	Optimization Strategy	Method	Performance	Robustness	Attacks Performed	Findings /Gap
2023 [93]	Swarm Intelligence	3D chaotic map and quantum walk in PSO to find out pixel position and secret bits selection	PSNR: 44.10 SSIM: 0.9885	High	Security analysis on quantum walk	High computational cost
2023 [94]	Evolutionary	Optimal pixel position using GA	PSNR: 37 EC: 1,200,010 BER: 0 Complexity: $O(N^2)$	High	RS and histogram attack	Better EC with zero bit error ratio by maintaining promising quality and security
2023 [95]	Swarm Intelligence	Bald eagle search algorithm for optimal pixel selection and chaotic encryption to encrypt the message	PSNR: 55.5, SSIM: 0.9984, MSE: 0.1827, NCC: 0.9992	NA	NA	Attacks are not employed to test the security of the algorithm
2023 [96]	Swarm Intelligence	Whale optimization algorithm is applied on blocks to find the scanning order	PSNR: 62.44, SSIM: 0.9942 EC: 4.44 bpp	Moderate	RS and histogram analysis	Image compression may affect the quality extracted secret image
2024 [89]	Swarm Intelligence	Chaotic coyote optimization for key selection and location selection	NPCI: 99.5 UACI: 33.2	Moderate	Statistical Attacks	Enhances the robustness and imperceptibility with less computational time
2024 [90]	Swarm Intelligence	Crow algorithm is improved to find optimal pixel, and LSB is applied to conceal information	PSNR: 76.25 dB	Moderate	Salt and paper noise and regular and singular analysis	Enhances imperceptibility and robustness

4.4. Deep Learning-Based Image Steganography Techniques

With significant advancements in deep learning (DL), image steganography has transformed from simple handcrafted embedding methods to highly adaptive, intelligent, and secure data-hiding techniques. DL enables automatic feature extraction, dynamic embedding strategies, and robustness against modern steganalysis. This review includes the methods based on DL architectures (CNNs, GANs, Autoencoders, RNNs, and Transformers); discusses their performance in terms of payload capacity, imperceptibility, and security; and identifies research trends, challenges, and future directions.

4.4.1. CNN-Based Image Steganography Techniques

Convolutional neural network-based image steganography using StegNet learns end-to-end mapping between the cover image and stego image as well as between the secret image and the extracted secret image [97]. The StegNet network, with a combination of hiding and extraction networks, enables steganography and the extraction of full-size images to boost steganographic capacity. A new steganography method inserts a large amount of data in the cover image through four security layers using an adaptive neural network [98] with high execution time. Experimental results are considered on many color images to check the largest amount of payload capacity and the concealing percentage. It

conceals a large amount of data, reaching 12 bpp (bit per pixel) with the use of non-uniform adaptive image segmentation, which allows data to be embedded preferably in textured and high-variance regions where visual distortion is less noticeable. The embedding rate is adaptively controlled based on neighboring pixel variance, preventing excessive modification in smooth regions. Additionally, the suggested method minimizes pixel value differences between the cover and stego images, preserving structural consistency. For a high-quality image such as “Baboon” as the cover image of 512×512 size, the capacity of bits for each channel (R, G, B) is greater than one million and naturally supports larger payloads due to the strong local color variations that provide perceptual masking. As a result, even at high embedding rates, severe structural degradation is effectively avoided. A steganography technique where the cryptography approach is convoluted with a deep neural network can provide additional security [99]. In [100], a deep network called HiDDeN is presented in which both the encoder and decoder are trained to embed the message and to extract the message, respectively. Using this approach, a sender does not need to send the original cover image to the receiver. In another steganography technique used to increase the anti-detection characteristic of the resultant image, the secret image is transformed using DCT and encrypted using elliptic curve cryptography [101].

4.4.2. GAN-Based Image Steganography Techniques

A generative adversarial network (GAN) is a modern DL technique used to embed the data using adversarial training. It has two neural networks: a generator to embed the data and to generate the stego image and a discriminator to discriminate between the stego and original cover image, as shown in Figure 12. This adaptive embedding technique has drawn the attention of many researchers in the drive to develop new steganography approaches. In another steganography approach, the secret image is not hidden in the cover media; instead, it is mapped with the noise vector [102]. It dispenses the secret message bits throughout the cover image. A chaotic map to enhance the security of the secret data and a GAN network to hide the message has also been designed to enhance the imperceptibility of the recovered image [103].

A well-generalized generative adversarial network (GAN) should maintain high embedding security and extraction accuracy when applied to “unseen” cover images that differ significantly from the training data. It should not simply “memorize” the specific images it is trained on but should be able to hide and extract data from any image. In a GAN, the extractor will not be able to extract the secret data if the test data are different in brightness and contrast from the training image data, which reduces the robustness of the system. Furthermore, the training images may undergo diverse transformations and originate from heterogeneous sources, which can lead to variations in the statistical features and result in degraded imperceptibility in the case of unseen data.

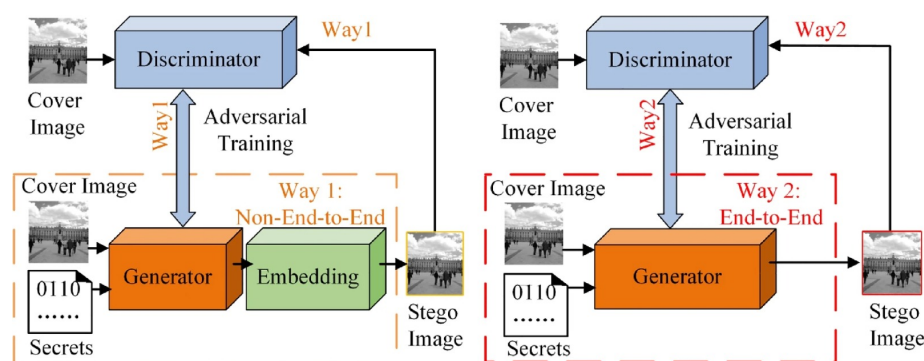


Figure 12. Generative adversarial network (GAN)-based image steganography [104].

4.4.3. Autoencoder-Based Image Steganography Techniques

The encoder–decoder design has had a lot of influence on image steganography using CNN models. The encoder takes the cover image and the secret image as input to generate the stego image, and the decoder extracts the embedded secret image from the stego image, the basic model is shown in Figure 13. While the core concept stays unchanged, researchers have explored a variety of architectural designs [105]. Each encoding algorithm combines the cover image and secret image differently; therefore, the convolutional layer and pooling layer are expected to vary accordingly. Depending on the technique, the parameters may also vary, such as the number of filters, the strides taken, the filter size, the activation function, and the loss function. The size of the cover image and its proportions are essential considerations. 3D-CNN Autoencoder has a square rearrangement to handle sub-aperture sequence data for compression. The encoder downsamples the angular resolution, and the decoder reconstructs the original plenoptic image [106]. Instead of concatenating raw images, feature extraction and data fusion are used for efficient hiding in [107]. A CNN-based autoencoder can be used for hiding and revealing secret images. Encoder–decoder networks have three stages: Preparation (feature extraction), Hiding (embedding secret image), and Reveal (decoding container image) [108]. LSB, DCT, and autoencoders with encoder–decoder latent features have been combined, where LSB ensures imperceptibility and DCT adds robustness to the CNN-based encoder–decoder [109]. StegFormer is a Transformer-enhanced autoencoder designed for high-capacity multi-image hiding, embedding multiple secret images in a single cover. It combines a U-Net architecture with Transformer blocks and skip connections, capturing all the features of the image [110]. A new steganography using Huffman encoding and an autoencoder is developed to enhance the imperceptibility and robustness [111]. The comparative analysis among exiting and studied deep learning-based image steganography is presented in Table 11.

Table 11. DL-based image steganography techniques.

Ref.	Method	Performance	Robustness	Attacks Performed	Findings/Gap
2013 [98]	Hybrid adaptive neural networks with genetic algorithms	BMP Images	EC: 12 bpp	High	Conceals large amounts of data with better adaptability.
2018 [97]	Deep CNN-based steganography (StegNet)	ImageNet	EC: 23.57 bpp Accuracy: 98.2%	Moderate	Achieves high payload capacity with minimal distortion and robust to statistical analysis.
2018 [100]	Deep network-based hiding (HiDDeN)	COCO	PSNR: 42 dB	High	Demonstrates hiding data with minimal perceptual loss.
2018 [102]	Deep convolutional GAN	Celebrities, Food101	EC: 0.073 bpp	Moderate	Enhances image realism while embedding hidden content.
2019 [99]	Cryptography + deep neural network	Flickr30k	MSE: 8.81926	High	Combines encryption and learning-based embedding for improved secrecy
2020 [101]	Elliptic curve cryptography (ECC) + deep neural network	Benchmark Images	PSNR: 43.1383 dB, SSIM: 0.9683	High	Balance between the imperceptibility and robustness
2020 [106]	3D conv autoencoder + square rearrangement	EPFL Light Field	PSNR: 40.2 dB SSIM: 0.941	Low	Best plenoptic compression
2021 [107]	Lightweight feature extraction and fusion autoencoder	CelebA, COCO, ImageNet	PSNR: 46.8 dB SSIM: 0.961	High	High embedding capacity and efficient by preserving angular and spatial structure

Table 11. Cont.

Ref.	Method	Performance	Robustness	Attacks Performed	Findings/Gap
2022 [108]	CNN autoencoder	Own Dataset	PSNR: 44.6 dB	Moderate	Better than traditional CNN
2022 [112]	Swin Transformer with logistic-map-based encryption	ImageNet	PSNR: 45.19, SSIM: 0.992, EC: $\approx$ 48 bpp	Large model size; training uses only MSE loss	
2023 [109]	Hybrid spatial + frequency + CNN autoencoder	BossBase	PSNR: 50.3 dB SSIM: 0.9720	High	Strong imperceptibility and high robustness due to spatial and frequency domain usage; limited payload capacity, higher computational complexity
2024 [110]	Transformer + U-Net autoencoder	COCO	PSNR: 52.4 dB SSIM: 0.9840	High	Multi-image hiding with the highest payload capacity and excellent resilience. However, it has a more complex architecture and requires higher computational resources and training time
2024 [103]	Chaotic map and GAN	DIV2K, COCO, ImageNet	PSNR: 33.09 dB SSIM: 0.9100	Moderate	Imperceptibility of recovered image is better, scope in enhancing the imperceptibility and capacity of the stego image
2025 [111]	Huffman encoding+ LSB+ Autoencoder	Tiny ImageNet, COCO, CelebA	PSNR: 62.12 dB SSIM: 0.9992 Capacity: 2.31 bpp	High	Adaptive embedding by comprising Huffman compression technique and deep learning technique to improve robustness and imperceptibility
2025 [113]	Vision Transformer	ImageNet, COCO, Visual Genome	PSNR: 35.53, MS-SSIM: 0.9582, Acc: 98.99%	High	Higher computational complexity
2025 [114]	DWT and Unified Swin Transformer	ImageNet, DIV2K, COCO, and 102 Flower	PSNR: 44.0979 SSIM: 0.9884	Moderate	Fails against JPEG compression attack
2025 [115]	Diffusion model	DescGPT, LAION-10K, MS-COCO, Flickr8K	EC: 0.0625 bpp	High	Diffusion sampling improves extraction accuracy but increase computational cost

4.4.4. Other Deep Learning Architecture-Based Image Steganography Techniques

Recently, a deep-learning model named Transformer, which was designed for natural language processing, has also gained the attention of researchers in the field of image steganography. The authors of [112] encrypt the secret image using a recursive permutation based on a logistic map. The encrypted secret image and the cover image are then fed into a Swin Transformer-based hiding network to generate the stego image. A vision Transformer is used for feature extraction in the encoding network and spatial transformation network for correcting geometric distortions at the decoder network [113]. In [114], the steganography framework consists of shallow and deep feature extraction modules using DWT and Unified Swin Transformer Blocks (USTBs) for hiding and extracting the secret message.

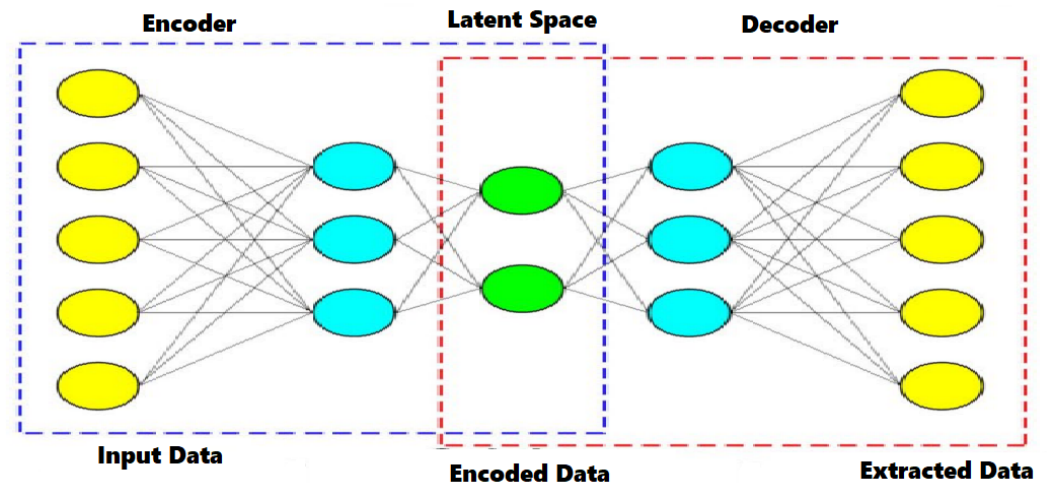


Figure 13. Autoencoder-based image steganography.

A diffusion model learns the underlying distribution of data by gradually adding Gaussian noise to training images and then learning to reverse this process through iterative denoising to reconstruct or generate realistic images [116]. In the image steganography model, the model learns the statistical distribution of natural images and embeds hidden information during the reverse denoising process [117]. In diffusion-based generative image steganography framework, the secret message is encrypted and transformed into a Gaussian-distributed latent representation using an orthogonal mapping module with random seed generation, Schmidt orthogonalization, and random shuffling [115]. This noise is then used as the initial noise input to the latent diffusion model, and iterative denoising is used to generate the stego image.

4.4.5. Comparative Analysis

The training time of a DL model refers to the duration required to learn its parameters from data, which are typically measured over multiple epochs on a large dataset. Training time is typically measured in minutes, hours, or days, depending on the hardware and model size. Inference time is the time required by the trained model to generate a prediction and is very low, on the order of milliseconds or seconds, depending on the hardware and framework [118]. The computational cost of neural network methods is segmented by model type and task complexity. Models trained on high-resolution datasets such as ImageNet and CelebA-HQ often require GPU days [119,120]. The comparative analysis is presented in Table 12. For instance, a CNN model such as ResNet-50 requires approximately 18 h for training on ImageNet. Empirical studies indicate that inference latency for CNN-based models ranges from 5 to 10 ms per image, whereas more complex generative models such as GANs and AE exhibit higher latency, often reaching 40 ms and up to 5 ms, respectively. The observation shows that deep learning models achieve high accuracy, but they impose significant computational overhead in both training and inference stages, motivating the need for lightweight and efficient architectures.

Table 12. Comparison of training time and inference latency across deep learning architectures.

Architecture	Dataset	GPU Hardware	Avg Training Time	Inference Latency	Ref.
CNN (ResNet-50)	ImageNet	NVIDIA A100, 1 to 8 GPUs	18–2 h (90 epochs)	5 to 10 ms	[119,120]
AE (CNN-based)	CelebA	NVIDIA	5–10 h	~1–5 ms	[121]

4.5. Understanding the Tradeoffs: Evolution of Steganography Methods

Imperceptibility, robustness and embedding capacity are major attributes in steganography that determine the efficacy of a steganography technique. The design of a steganography method is governed by specific requirements that vary according to its intended application. However, there is a tradeoff between the amount of data to be embedded (the size of the secret data) and the imperceptibility or visual quality of the stego image. Increasing the embedding capacity requires more modifications to the cover image, which may degrade its visual quality and compromise imperceptibility. Consequently, optimizing the tradeoff between embedding capacity and imperceptibility is a fundamental objective in steganographic design.

Traditional spatial-domain techniques such as LSB substitution, PVD, and the EMD methods discussed above have gained popularity due to their high embedding capacity and minimal computational overhead. As these methods directly manipulate pixel values, they leave statistical trails that make them highly vulnerable to steganalysis attacks (e.g., chi-square and RS analysis). Furthermore, these methods are less robust to common image processing operations such as JPEG compression, cropping, rotation, and filtering; any such modification can easily destroy the hidden data. In spatial-domain steganography, increasing the embedding capacity directly degrades the imperceptibility, as it modifies a larger volume of pixels, which distorts the statistical structure of the image. In the case of low payload, algorithms selectively hide data within highly textured or noisy regions where modifications do not affect image quality; however, a higher payload forces data to be embedded into smooth, homogeneous regions, creating visible changes such as salt and pepper noise. Even if these changes escape the human eye, they completely shatter statistical imperceptibility by disrupting the high correlation between neighboring pixels and altering pixel histograms.

To overcome these limitations, the field shifted toward transform-domain methods utilizing DCT, DWT, etc. By applying any of these transforms, an image is mapped into frequency components containing low, mid, and high frequencies. By embedding secret data into carefully selected coefficients, transform-domain methods can better preserve visual quality while improving robustness against common image processing operations such as compression, filtering, and noise. However, only a limited number of transform coefficients can be modified without introducing noticeable distortion, resulting in relatively lower embedding capacity compared with spatial-domain techniques. Arbitrarily altering too many frequency coefficients presents visible changes, destroying visual imperceptibility.

A stand-alone image steganography has a uniform embedding mechanism, whether it is in the spatial domain or the transform domain. Modern adaptive techniques explicitly try to preserve statistical properties, whereas basic DCT and DWT methods do not guarantee this. These adaptive methods employ adaptive embedding strategies, which are optimization algorithms to identify optimal embedding locations and embedding strengths. The hybrid techniques are designed by combining two or more approaches to optimally balance the security, robustness, payload capacity and visual quality. Chaotic-map-based methods were introduced to enhance security by randomizing embedding locations, which spreads the changes uniformly throughout the image while preserving the visual quality of the image. This random selection of hiding the data also increases the embedding capacity and robustness by making it difficult for image processing operations to remove all hidden information. Deep learning has further advanced this evolution by learning data-driven feature extraction and an adaptive embedding strategy directly from the dataset. All these approaches seek to balance the conflicting objectives of payload capacity, imperceptibility, and robustness, representing the current direction of image steganography research by overcoming the limitations of standalone techniques. The evolution of the steganogra-

phy strategies, their strength and limitations are summarized in Table 13. From the table, it is concluded that a single method cannot achieve or balance the tradeoff parameters, and research should be shifted to develop hybrid, adaptive and intelligent, multi-objective steganographic systems capable of balancing the required parameters.

Table 13. Evolution of image steganography techniques from conventional to adaptive frameworks.

Technique	Representation Methods	Approach	Strength	Limitation
Spatial-domain	LSB, PVD, EMD	Direct embedding into pixel values	Higher embedding capacity, low computational complexity	Poor robustness, low imperceptibility at high payloads
Transform-domain	DCT, DWT, DFT	Embed data into transform coefficients	Better imperceptibility, increased robustness	Low payload capacity and fixed embedding strategy
Chaotic map-based	Logistic Map, Arnold Cat Map, Henon Map + DWT/LSB	Pseudo-random embedding and encryption of the secret data	High security, reduced statistical detectability, and improved imperceptibility	High computational cost, sensitive to parameter selection
Optimization-based	GA, PSO, ACO, CSO + DWT/DCT/LSB	Optimizing embedding factor and location	Better payload, imperceptibility, robustness, and security	Increased computational cost, performance depends on the optimization convergence
Deep-learning based	CNN, GAN, AE	Learn adaptive embedding based on the dataset	Intelligent feature learning, highly robust, improved imperceptibility	Large training datasets and high computational cost

5. Applications and Fields of Steganography

Steganography obfuscates the information within other digital media like images, audio files, video, and text so that it is not immediately visible to the observer. Here are a few real-world applications of steganography:

1. **Secret Communication:** Steganography can be used to conceal important or classified information within seemingly other files, enabling people or organizations to communicate secretly without raising any suspicion.
2. **Copyright Protection:** Steganography can be used to insert invisible watermarks or copyright information within digital media, such as image or audio files, to provide copyright protection. This makes it possible for content creators to establish their intellectual property ownership and defend it against theft or copying.
3. **Covert Surveillance:** Steganography can be used by law enforcement authorities and intelligence organizations to implant identification or tracking information within digital content, enabling covert surveillance of suspects. This method can help in the detection and stoppage of illicit activity.
4. **Digital Forensics:** To find evidence or information that has been hidden during an inquiry. Investigators can find hidden communications or files that may be important to a case by examining files for the existence of steganographic data.
5. **Security in Data Storage:** By hiding important information within files, steganography can improve data security. For added security against unauthorized access, secret information like passwords, encryption keys, or documents might be buried inside other files.
6. **Healthcare:** Steganography can be used to protect patient data in the medical image, authenticate the image, and protect the medical device security.
7. **Digital Entertainment:** Steganography can be used to embed a hidden message in movies or music files, and it can also be used to embed virtual objects in augmented and virtual reality.

8. Cloud Security and Data Leakage Prevention: In recent years, due to the popularity of cloud-based technologies, steganography is widely used to secure sensitive data during storage and transmission, and it reduces the risk of data leakage.
9. Blockchain and Digital Ledger Systems: Nowadays, steganography is used to embed transaction metadata inside digital assets stored on blockchain systems, which helps in improving data integrity and hiding sensitive transaction details. Recent studies have integrated blockchain with steganographic frameworks. The technique uses blockchain to securely transmit stego images containing hash data of the patients' records across a decentralized network of hospitals, where each image is associated with a cryptographic hash stored as a transaction [122]. The Ozyavas Takaoglu Ajlouni method [123], integrated with blockchain technology, has been developed, where the generated index-based secret data are securely stored and shared across a decentralized network.
10. Bio-Steganography: Steganography where the DNA sequences are used for hiding secret data by transforming it into biological information or encoding it into DNA sequences. DNA-based encoding techniques have emerged in this direction, where biological sequences are used for high-capacity and secure data embedding [124,125].

6. Steganalysis

It is a technique for detecting, altering, or destroying the message hidden within any cover medium [126]. The objective of a steganography approach fails when the existence of hidden information is detected using the steganalysis attacks. It may happen due to a change in image characteristics in terms of the visual quality of a generated stego image after embedding the hidden data. Steganalysis attacks are divided into two main categories: active steganalysis and passive steganalysis. The active steganalysis attacks extract or alter the hidden secret data, whereas in passive steganalysis attacks, one only finds out the existence of hidden data to uncover the steganography process. In this section, we are discussing those attacks that were taken into consideration by researchers in the literature.

6.1. Statistical Attacks

A statistical attack is a passive attack performed on an image using some mathematical formulas to find the existence of hidden data. Hidden information is generally embedded randomly, and that randomness is used to reveal the presence of hidden data. The presence of concealed data can be detected using a variety of statistical analysis methods, such as the chi-square test, bit plane analysis, RS analysis, and histogram analysis.

- (1) Chi-Square Test: The test is performed to check the robustness of the steganography algorithm. The probability analysis of the stego image and original image is calculated to compare the difference between the two; if it is zero, then no message is hidden. This technique mainly finds out the messages that are hidden using the LSB technique.
- (2) Histogram-Based Analysis: The histogram of the original image and the stego image is compared to see the difference in the frequency distribution of both images due to embedding. This technique is mainly useful in the analysis of images, where pixel value differencing-based techniques are applied to embed the secret.
- (3) RS Steganalysis: The RS analysis approach, which stands for regular and singular analysis, was developed to detect randomized LSB embedding techniques. The methodology allows the use of minor modifications in the LSB of pixels and uses this flipping operation and a discriminating function to categorize pixels into regular and singular groups. The frequency of these groups determines the length of the hidden message in stego images. In the literature, several steganographic approaches are examined using RS steganalysis to check their security or undetectability.

6.2. Visual Attack

A visual attack is a visualization technique, such as bit-plane analysis and difference image analysis, to examine a stego image to identify visible embedding artifacts. Bit-plane analysis examines the distribution of individual bit planes, and difference image analysis highlights pixel modifications by comparing the cover and stego images. Many spatial-domain steganography methods embed secret data by modifying LSBs. A visualization of the LSB plane can reveal abnormal patterns or structured artifacts that may indicate the presence of hidden data. Difference image analysis computes the pixel-wise absolute difference between the original cover image and the corresponding stego image. The difference highlights the pixel position where pixel modifications were introduced during the embedding process. Visual attacks are generally ineffective against modern steganography methods, which is why statistical steganalysis techniques and machine learning/deep learning-based detectors have become the predominant approaches for identifying hidden information.

Steganalysis in the context of color images consists of additional complexity due to the strong statistical dependencies among color channels. Conventional steganalysis approaches primarily relied on statistical features, including higher-order co-occurrence statistics and inter-channel correlations. With the advancement in security systems using artificial intelligence, most research has used machine learning classifiers to distinguish cover and stego images. Although these methods are effective against earlier embedding schemes, they often struggle to generalize against highly adaptive and content-aware steganographic techniques. Recent advances have seen a significant shift toward deep learning-based steganalysis, particularly convolutional neural networks tailored for color images. These models are designed to preserve subtle residual information across individual color channels and to exploit inter-channel relationships without prematurely suppressing embedding artifacts. By learning discriminative features directly from data, deep steganalysis frameworks have demonstrated superior detection performance against modern steganographic algorithms. The rapid progress in color image steganalysis has, in turn, driven the development of more secure, adaptive, and steganalysis-resistant steganographic methods, reinforcing the adversarial nature of the field.

6.3. Spatial Rich Models

Spatial rich models (SRMs) [126] are used to detect hidden information in digital images; they do not extract hidden data but rather decide whether the image is stego or not. SRM uses a two-stage pipeline in which feature extraction and classification are performed separately. SRM applies high-pass filters to the image, which suppresses the normal image content and highlights the noise residuals obtained from filtering operations. These residuals are used to model statistical dependencies via co-occurrence matrices by analyzing the relationship between neighborhood pixels. These statistical measurements are combined and converted into high-dimensional feature vectors. The extracted feature vectors are typically classified using machine learning classifiers to distinguish between cover and stego images. The performance of SRM models is assessed using probability error, detection accuracy, and false alarm rate; the formulas are shown in Section 6.4. Steganography modifies pixel values slightly and disrupts natural statistics, which SRM captures. If embedding is advanced, then it is hard to detect the existence of hidden data; it is also sensitive to common image processing operations like noise addition, compression, and scaling. SRM generates high-dimensional feature vectors, sometimes exceeding more than thousands of dimensions, resulting in increased complexity, memory overhead, and longer training times for classifiers. Due to these limitations, deep learning approaches are needed.

6.4. Modern Steganalysis: A Shift from Traditional to DL-Based

Deep learning has introduced a significant paradigm shift in steganalysis by integrating feature extraction and classification into a single, unified workflow. These modern models learn descriptive features directly from the input data [127]. A major strength of this approach is the “backward communication” it facilitates between the classification and extraction stages; this essentially creates a feedback loop that refines the model’s internal logic and significantly boosts detection performance. These architectures are highly scalable, ranging from simple shallow networks with a single perceptron layer to massive deep neural networks featuring hundreds of hidden layers. In the current landscape of steganalysis, the most prominent architectures used to achieve these results are CNN, RNN, GAN and autoencoders. The performance of steganalysis algorithms can be assessed using their prediction error and detection accuracy. To assess the accuracy, the results are classified into four categories: false negative (F_N) shows the stego images predicted as cover images, false positive (F_P) represents the class of cover images predicted as stego images, true positive (T_P) means the stego images predicted as stego, and true negative (T_N) means the cover images predicted as cover images. The prediction error P_e in Equation (19) is calculated using the false alarm rate and misdetection rate using Equations (17) and (18). The detection accuracy D_{Acc} and sensitivity are calculated using Equation (20) and Equation (21), respectively.

$$FA = \frac{F_P}{F_P + T_N} \quad (17)$$

$$MD = \frac{F_N}{F_N + T_P} \quad (18)$$

$$P_e = \frac{FA + MD}{2} \quad (19)$$

$$D_{Acc} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (20)$$

$$Sensitivity = \frac{T_P}{T_P + F_N} \quad (21)$$

7. Discussion and Key Observations

Image steganography has undergone a substantial transformation over the past decades, evolving from simple LSB substitution techniques to highly intelligent, adaptive security frameworks. In its earliest stage, spatial-domain methods dominated the field due to their simplicity, low computational cost, and high payload capacity. However, these techniques were vulnerable to statistical detection and image processing attacks and operations. As steganalysis techniques became more advanced, the limitations of fixed-pattern embedding became increasingly evident. To overcome these weaknesses, transform-domain methods were introduced, embedding secret information within frequency coefficients rather than modifying the pixel intensities. By operating in DCT- or DWT-based domains, these approaches improved robustness but did not fundamentally shift the paradigm toward intelligent security. With the advancements, these wavelet-based approaches have taken some additional steps to provide more security by applying chaotic maps to scramble the secret data before embedding or to generate the key for data encryption or chaos-based embedding. These techniques are presented as hybrid techniques. Chaos theory introduces unpredictability that strengthens resistance to statistical detection. The next major advancement came as adaptive techniques such as optimization-based steganography, where metaheuristic algorithms such as genetic algorithms, PSO, etc., were used to select embedding positions on the basis of image characteristics or to select the

optimal value of the embedding factor to minimize distortion and improve imperceptibility. The optimization techniques enhanced embedding efficiency but also increased the computational complexity. The emergence of neural network-based methods marked a more significant shift toward intelligent adaptive techniques due to the popular usage of neural networks in different domains. DL architectures, particularly CNNs and GAN models, enabled automatic feature extraction and data-driven embedding techniques. These models showed improved resistance to steganalysis, classical image processing attacks and improved imperceptibility. However, they introduced challenges such as high computational complexity and dependency on large training datasets.

Moreover, most prior surveys focused on architectural innovations without positioning these methods within a broader evolutionary aspect that connects them to earlier paradigms. Unlike prior literature reviews that focus on isolated methods, this paper has provided a unified multi-domain perspective. We have shown that the most effective modern schemes are hybrid in nature, combining the spatial-adaptive logic with the cryptographic security of chaotic maps. Rather than treating spatial, transform, optimization, chaos-based, and DL driven techniques as isolated categories, this paper presents them sequentially from traditional to modern steganographic systems. This structured progression clarifies not only how techniques differ but also why the field naturally progressed toward adaptive frameworks in response to increasingly powerful detection mechanisms. A central contribution of this review lies in identifying three dominant pillars of modern steganography: chaos-based randomness, optimization-driven embedding, and intelligent DL models. By integrating these pillars under a unified conceptual framework, the review bridges theoretical foundations. The proposed unified adaptive framework reflects the shift from traditional embedding to context-aware embedding, from less security to multi-layered security, and from isolated methods to hybrid integration of techniques. Overall, the dominance of the present review lies in its balanced theoretical and practical perspective, its evolutionary synthesis across domains, and its articulation of a unified adaptive paradigm. By systematically connecting classical foundations with contemporary intelligent systems, this review not only consolidates existing knowledge but also provides a structured roadmap for next-generation steganography research. To synthesize the findings of this review, Table 14 provides a technical comparison between the proposed multi-domain framework and existing single-domain surveys.

Table 14. Comparison of prior reviews and this paper.

Title	Focus Area	Distinguish Feature	Complexity	Security Layer	Gap Filled
Image data hiding schemes based on metaheuristic optimization: a review [14]	Metaheuristic optimization	Review is classified into optimization-based embedding, optimisation purpose and level of metaheuristics	High (Algorithmic)	Minimal	Specific algorithm usage
Neural networks-based data hiding in digital images: overview [15]	Neural network	Focused on AI-driven hiding	High (Architectural)	Variable	Intelligent embedding
Overview of methods to improve execution time in image steganography and watermarking [16]	Performance improvement	Analyzes efficiency and robustness of algorithms	Theoretical/Numerical	Not a primary focus	Time complexity
This paper	Spatial to adaptive methods	Evolutionary synthesis across domains	Balanced (Practical and Theoretical)	Strong (Chaos and adaptive logic)	Unified adaptive framework

8. Conclusions

Image steganography has evolved significantly over the years, transitioning from simple spatial domain methods to complex hybrid and intelligent techniques that incorporate transform domains, cryptographic integration, and deep learning models. This paper offers an in-depth examination of existing steganography techniques, highlighting their strengths and limitations. These classical methods are employed to enhance the embedding capacity while maintaining the image's visual quality within a specified range. Modern techniques mainly emphasize robustness, security, and capacity enhancement. With the advent of deep learning, chaotic theory, and optimization algorithms, a new paradigm of adaptive and intelligent steganographic systems is emerging. Despite these advancements, challenges remain in achieving an optimal balance between embedding capacity, imperceptibility, and robustness. In this paper, for vulnerability assessment, a limited number of images were taken into consideration to give insight of robustness of the reviewed techniques; however, it may vary on a large dataset. Future research should focus on the use of explainable and interpretable AI models for steganography, standardized evaluation frameworks, and application-specific designs, especially in contexts like digital forensics, secure communications, and privacy-preserving data sharing.

In conclusion, image steganography continues to be a vital and dynamic area of research, and its future lies in the convergence of traditional principles with advanced computational intelligence, ensuring secure and undetectable information hiding in an increasingly digital world.

Author Contributions: S.C., V.K.M. and P.K.S. conceived the idea for this review and developed the overall framework for the comparative analysis of existing methods. The study was carried out under the supervision of G.G. and V.B., who provided conceptual guidance and contributed to refining the research direction. S.C. and P.K.S. designed the review methodology. G.G., P.K.S. and V.B. critically reviewed the manuscript, provided valuable suggestions, and substantially improved its scientific quality. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The dataset used in this study can be made available from the corresponding authors on reasonable request.

Acknowledgments: Authors are thankful to Manipal University Jaipur.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Caelli, W.; Longley, D.; Shain, M. *Information Security Handbook*; Stockton Press: Stockton, CA, USA, 1991.
2. Pfitzmann, B. Information hiding terminology. *Lect. Notes Comput. Sci.* **1996**, *1174*, 347–350. [\[CrossRef\]](#)
3. Eloff, J.; Eloff, M. Information security architecture. *Comput. Fraud Secur.* **2005**, *2005*, 10–16. [\[CrossRef\]](#)
4. Taha, M.S.; Mohd Rahim, M.S.; Lafta, S.A.; Hashim, M.M.; Alzuabidi, H.M. Combination of steganography and cryptography: A short survey. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *518*, 052003. [\[CrossRef\]](#)
5. Anderson, R. *Security Engineering: A Guide to Building Dependable Distributed Systems*; John Wiley & Sons: Hoboken, NJ, USA, 2010.
6. Hartung, F.; Kutter, M. Multimedia watermarking techniques. *Proc. IEEE* **1999**, *87*, 1079–1107. [\[CrossRef\]](#)
7. Alwahbani, S.M.H.; Elshoush, H.T.I. Hybrid Audio Steganography and Cryptography Method Based on High Least Significant Bit (LSB) Layers and One-Time Pad—A Novel Approach. In *Proceedings of the Intelligent Systems and Applications*; Bi, Y., Kapoor, S., Bhatia, R., Eds.; Springer: Cham, Switzerland, 2018; pp. 431–453. [\[CrossRef\]](#)
8. Kahn, D. The history of steganography. In *Proceedings of the International Workshop on Information Hiding*; Springer: Cham, Switzerland, 1996; pp. 1–5. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Pilania, U.; Kumar, M. *Unveiling the Art of Steganography: A Modern Approach*; CRC Press: Boca Raton, FL, USA, 2026.
10. Cheddad, A.; Condell, J.; Curran, K.; Mc Kevitt, P. Digital image steganography: Survey and analysis of current methods. *Signal Process.* **2010**, *90*, 727–752. [\[CrossRef\]](#)

11. Fridrich, J. *Steganography in Digital Media: Principles, Algorithms, and Applications*; Cambridge University Press: Cambridge, UK, 2009.
12. Rabie, T. *Digital Image Steganography: An FFT Approach*; Springer: Cham, Switzerland, 2012; pp. 217–230. [\[CrossRef\]](#)
13. Mostafa, H.; Ali, A.F.; Taweel, G.E. Hybrid curvelet transform and least significant bit for image steganography. In *Proceedings of the 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS)*, Cairo, Egypt, 12–14 December 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 300–305. [\[CrossRef\]](#)
14. Melman, A.; Evsutin, O. Image data hiding schemes based on metaheuristic optimization: A review. *Artif. Intell. Rev.* **2023**, *56*, 15375–15447. [\[CrossRef\]](#)
15. Dzhanashia, K.; Evsutin, O. Neural networks-based data hiding in digital images: Overview. *Neurocomputing* **2024**, *581*, 127499. [\[CrossRef\]](#)
16. Melman, A.; Dzhanashia, K.; Evsutin, O. Overview of methods to improve execution time in image steganography and watermarking. *Comput. Stand. Interfaces* **2025**, *96*, 104066. [\[CrossRef\]](#)
17. Majeed, M.A.; Sulaiman, R.; Shukur, Z.; Hasan, M.K. A Review on Text Steganography Techniques. *Mathematics* **2021**, *9*, 2829. [\[CrossRef\]](#)
18. Kadhim, I.J.; Premaratne, P.; Vial, P.J.; Halloran, B. Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research. *Neurocomputing* **2019**, *335*, 299–326. [\[CrossRef\]](#)
19. Begum, M.; Uddin, M.S. Digital Image Watermarking Techniques: A Review. *Information* **2020**, *11*, 110. [\[CrossRef\]](#)
20. Pradhan, A.; Sahu, A.K.; Swain, G.; Sekhar, K.R. Performance evaluation parameters of image steganography techniques. In *Proceedings of the 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS)*, Bangalore, India, 6–7 May 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–8. [\[CrossRef\]](#)
21. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 248–255. [\[CrossRef\]](#)
22. Bas, P.; Filler, T.; Pevný, T. “Break Our Steganographic System”: The Ins and Outs of Organizing BOSS. In *Information Hiding*; Springer: Cham, Switzerland, 2011; pp. 59–70. [\[CrossRef\]](#)
23. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In *Proceedings of the European Conference on Computer Vision (ECCV)*; Springer: Cham, Switzerland, 2014; pp. 740–755. [\[CrossRef\]](#)
24. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep Learning Face Attributes in the Wild. In *Proceedings of the International Conference on Computer Vision (ICCV)*, Santiago, Chile, 7–13 December 2015.
25. Chandramouli, R.; Memon, N. Analysis of LSB based image steganography techniques. In *Proceedings of the Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, Thessaloniki, Greece, 7–10 October 2001; IEEE: Piscataway, NJ, USA, 2001; Volume 3, pp. 1019–1022.
26. Lee, B.G.; Kim, S.C. *Scrambling Techniques for Digital Transmission*; Springer Science & Business Media: Heidelberg/Berlin, Germany, 2012. [\[CrossRef\]](#)
27. Neeta, D.; Snehal, K.; Jacobs, D. Implementation of LSB Steganography and Its Evaluation for Various Bits. In *Proceedings of the 2006 1st International Conference on Digital Information Management*, Bangalore, India, 6 December 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 173–178. [\[CrossRef\]](#)
28. Al-Aidroos, N.M.; Bahamish, H.A. Image steganography based on LSB matching and image enlargement. In *Proceedings of the 2019 First International Conference of Intelligent Computing and Engineering (ICOICE)*, Hadhramout, Yemen, 15–16 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6. [\[CrossRef\]](#)
29. Chan, C.K.; Cheng, L. Hiding data in images by simple LSB substitution. *Pattern Recognit.* **2004**, *37*, 469–474. [\[CrossRef\]](#)
30. Amirtharajan, R.; Rayappan, J.B.B. An intelligent chaotic embedding approach to enhance stego-image quality. *Inf. Sci.* **2012**, *193*, 115–124. [\[CrossRef\]](#)
31. Hussain, M.; Wahab, A.W.A.; Ho, A.T.; Javed, N.; Jung, K.H. A data hiding scheme using parity-bit pixel value differencing and improved rightmost digit replacement. *Signal Process. Image Commun.* **2017**, *50*, 44–57. [\[CrossRef\]](#)
32. Chakraborty, S.; Jalal, A.S.; Bhatnagar, C. Secret image sharing using grayscale payload decomposition and irreversible image steganography. *J. Inf. Secur. Appl.* **2013**, *18*, 180–192. [\[CrossRef\]](#)
33. Potdar, V.; Chang, E. Grey level modification steganography for secret communication. In *Proceedings of the 2nd IEEE International Conference on Industrial Informatics*, 2004. INDIN’04. 2004, Berlin, Germany, 24–26 June 2004; IEEE: Piscataway, NJ, USA, 2004; pp. 223–228. [\[CrossRef\]](#)
34. Das, P.; Kushwaha, S.C.; Chakraborty, M. Multiple embedding secret key image steganography using LSB substitution and Arnold Transform. In *Proceedings of the 2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, India, 26–27 February 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 845–849. [\[CrossRef\]](#)

35. Muhammad, K.; Ahmad, J.; Rehman, N.U.; Jan, Z.; Sajjad, M. CISSKA-LSB: Color image steganography using stego key-directed adaptive LSB substitution method. *Multimed. Tools Appl.* **2017**, *76*, 8597–8626. [\[CrossRef\]](#)
36. Biswas, B.; Sendrier, N. McEliece cryptosystem implementation: Theory and practice. In *Proceedings of the Post-Quantum Cryptography: Second International Workshop, PQCrypto 2008, Cincinnati, OH, USA, 17–19 October 2008 Proceedings 2*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 47–62. [\[CrossRef\]](#)
37. Alanzy, M.; Alomrani, R.; Alqarni, B.; Almutairi, S. Image Steganography Using LSB and Hybrid Encryption Algorithms. *Appl. Sci.* **2023**, *13*, 11771. [\[CrossRef\]](#)
38. Rustad, S.; Setiadi, D.R.I.M.; Syukur, A.; Andono, P.N. Inverted LSB image steganography using adaptive pattern to improve imperceptibility. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 3559–3568. [\[CrossRef\]](#)
39. Wu, D.C.; Tsai, W.H. A steganographic method for images by pixel-value differencing. *Pattern Recognit. Lett.* **2003**, *24*, 1613–1626. [\[CrossRef\]](#)
40. Pradhan, A.; Sekhar, K.R.; Swain, G. Digital image steganography using LSB substitution, PVD, and EMD. *Math. Probl. Eng.* **2018**, *2018*, 1804953. [\[CrossRef\]](#)
41. Lee, C.F.; Wang, Y.R.; Chang, C.C. A Steganographic Method with High Embedding Capacity by Improving Exploiting Modification Direction. In *Proceedings of the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2007)*, Kaohsiung, Taiwan, 26–28 November 2007; IEEE: Piscataway, NJ, USA, 2007; Volume 1, pp. 497–500. [\[CrossRef\]](#)
42. Zhang, X.; Wang, S. Efficient steganographic embedding by exploiting modification direction. *IEEE Commun. Lett.* **2006**, *10*, 781–783. [\[CrossRef\]](#)
43. Islam, A.U.; Khalid, F.; Shah, M.; Khan, Z.; Mahmood, T.; Khan, A.; Ali, U.; Naeem, M. An improved image steganography technique based on MSB using bit differencing. In *Proceedings of the 2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, Dublin, Ireland, 24–26 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 265–269. [\[CrossRef\]](#)
44. Chang, C.C.; Chen, T.S.; Chung, L.Z. A steganographic method based upon JPEG and quantization table modification. *Inf. Sci.* **2002**, *141*, 123–138. [\[CrossRef\]](#)
45. Chang, C.C.; Lin, C.C.; Tseng, C.S.; Tai, W.L. Reversible hiding in DCT-based compressed images. *Inf. Sci.* **2007**, *177*, 2768–2786. [\[CrossRef\]](#)
46. Tseng, H.W.; Chang, C.C. High Capacity Data Hiding in JPEG-Compressed Images. *Informatica* **2004**, *15*, 127–142. [\[CrossRef\]](#)
47. Abdelwahab, A.A.; Hassaan, L.A. A discrete wavelet transform based technique for image data hiding. In *Proceedings of the 2008 National Radio Science Conference, Tanta, Egypt, 18–20 March 2008*; IEEE: Piscataway, NJ, USA, 2008; pp. 1–9. [\[CrossRef\]](#)
48. Kumar, V.; Kumar, D. A modified DWT-based image steganography technique. *Multimed. Tools Appl.* **2018**, *77*, 13279–13308. [\[CrossRef\]](#)
49. Narasimmalou, T.; Allen, J.R. Optimized discrete wavelet transform based steganography. In *Proceedings of the 2012 IEEE International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, Ramanathapuram, India, 23–25 August 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 88–91. [\[CrossRef\]](#)
50. Ioannidou, A.; Halkidis, S.T.; Stephanides, G. A novel technique for image steganography based on a high payload method and edge detection. *Expert Syst. Appl.* **2012**, *39*, 11517–11524. [\[CrossRef\]](#)
51. Lakshmi Sirisha, B. Image steganography based on SVD and DWT techniques. *J. Discret. Math. Sci. Cryptogr.* **2020**, *23*, 779–786. [\[CrossRef\]](#)
52. Sharma, V.; Srivastava, D.; Mathur, P. A Daubechies DWT Based Image Steganography Using Smoothing Operation. *Int. Arab J. Inf. Technol.* **2020**, *17*, 17–24. [\[CrossRef\]](#)
53. Durafe, A.; Patidar, V. Development and analysis of IWT-SVD and DWT-SVD steganography using fractal cover. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 4483–4498. [\[CrossRef\]](#)
54. Miri, A.; Faez, K. An image steganography method based on integer wavelet transform. *Multimed. Tools Appl.* **2018**, *77*, 13133–13144.
55. Pramanik, S. An adaptive image steganography approach depending on integer wavelet transform and genetic algorithm. *Multimed. Tools Appl.* **2023**, *82*, 34287–34319. [\[CrossRef\]](#)
56. Chowdhuri, P.; Pal, P.; Si, T. A novel steganographic technique for medical image using SVM and IWT. *Multimed. Tools Appl.* **2023**, *82*, 20497–20516. [\[CrossRef\]](#) [\[PubMed\]](#)
57. Do, M.N.; Vetterli, M. The contourlet transform: An efficient directional multiresolution image representation. *IEEE Trans. Image Process.* **2005**, *14*, 2091–2106. [\[CrossRef\]](#) [\[PubMed\]](#)
58. Subhedar, M.S.; Mankar, V.H. Performance Evaluation of Image Steganography Based on Cover Selection and Contourlet Transform. In *Proceedings of the 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies, Pune, India, 15–16 November 2013*; IEEE: Piscataway, NJ, USA, 2013; pp. 172–177. [\[CrossRef\]](#)
59. Subhedar, M.S.; Mankar, V.H. Image steganography using contourlet transform and matrix decomposition techniques. *Multimed. Tools Appl.* **2019**, *78*, 22155–22181. [\[CrossRef\]](#)

60. Kumar, R.; Singhal, S.; Sharma, V.K. Efficient image steganography method using contourlet transform and geometric-based pixel encryption for enhanced security. *Sci. Rep.* **2026**, *16*, 16771. [[CrossRef](#)] [[PubMed](#)]
61. Ma, J.; Plonka, G. The curvelet transform. *IEEE Signal Process. Mag.* **2010**, *27*, 118–133. [[CrossRef](#)]
62. Subhedar, M.S.; Mankar, V.H. Curvelet transform and cover selection for secure steganography. *Multimed. Tools Appl.* **2018**, *77*, 8115–8138.
63. Gharavi, H.; Rajaei, B. A Robust Steganography Algorithm Based on Curvelet Transform. In *Proceedings of the Electrical Engineering (ICEE), Iranian Conference on, Mashhad, Iran, 8–10 May 2018*; IEEE: Piscataway, NJ, USA, 2018; pp. 1624–1628. [[CrossRef](#)]
64. Zia, U.; McCartney, M.; Scotney, B.; Martinez, J.; AbuTair, M.; Memon, J.; Sajjad, A. Survey on image encryption techniques using chaotic maps in spatial, transform and spatiotemporal domains. *Int. J. Inf. Secur.* **2022**, *21*, 917–935. [[CrossRef](#)]
65. Zhou, N.R.; Wu, J.W.; Chen, M.X.; Wang, M.M. A quantum image encryption and watermarking algorithm based on qdct and baker map. *Int. J. Theor. Phys.* **2024**, *63*, 100. [[CrossRef](#)]
66. Rani, M.; Agarwal, R. A new experimental approach to study the stability of logistic map. *Chaos Solitons Fractals* **2009**, *41*, 2062–2066. [[CrossRef](#)]
67. Hénon, M. A two-dimensional mapping with a strange attractor. *Commun. Math. Phys.* **1976**, *50*, 69–77. [[CrossRef](#)]
68. Michielin, O.; Phillipson, P.E. Map dynamics study of the Lorenz equations. *Int. J. Bifurc. Chaos* **1997**, *7*, 373–382. [[CrossRef](#)]
69. Roy, R.; Sarkar, A.; Changder, S. Chaos based edge adaptive image steganography. *Procedia Technol.* **2013**, *10*, 138–146. [[CrossRef](#)]
70. Najeena, K.; Imran, B. An efficient steganographic technique based on chaotic maps and adaptive PPM embedding. In *Proceedings of the 2013 International Conference on Signal Processing, Image Processing & Pattern Recognition, Coimbatore, India, 7–8 February 2013*; IEEE: Piscataway, NJ, USA, 2013; pp. 293–297. [[CrossRef](#)]
71. Anees, A.; Siddiqui, A.M.; Ahmed, J.; Hussain, I. A technique for digital steganography using chaotic maps. *Nonlinear Dyn.* **2014**, *75*, 807–816. [[CrossRef](#)]
72. Valandar, M.Y.; Ayubi, P.; Barani, M.J. A new transform domain steganography based on modified logistic chaotic map for color images. *J. Inf. Secur. Appl.* **2017**, *34*, 142–151. [[CrossRef](#)]
73. Saidi, M.; Hermassi, H.; Rhouma, R.; Belghith, S. A new adaptive image steganography scheme based on DCT and chaotic map. *Multimed. Tools Appl.* **2017**, *76*, 13493–13510. [[CrossRef](#)]
74. Savithri, G.; Vinupriya; Mane, S.; Saira Banu, J. Parallel implementation of RSA 2D-DCT steganography and chaotic 2D-DCT steganography. In *Proceedings of the International Conference on Computer Vision and Image Processing: CVIP 2016*; Springer: Singapore, 2017; Volume 1, pp. 593–605. [[CrossRef](#)]
75. Benlashram, A.; Al-Ghamdi, M.; AlTalhi, R.; Laabidi, P.K. A novel approach of image encryption using pixel shuffling and 3D chaotic map. *J. Phys. Conf. Ser.* **2020**, *1447*, 012009. [[CrossRef](#)]
76. Chaudhary, S.; Hiranwal, S.; Gupta, C.P. A New Hybrid Chaotic Map-Based Image Steganography Using Spectral Graph Wavelet. In *Proceedings of the Intelligent Computing Techniques for Smart Energy Systems*; Tripathi, A., Soni, A., Shrivastava, A., Swarnkar, A., Sahariya, J., Eds.; Springer: Singapore, 2022; pp. 211–223. [[CrossRef](#)]
77. Dey, D.; Jana, D.K.; Bhunia, P. A Novel Approach to Steganography Security: Embedding Encrypted Text in Color Images Using Langton's Ant Cellular Automaton and Hybrid Chaotic Map. *SN Comput. Sci.* **2025**, *6*, 402. [[CrossRef](#)]
78. Zhang, H.; Feng, X.; Sun, J.; Yan, P. Chaotic Image Security Techniques and Developments: A Review. *Mathematics* **2025**, *13*, 1976. [[CrossRef](#)]
79. Wen, H.; Lin, Y. Cryptanalysis of an image encryption algorithm using quantum chaotic map and DNA coding. *Expert Syst. Appl.* **2024**, *237*, 121514. [[CrossRef](#)]
80. Roy, R.; Laha, S. Optimization of stego image retaining secret information using genetic algorithm with 8-connected PSNR. *Procedia Comput. Sci.* **2015**, *60*, 468–477. [[CrossRef](#)]
81. Uma Maheswari, S.; Jude Hemanth, D. Performance enhanced image steganography systems using transforms and optimization techniques. *Multimed. Tools Appl.* **2017**, *76*, 415–436. [[CrossRef](#)]
82. Nipanikar, S.; Deepthi, V.H.; Kulkarni, N. A sparse representation based image steganography using particle swarm optimization and wavelet transform. *Alex. Eng. J.* **2018**, *57*, 2343–2356. [[CrossRef](#)]
83. Mohsin, A.H.; Zaidan, A.; Zaidan, B.; Albahri, O.; Albahri, A.; Alsalem, M.; Mohammed, K.; Nidhal, S.; Jalood, N.S.; Jasim, A.N.; et al. New method of image steganography based on particle swarm optimization algorithm in spatial domain for high embedding capacity. *IEEE Access* **2019**, *7*, 168994–169010. [[CrossRef](#)]
84. Muhuri, P.K.; Ashraf, Z.; Goel, S. A novel image steganographic method based on integer wavelet transformation and particle swarm optimization. *Appl. Soft Comput.* **2020**, *92*, 106257. [[CrossRef](#)]
85. Sharma, N.; Batra, U. An enhanced Huffman-PSO based image optimization algorithm for image steganography. *Genet. Program. Evolvable Mach.* **2021**, *22*, 189–205. [[CrossRef](#)]
86. Gurunathan, K.; Rajagopalan, S. A stegano-visual cryptography technique for multimedia security. *Multimed. Tools Appl.* **2020**, *79*, 3893–3911. [[CrossRef](#)]

87. Laimeche, L.; Meraoumia, A.; Bendjenna, H. Enhancing LSB embedding schemes using chaotic maps systems. *Neural Comput. Appl.* **2020**, *32*, 16605–16623. [\[CrossRef\]](#)
88. Roselin Kiruba, R.; Sree Sharmila, T. Secure data hiding by fruit fly optimization improved hybridized seeker algorithm. *Multidimens. Syst. Signal Process.* **2021**, *32*, 405–430. [\[CrossRef\]](#)
89. Tong, H.; Li, T.; Xu, Y.; Su, X.; Qiao, G. Chaotic coyote optimization algorithm for image encryption and steganography. *Multimed. Tools Appl.* **2024**, *83*, 20861–20887. [\[CrossRef\]](#)
90. Rashad, T.E.; Khamis, S.M.; Bahig, H.M.; Nassr, D.I. A recent approach based on inspired crows behavior in steganography. *Evol. Intell.* **2024**, *17*, 2617–2636. [\[CrossRef\]](#)
91. Bhandari, M.; Panday, S.; Bhatta, C.P.; Panday, S.P. Image Steganography Approach Based Ant Colony Optimization with Triangular Chaotic Map. In *Proceedings of the 2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM), Gautam Buddha Nagar, India, 23–25 February 2022*; IEEE: Piscataway, NJ, USA, 2022; Volume 2, pp. 429–434. [\[CrossRef\]](#)
92. Eshmawi, A.; Alsubhany, S.A.; Abdel-Khalek, S.; Mansour, R.F. Competitive Swarm Optimization with Encryption Based Steganography for Digital Image Security. *Comput. Mater. Contin.* **2022**, *72*. [\[CrossRef\]](#)
93. Abd-El-Atty, B. A robust medical image steganography approach based on particle swarm optimization algorithm and quantum walks. *Neural Comput. Appl.* **2023**, *35*, 773–785. [\[CrossRef\]](#)
94. Fahim, A.; Raslan, Y. Optimized steganography techniques based on PVDS and genetic algorithm. *Alex. Eng. J.* **2023**, *85*, 245–260. [\[CrossRef\]](#)
95. Bahaddad, A.A.; Ali Almarhabi, K.; Abdel-Khalek, S. Image steganography technique based on bald eagle search optimal pixel selection with chaotic encryption. *Alex. Eng. J.* **2023**, *75*, 41–54. [\[CrossRef\]](#)
96. Fadel, M.M.; Said, W.; Hagrass, E.A.; Arnous, R. A Fast and Low Distortion Image Steganography Framework Based on Nature-Inspired Optimizers. *IEEE Access* **2023**, *11*, 125768–125789. [\[CrossRef\]](#)
97. Wu, P.; Yang, Y.; Li, X. StegNet: Mega Image Steganography Capacity with Deep Convolutional Network. *Future Internet* **2018**, *10*, 54. [\[CrossRef\]](#)
98. El-Emam, N.; Al-Zubidy, R. New steganography algorithm to conceal a large amount of secret message using hybrid adaptive neural networks with modified adaptive genetic algorithm. *J. Syst. Softw.* **2013**, *86*, 1465–1481. [\[CrossRef\]](#)
99. Sharma, K.; Aggarwal, A.; Singhania, T.; Gupta, D.; Khanna, A. Hiding Data in Images Using Cryptography and Deep Neural Network. *J. Artif. Intell. Syst.* **2019**, *1*, 143–162. [\[CrossRef\]](#)
100. Zhu, J.; Kaplan, R.; Johnson, J.; Fei-Fei, L. HiDDeN: Hiding Data With Deep Networks. In *Proceedings of the Computer Vision—ECCV 2018: 15th European Conference, Munich, Germany, 8–14 September 2018*; Springer: Cham, Switzerland, 2018; pp. 682–697. [\[CrossRef\]](#)
101. Duan, X.; Guo, D.; Liu, N.; Li, B.; Gou, M.; Qin, C. A New High Capacity Image Steganography Method Combined With Image Elliptic Curve Cryptography and Deep Neural Network. *IEEE Access* **2020**, *8*, 25777–25788. [\[CrossRef\]](#)
102. Hu, D.; Wang, L.; Jiang, W.; Zheng, S.; Li, B. A Novel Image Steganography Method via Deep Convolutional Generative Adversarial Networks. *IEEE Access* **2018**, *6*, 38303–38314. [\[CrossRef\]](#)
103. Huo, L.; Chen, R.; Wei, J.; Huang, L. A High-Capacity and High-Security Image Steganography Network Based on Chaotic Mapping and Generative Adversarial Networks. *Appl. Sci.* **2024**, *14*, 1225. [\[CrossRef\]](#)
104. Li, X.; Chen, L.; Lai, J.; Fu, Z.; Liu, S. GAN-based image steganography by exploiting transform domain knowledge with deep networks. *Multimed. Syst.* **2024**, *30*, 224. [\[CrossRef\]](#)
105. Kumar, V.; Laddha, S.; Sharma, A.; Dogra, N. Steganography techniques using convolutional neural networks. *Rev. Comput. Eng. Stud.* **2020**, *7*, 66–73. [\[CrossRef\]](#)
106. Zhong, T.; Jin, X.; Tong, K. 3D-CNN autoencoder for plenoptic image compression. In *Proceedings of the 2020 IEEE International Conference on Visual Communications and Image Processing (VCIP), Macau, China, 1–4 December 2020*; IEEE: Piscataway, NJ, USA, 2020; pp. 209–212. [\[CrossRef\]](#)
107. Subramanian, N.; Cheheb, I.; Elharrouss, O.; Al-Maadeed, S.; Bouridane, A. End-to-end image steganography using deep convolutional autoencoders. *IEEE Access* **2021**, *9*, 135585–135593. [\[CrossRef\]](#)
108. Murthy, M.L.; Madhavi, P.B.; Ahamad, S.A.; Vamsi, K.; Rao, Y.M. Implementation and analysis of image steganography using convolution neural networks. In *Proceedings of the 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC), Bengaluru, India, 18–19 November 2022*; IEEE: Piscataway, NJ, USA, 2022; pp. 108–113. [\[CrossRef\]](#)
109. Brindha, K.; Maruthi, R. A robust and secure image steganography using convolutional neural networks and transform methods. In *Proceedings of the 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 17–19 May 2023*; IEEE: Piscataway, NJ, USA, 2023; pp. 1338–1346. [\[CrossRef\]](#)
110. Ke, X.; Wu, H.; Guo, W. Stegformer: Rebuilding the glory of autoencoder-based steganography. In *Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 26–27 February 2024*; AAAI: Washington, DC, USA, 2024; Volume 38, pp. 2723–2731. [\[CrossRef\]](#)

111. Sanjalawe, Y.; Al-E'mari, S.; Fraihat, S.; Abualhaj, M.; Alzubi, E. A deep learning-driven multi-layered steganographic approach for enhanced data security. *Sci. Rep.* **2025**, *15*, 4761. [\[CrossRef\]](#) [\[PubMed\]](#)
112. Wang, Z.; Zhou, M.; Liu, B.; Li, T. Deep image steganography using transformer and recursive permutation. *Entropy* **2022**, *24*, 878. [\[CrossRef\]](#) [\[PubMed\]](#)
113. Xiao, C.; Peng, S.; Zhang, L.; Wang, J.; Ding, D.; Zhang, J. A transformer-based adversarial network framework for steganography. *Expert Syst. Appl.* **2025**, *269*, 126391. [\[CrossRef\]](#)
114. Yang, C.; Wang, S.; Huang, Y.; Guo, M. SNR: One single network for image steganography with robust post-save recovery. *Neurocomputing* **2025**, *651*, 130929. [\[CrossRef\]](#)
115. Hu, X.; Li, S.; Ying, Q.; Peng, W.; Zhang, X.; Qian, Z. Establishing robust generative image steganography via popular stable diffusion. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 8094–8108. [\[CrossRef\]](#)
116. Yang, L.; Zhang, Z.; Song, Y.; Hong, S.; Xu, R.; Zhao, Y.; Zhang, W.; Cui, B.; Yang, M.H. Diffusion Models: A Comprehensive Survey of Methods and Applications. *ACM Comput. Surv.* **2023**, *56*, 105. [\[CrossRef\]](#)
117. Zhang, S.; Xue, J.; Fu, J.; An, D. A review of generative coverless image steganography based on diffusion models. *Neurocomputing* **2026**, *677*, 133070. [\[CrossRef\]](#)
118. Shafique, M.A.; Munir, A.; Kong, J. Deep Learning Performance Characterization on GPUs for Various Quantization Frameworks. *AI* **2023**, *4*, 926–948. [\[CrossRef\]](#)
119. NVIDIA. ResNet-50 v1.5 for TensorFlow Performance, 2023. Available online: https://catalog.ngc.nvidia.com/orgs/nvidia/-/resources/resnet_50_v1_5_for_tensorflow/-?_lr=1 (accessed on 3 April 2026).
120. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016*; IEEE: Piscataway, NJ, USA, 2016; pp. 770–778. [\[CrossRef\]](#)
121. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In *Proceedings of the Artificial Neural Networks and Machine Learning—ICANN 2011*; Honkela, T., Duch, W., Girolami, M., Kaski, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 52–59. [\[CrossRef\]](#)
122. Mohsin, A.H.; Zaidan, A.; Zaidan, B.; Mohammed, K.; Albahri, O.S.; Albahri, A.S.; Alsalem, M. PSO–Blockchain-based image steganography: Towards a new method to secure updating and sharing COVID-19 data in decentralised hospitals intelligence architecture. *Multimed. Tools Appl.* **2021**, *80*, 14137–14161. [\[CrossRef\]](#) [\[PubMed\]](#)
123. Takaoğlu, M.; Özyavaş, A.; Ajlouni, N.; Alshahrani, A.; Alkasasbeh, B. A Novel and Robust Hybrid Blockchain and Steganography Scheme. *Appl. Sci.* **2021**, *11*, 10698. [\[CrossRef\]](#)
124. Lakkundi, P.; Tanuj, B.; M, M.; Manokaran, S.; Reddy, A.H.M. Secure Inter-Hospital Communication Using DNA Encryption and Image Steganography. In *Proceedings of the 2025 9th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, India, 20–22 November 2025*; IEEE: Piscataway, NJ, USA, 2025; pp. 1–6. [\[CrossRef\]](#)
125. Mahjabin, T.; Olteanu, A.; Xiao, Y.; Han, W.; Li, T.; Sun, W. A Survey on DNA-Based Cryptography and Steganography. *IEEE Access* **2023**, *11*, 116423–116451. [\[CrossRef\]](#)
126. Fridrich, J.; Kodovsky, J. Rich models for steganalysis of digital images. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 868–882. [\[CrossRef\]](#)
127. Croix, N.J.D.L.; Ahmad, T.; Han, F. Comprehensive survey on image steganalysis using deep learning. *Array* **2024**, *22*, 100353. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.