People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Bouira University

Faculty of Exact Sciences
Department of Computer Science

Master Thesis

Robust Steganography Techniques for Data Tampering Resilience

---

Submitted by:

Abid Akram
Tahraoui Mustapha

Supervised by:

Dr. Benaissi Sellami

Submitted in partial fulfillment of the requirements for the degree of
Master in Computer Science

Academic Year: 2025 – 2026

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General Background

In contrast to cryptography, which focuses on protecting the content of a message by transforming it into an unreadable form, steganography aims to conceal the very existence of communication. While digital watermarking shares similarities with steganography, it is primarily designed for ownership verification and copyright protection rather than covert communication. Robust steganography lies at the intersection of these domains, seeking to maintain hidden communication while ensuring resistance to intentional or unintentional image modifications.

In real-world digital environments, images rarely remain unchanged after distribution. Common platforms such as social media networks, cloud storage services, and messaging applications routinely apply operations including lossy compression, resizing, filtering, and format conversion. These automatic transformations pose a significant threat to traditional steganographic techniques, which are often designed under ideal transmission assumptions. As a result, the development of steganographic systems capable of surviving such uncontrolled and hostile processing environments has become an essential research challenge.

Steganography is the art and science of invisible communication, achieved by concealing information within other digital content in such a way that the existence of the hidden message remains undetectable. The term steganography is derived from the Greek words stegos, meaning "cover," and graphia, meaning "writing," and is commonly interpreted as "covered writing."[5]. In the context of image steganography, secret information is embedded exclusively within digital images, which serve as the cover medium for hidden communication.

1.2 Problem Statement

From the analysis above, the core challenges addressed in this work can be summarized as follows:

- Existing steganographic methods exhibit high fragility when subjected to common signal processing operations such as compression, noise addition, and filtering, resulting in partial or complete loss of embedded information.
- Geometric transformations, including cropping and resizing, introduce spatial desynchronization between the embedding and extraction processes, severely limiting reliable data recovery.
- Most current approaches prioritize imperceptibility while neglecting the survivability of the hidden payload under compound or severe distortions.
- The majority of embedding strategies focus on protecting embedding locations rather than enabling resilience or reconstruction of the hidden message itself.

These limitations highlight the need for a steganographic framework that shifts the design focus from invisibility-centered embedding toward payload-oriented survivability, enabling partial recovery and reliable communication in hostile and uncontrolled digital environments.

1.3 Research Objectives

The primary objective of this research is to address the limitations of conventional image steganography techniques by prioritizing the survivability of hidden data under adverse and uncontrolled image degradation. Rather than focusing exclusively on high-capacity or imperceptible embedding, this work aims to enhance the resilience of the embedded payload against both signal processing and geometric distortions.

The specific objectives of this thesis are as follows:

- To design a robust steganographic framework that improves the survivability of hidden information under severe and compound image degradations.
- To structure the secret payload into interconnected fragments that enable partial data recovery when segments of the stego-image are lost or corrupted.
- To distribute embedded message fragments across stable image regions and multiple transform domains in order to reduce the impact of localized tampering.
- To develop an extraction and reconstruction strategy capable of recovering usable information even in the presence of spatial misalignment and partial data loss.
- To evaluate the proposed framework using standard image quality and robustness metrics, including PSNR [6] and Bit Error Rate (BER) [7], under realistic attack scenarios.

1.4 Scope and Contributions

This research focuses on robust data hiding in digital images, with particular emphasis on improving the survivability of hidden information under common image degradation and tampering scenarios. The scope of this work is limited to still images and does not address steganography in video, audio, or real-time streaming media. Furthermore, the proposed framework does not aim to maximize payload capacity, but instead prioritizes reliable data recovery under adverse conditions.

The study assumes that image degradation may occur due to both unintentional processing (such as compression and resizing) and intentional tampering (such as cropping or localized modification). However, the framework does not guarantee successful recovery under all extreme or adversarial scenarios, particularly in cases of complete image destruction or aggressive multi-stage attacks.

The main contributions of this thesis can be summarized as follows:

- Proposing a robustness-oriented steganographic framework that emphasizes payload survivability rather than solely protecting embedding locations.
- Introducing a structured message fragmentation strategy that enables partial reconstruction of the hidden data when portions of the stego-image are lost or corrupted.
- Designing a multi-domain embedding strategy that distributes message fragments across different image regions to reduce sensitivity to localized distortions.
- Developing an extraction and reconstruction mechanism capable of recovering meaningful information under spatial misalignment and partial data loss.
- Providing an experimental evaluation of the proposed framework under realistic signal processing and geometric attack scenarios.

1.5 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 presents a comprehensive review of existing steganographic techniques, with particular emphasis on robustness-oriented approaches and their limitations under image degradation and tampering. Chapter 3 introduces the proposed robust steganography framework, detailing its design principles, message structuring strategy, and embedding and extraction mechanisms. Chapter 4 describes the implementation details, experimental setup, datasets, attack models, and evaluation metrics used in this study. Chapter 5 presents and discusses the experimental results, including imperceptibility and robustness analyses as well as comparative evaluations with classical methods. Finally, Chapter 6 concludes the thesis by summarizing the main findings, discussing limitations, and outlining directions for future research.

Chapter 2

Literature Review

2.1 Fundamentals of Image Steganography

Image steganography refers to techniques that embed secret information within digital images while preserving the visual appearance of the carrier. In contrast to introductory definitions presented earlier, this chapter focuses on how steganographic systems are modeled, evaluated, and constrained in practical scenarios. A widely adopted theoretical model is the prisoner's problem, in which the security of a steganographic system depends on the inability of an adversary to statistically distinguish between cover images and stego-images [8]. This model emphasizes that effective steganography must balance invisibility with functional robustness under realistic transmission conditions.

2.1.1 Imperceptibility, Capacity, and Robustness Trade-off

The efficacy of the image steganography system is inevitably limited by the trade-off between the following two factors: between three key performance criteria: imperceptibility, payload capacity, and robustness. These Three properties are necessarily interconnected so that enhancement in one area will result in improvement in other areas as well. decline in at least one of the others. This pattern is often represented in a triangular relationship: This represents a trade-off and is one of the main challenges in designing a steganographic system [9].

- **Imperceptibility** refers to the degree to which a stego-image remains visually and statistically indistinguishable from the corresponding cover image. Imperceptibility is a crucial factor in steganography to avoid being detected either by human observers or statistical steganalysis algorithms. Methods which Agressively embedded data could introduce artifacts that affect either the statistical distribution or signal quality. For example, data that has risk of detection [10].

- **Payload capacity** denotes the amount of secret information that can be embedded within a cover image. While higher capacity improves communication efficiency, it typically requires stronger or more frequent modifications to the image content, which negatively impacts imperceptibility and increases vulnerability to image processing operations. Consequently, high-capacity embedding schemes often sacrifice robustness in order to maximize payload [11].

- **Robustness** describes the ability of the embedded information to survive image degradation caused by signal processing operations or intentional attacks. Robust steganographic schemes are designed to withstand distortions such as compression, noise addition, filtering, and geometric transformations. Achieving robustness usually requires redundancy, structured embedding, or the use of stable image components, which in turn reduces the effective payload capacity [1].

In the context of robust steganography, robustness is frequently prioritized over capacity, particularly in applications where data recovery is more critical than transmission efficiency. As a result,

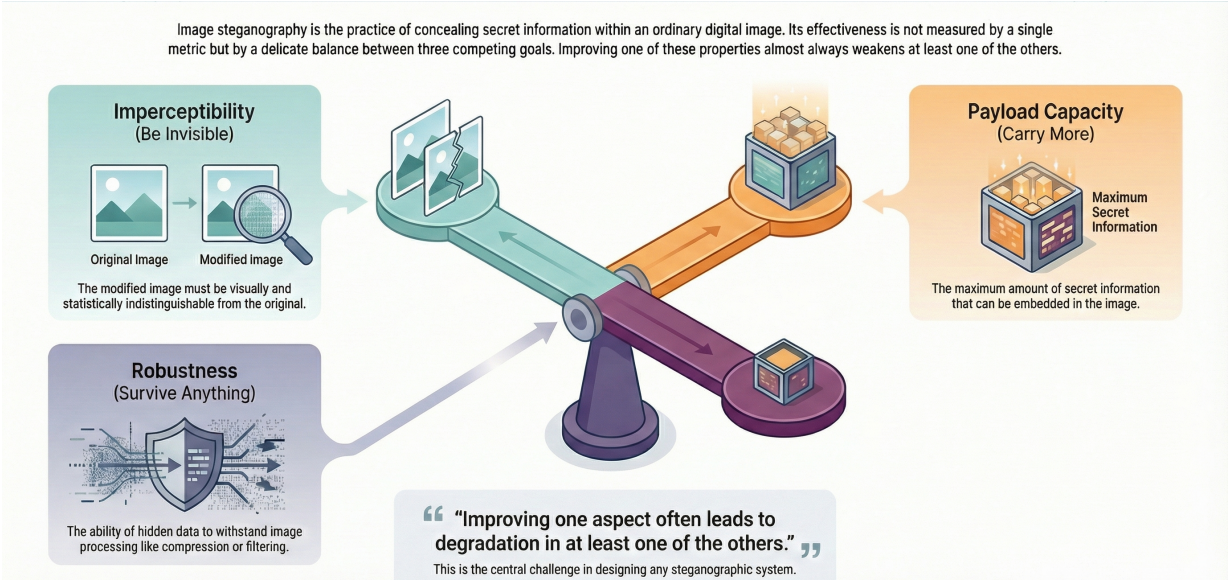


Figure 2.1: Steganography Trilemma: Three Competing Goals

many robust systems intentionally operate at lower payload rates to ensure reliable extraction under adverse conditions. Balancing these three competing objectives remains a key research challenge and motivates the exploration of resilience-oriented steganographic frameworks.

2.2 Spatial Domain Techniques

Spatial domain approaches embed the secret by directly modifying the intensities of pixels. the cover image. Among the methods, one can distinguish LSB substitution and its variants are the most studied families due to their simplicity and low computational complexity. High embedding capacity[11]. In LSB-based methods, the least significant bits of pixel values are changed to convey secret information, leading to less than ideally perceivable distortion.

Spatial domain techniques, despite all their advantages, suffer from some serious drawbacks in terms of robustness. Since the embedding procedure acts directly on pixel values, the hidden data is highly susceptible under common image processing manipulations. Even light lossy compression, noise addition, Filtering irreversibly destroys some fractional bits of the least significant bits, and the result can be a partial or complete loss. of the embedded message[10]. Hence, most spatial domain methods are unsuitable for applications that require robustness against image degradation or tampering as a main constraint.

2.2.1 Transform Domain Techniques

According to the Discrete Cosine Transform (DCT), the DCT is an example of a method of digital image steganography that embeds information into the transform domain of a digital image, instead of modifying the pixel intensity values directly. The DCT takes advantage of the ability of the human visual system to be less sensitive to different spectral components in the image, particularly when working in the frequency or multi-resolution space[1].

2.2.2 Discrete Cosine Transform (DCT)-Based Steganography

The Discrete Cosine Transform (DCT) is a fundamental transform-domain technique widely used in image steganography due to its strong connection with the JPEG compression standard. Unlike spatial-domain approaches that directly modify pixel intensity values, DCT-based steganography embeds secret information within the frequency coefficients of an image. This allows data hiding

to exploit the characteristics of the human visual system, which is less sensitive to modifications in certain frequency components [9, 1].

Overview of the DCT Transformation

The DCT converts spatial image data into a frequency-domain representation. In practice, a grayscale or luminance image is first divided into non-overlapping blocks of size 8×8 pixels. Each block is then transformed independently using the DCT, resulting in 64 coefficients that represent different spatial frequency components.

The resulting coefficients can be categorized as:

- **Low-frequency coefficients:** Represent the average intensity and coarse image structures.
- **Mid-frequency coefficients:** Represent edges and moderate texture details.
- **High-frequency coefficients:** Represent fine details and noise-like components.

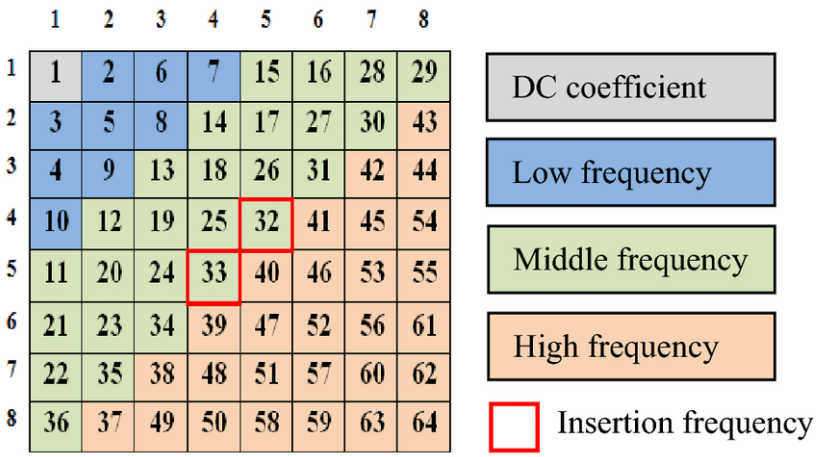


Figure 2.2: DCT coefficient distribution showing low-, mid-, and high-frequency components in an 8×8 block

Embedding Process in DCT-Based Steganography

The embedding procedure in DCT-based steganography typically follows a structured sequence of steps:

1. **Color Space Conversion:** For color images, the RGB image is commonly converted into a luminance-chrominance color space (such as YCbCr). Data embedding is primarily performed in the luminance (Y) channel, as it carries the most perceptual information.

   The YCbCr Components The YCbCr space represents an image through three distinct components:

   - **Y (Luminance):** Represents the brightness information of the image.
   - **Cb (Chroma Blue):** Represents the difference between the blue component and a reference value.
   - **Cr (Chroma Red):** Represents the difference between the red component and a reference value.

Mathematical Transformation The conversion from the digital RGB space to YCbCr is defined by a linear transformation. According to the ITU-R BT.601 standard, the transformation matrix for 8-bit digital signals is expressed as follows:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.1}$$

In a more simplified normalized form, the equations can be written as:

$$Y = 0.299R + 0.587G + 0.114B \tag{2.2}$$
$$Cb = -0.1687R - 0.3313G + 0.5B + 128 \tag{2.3}$$
$$Cr = 0.5R - 0.4187G - 0.0813B + 128 \tag{2.4}$$

Significance for Robust Steganography By utilizing the YCbCr space, the steganographic algorithm can selectively embed data into the chrominance channels (Cb or Cr). Since the Human Visual System (HVS) is less sensitive to these channels, the hidden data can better withstand degradation such as lossy JPEG compression, which typically applies heavier quantization to the chrominance components than to the luminance.

2. **Block Division:** The luminance component is divided into non-overlapping 8×8 blocks to match the JPEG compression structure.

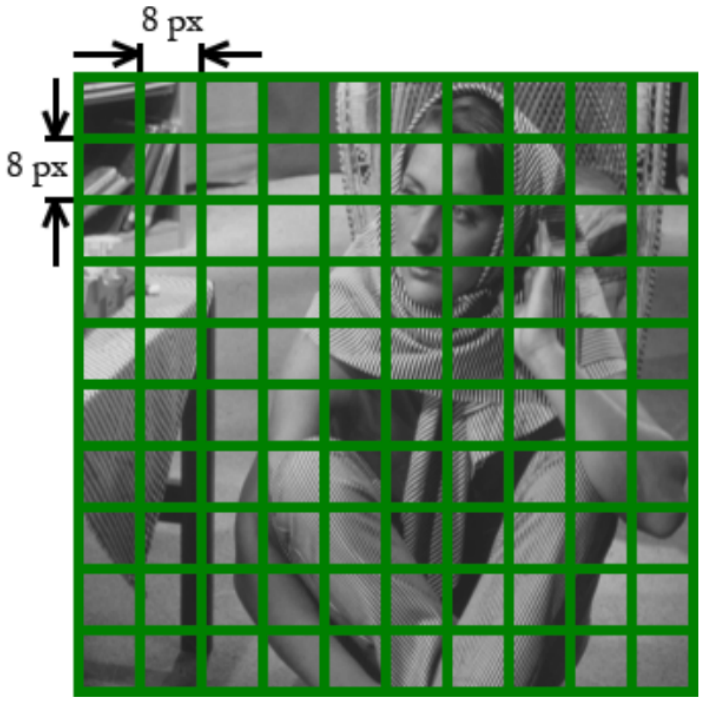


Figure 2.3: DCT deivision image in an 8×8 blocks

3. **DCT Application:** Each block is transformed from the spatial domain to the frequency domain using the DCT, producing a matrix of frequency coefficients.

Frequency Domain Transformation: Block-Based DCT To achieve robustness against signal processing attacks, the proposed framework employs the Discrete Cosine Transform (DCT). The process follows these steps:

- Division of the image into 8×8 pixel blocks.
- Application of the DCT to each block.

Block Partitioning The image channel (e.g., the Cb or Cr component) is first partitioned into non-overlapping blocks of size $N \times N$, where $N = 8$. This localization allows the algorithm to handle local image characteristics effectively. Let $f(i, j)$ represent the intensity value at coordinates (i, j) within an 8×8 block.

Two-Dimensional DCT (2D-DCT) For each block, the 2D-DCT is applied to convert the spatial data into the frequency domain. The DCT-II variant is defined as:

$$F(u, v) = \frac{1}{4} C(u)C(v) \sum_{i=0}^{7} \sum_{j=0}^{7} f(i, j) \cos\left[\frac{(2i+1)u\pi}{16}\right] \cos\left[\frac{(2j+1)v\pi}{16}\right] \tag{2.5}$$

where u, v are the horizontal and vertical frequencies $u, v \in \{0, 1, \ldots, 7\}$, and the normalization factors $C(u)$ and $C(v)$ are defined as:

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } k = 0 \\ 1 & \text{if } k > 0 \end{cases} \tag{2.6}$$

Coefficient Analysis and Robustness The resulting 8×8 matrix $F(u, v)$ consists of:

- **DC Coefficient ($F(0, 0)$):** Represents the average intensity of the block. While it holds the most energy, modifying it significantly impacts visual quality.
- **AC Coefficients ($F(u, v)$ where $u, v \neq 0$):** Represent higher frequency details.

DC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC
AC	AC	AC	AC	AC	AC	AC	AC

Figure 2.4: DC and AC coefficient on DCT Transformation

In this research, we target the **mid-frequency coefficients** for data embedding. High-frequency coefficients are often discarded during lossy compression (quantization), while low-frequency/DC coefficients are too sensitive to changes. Mid-frequency embedding provides an optimal balance between imperceptibility and robustness against tampering.

4. **Coefficient Selection:** A predefined set of mid-frequency coefficients is selected for embedding. Low-frequency coefficients are avoided to prevent visible distortion, while high-frequency coefficients are avoided due to their vulnerability to compression.
5. **Data Embedding:** Secret bits are embedded by modifying the selected DCT coefficients using techniques such as coefficient quantization, parity modification, or least significant bit alteration of coefficient values.
6. **Inverse DCT:** After embedding, the modified coefficients are transformed back into the spatial domain using the inverse DCT to reconstruct the stego-image.

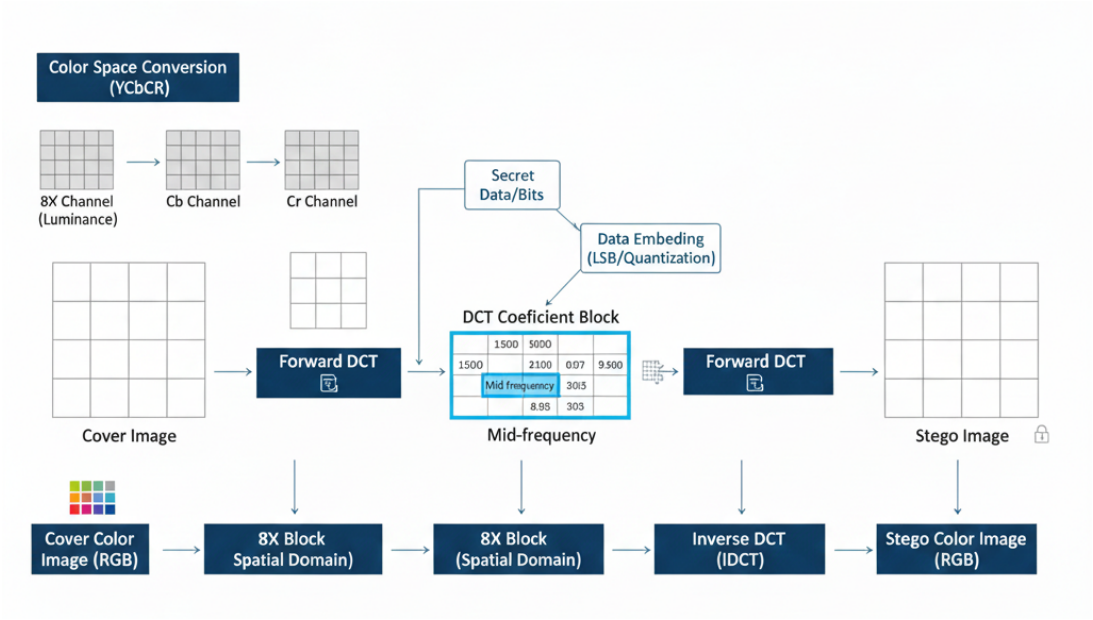


Figure 2.5: Block-based DCT embedding process illustrating transformation, coefficient selection, and inverse transformation.

Extraction Process The extraction operation follows the steps in the embedding operation as follows:

1. Conversion of the stego-image to the appropriate color space (from the RGB color space to the YCbCr color space).
2. Aplication of DCT
3. Identification of the same mid-frequency coefficients used during embedding.
4. Recovery of the hidden bits based on the coefficient modification rule.

Accurate extraction relies on preserving coefficient alignment and consistent block structure between embedding and extraction.

DCT Method Benefits

compared with the spatial domain method of steganography, DCT offers many benefits:

- Robustness against JPEG compression is significantly improved.
- and embedding information into the frequency domain allows for better imperceptibility than embedding the hidden message into the spatial domain.
- Compatibility with widely used image compression standards.
- DCT-based techniques are less vulnerable to simple noise additions.

Limitations and Challenges

While DCT-based steganography has significant advantages, many disadvantages exist.

- Have limited robustness when exposed to combined or severe geometric distortions such as cropping, resizing, and rotation.
- Sensitivity to block misalignment caused by image re-sampling.
- The trade-off between payload capacity and robustness is significant.

Although DCT techniques offer substantially increased robustness compared to spatial-domain approaches, DCT alone cannot be relied upon for successful recovery of data under highly adversarial or degradation conditions. [2].

2.2.3 Discrete Wavelet Transform (DWT)-Based Steganography

Embedding Data in Images Using DWT-Based Steganography The Discrete Wavelet Transform (DWT) is a popular method for representing images as a set of wavelet coefficients and DWT is used in image steganography because it is a transform-domain technique that allows for the representation of image information at various resolutions. In contrast to the Discrete Cosine Transform, which processes a fixed-size block of an image, the DWT processes an entire image at all frequency bands and all scales simultaneously, thereby providing a means of implementing image steganography with greater perceptual adaptability and resilience to localized distortions than other techniques [12, 13].

Overview of the DWT Decomposition

The DWT works by decomposing an image into a set of frequency sub-bands (subbands) by passing the entire image horizontally and vertically through pairs of low-pass and high-pass filters. After a single level DWT decomposition of an image, we end up with four distinct frequency subbands:

- **LL (Low-Low):** This is the approximation component, which contains the majority of an image's energy and the information regarding its structure.
- **LH (Low-High):** Captures horizontal edge information.
- **HL (High-Low):** Captures vertical edge information.
- **HH (High-High):** Represents diagonal details and fine textures.

The LL sub-band can be decomposed again using the same technique (that is recursion). Consequently, the image-represented data set can be represented in a multi-resolution, or hierarchical, manner.

Embedding Process in DWT-Based Steganography

The embedding procedure for DWT-based steganography generally consists of the following steps:

1. **Color Space Selection:** an image is first selected for steganography by choosing between either a color space or a luminance (brightness) component. Color images are generally embedded using the luminance component because it maintains a higher perceptual quality than using one of the color channels.
2. **Wavelet Decomposition:** DWT Decomposition After the luminance has been selected, a discrete wavelet transform (DWT) is applied to multiple levels (i.e., level 1, level 2, etc.) in order to obtain multiple subbands corresponding to each level of the decomposition.

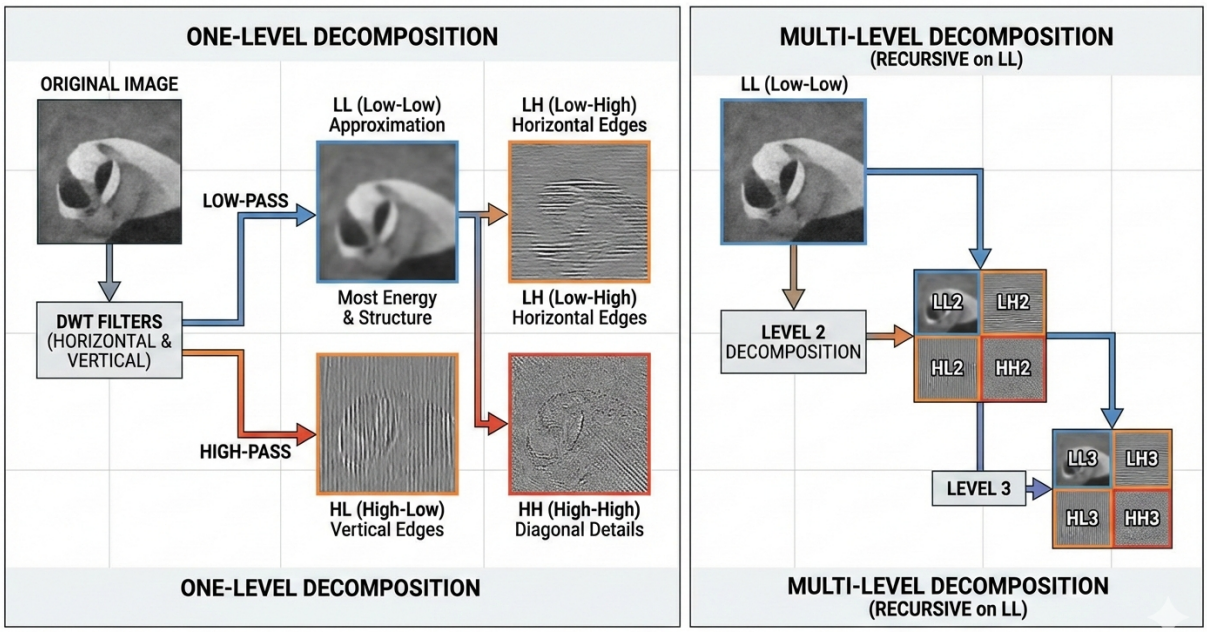


Figure X: One-level and multi-level DWT decomposition showing LL, LH, HL, and HH sub-bands.

Figure 2.6: One-level and multi-level DWT decomposition showing LL, LH, HL, and HH sub-bands.

3. **Sub-band Selection:** The subbands for embedding are generally chosen from the higher or mid-level frequency subbands (e.g., LH, HL, HH). The low-frequency subband, LL, may cause strong visual effects on the final output image.

4. **Coefficient Modification:** Modification of Wavelet Coefficients Wavelet coefficients are modified to embed secret data, which can be accomplished in various methods (e.g., coefficient quantization, adjusted threshold, or bit-level).

5. **Inverse DWT:** After modifying the coefficients of the appropriate subbands, an inverted DWT is performed, returning the steganographic image back to the spatial domain.

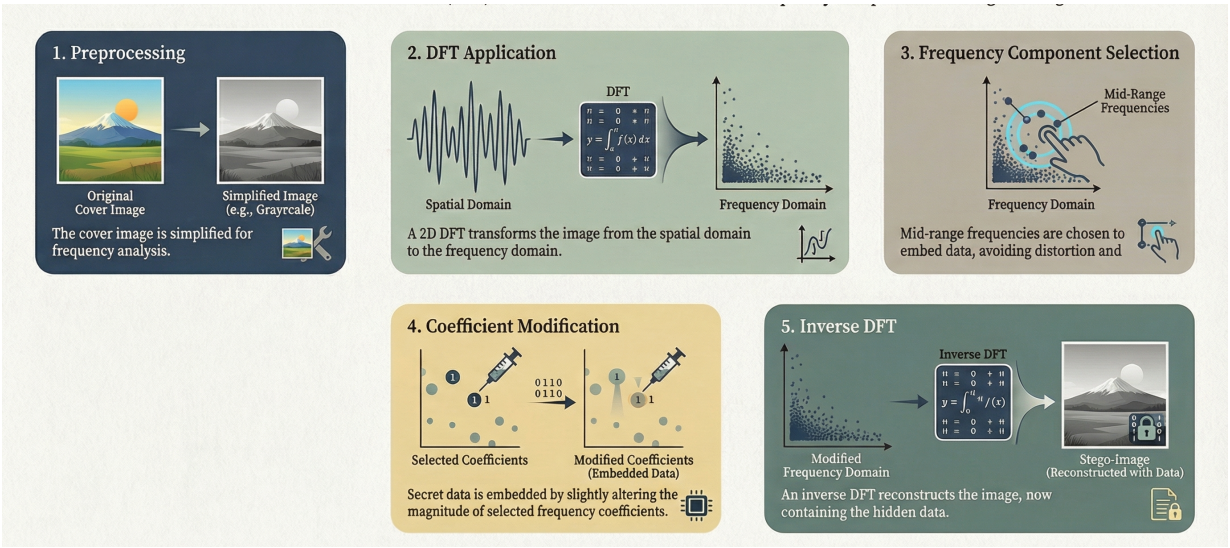


Figure 2.7: DFT-based embedding process illustrating frequency selection, magnitude modification, and inverse transformation

Extraction Process

The process of extracting information through DWT is the reverse of how it was added to the stego-image. The steps for extracting information from the DWT are as follows:

1. To use the same wavelet parameters, perform a DWT on the stego-image
2. Determine which sub-band(s) were used for embedding
3. Use the coefficient modification rule to determine how to retrieve embedded bit(s)
4. Reconstruct the original hidden message from the retrieved bits

The quality of extraction is dependent upon both the wavelet coefficient's stability and also the preservation of the sub-band structure after an image has been degraded.

Advantages of DWT-Based Steganography

DWT-based steganography offers several advantages:

- Multi-Resolution representation is consistent with the way our eyes work.
- Greater Robustness against localized distortions and the cropping of some parts of the image..
- Greater spatial frequency localization than other methods of embedding (block-based).
- The ability to choose where to embed information, depending on the requirements of the application.

Limitations and Challenges

Although DWT Steganography has many benefits, it has a variety of obstacles that may limit its utility. Examples of these limitations include

- The susceptibility of the system to geometric transformations (e.g., resizing/rotating).
- Much higher complexity than methods based on space.
- Potential susceptibility to a multitude of compression attacks or aggressive multi-staged attacks.
- Trade-offs between robustness, imperceptibility, and payload capacity.

These limitations show that DWT techniques provide more robust solutions to Spatial Domain techniques and in some cases may offer advantages over Block-Coding Transform based techniques, however the implementation of other means of ensuring the safety of data recovery may be needed in the event of extreme or compound degradation to the image. [1].

2.2.4 Discrete Fourier Transform (DFT)-Based Steganography

The Discrete Fourier Transform (DFT) is a signal processing technique that decomposes an entire image into its constituent frequency components, producing a global spectral representation of the whole image at once. Unlike the DCT, which divides the image into independent 8×8 blocks and processes each separately, the DFT operates on the full image as a single entity, so every frequency coefficient carries information about the entire spatial domain [16]. Unlike the DWT, which separates frequency and spatial information simultaneously through multi-resolution decomposition [12], the DFT provides a purely frequency-domain representation without any spatial localisation. This global nature is both the key advantage and the key limitation of DFT-based steganography: it confers partial invariance to geometric transformations such as rotation and translation, but it also means that a single spatial modification (such as cropping) affects every frequency coefficient in the spectrum [17, 2].

Overview of the DFT Representation

The two-dimensional DFT transforms a spatial-domain image $I(x, y)$ of dimensions $M \times N$ into a complex-valued frequency-domain matrix $F(u, v)$ defined as:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \cdot e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}, \tag{2.7}$$

where u and v are the horizontal and vertical frequency indices, and $j = \sqrt{-1}$. The inverse DFT reconstructs the spatial image from the spectrum:

$$I(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \cdot e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}. \tag{2.8}$$

Each complex coefficient $F(u, v)$ can be separated into two components that carry different types of image information:

$$F(u,v) = |F(u,v)| \cdot e^{j\,\phi(u,v)}, \tag{2.9}$$

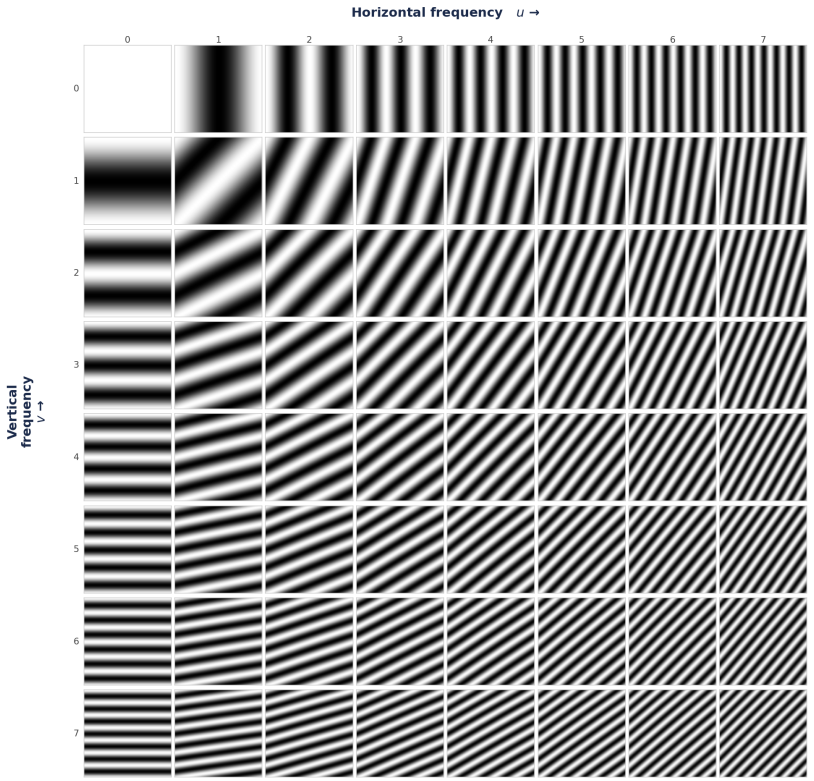


Figure 2.8: Mathematical visualization of 2D DFT basis functions showing sinusoidal patterns at various frequencies and orientations.

where $|F(u, v)|$ is the **magnitude spectrum** and $\phi(u, v)$ is the **phase spectrum**. The magnitude encodes the strength of each frequency component — how much energy the image contains at that frequency — while the phase encodes the spatial positions and structural relationships of image features [20]. Research has shown that the phase spectrum preserves the majority of perceptual information: an image reconstructed from phase alone is recognisable, while an image reconstructed from magnitude alone is not. This finding directly shapes the DFT embedding strategy: steganographic modifications are applied to the **magnitude spectrum only**, leaving the phase unchanged in order to preserve visual quality.

Like DCT and DWT, the DFT frequency plane is partitioned into three regions based on distance from the centre (DC component at $u = v = 0$ after centring):

- **Low-frequency region:** Near the centre. Encodes smooth intensity variations and the overall brightness of the image. Modifying this region produces clearly visible intensity shifts.
- **Mid-frequency region:** Intermediate distance from the centre. Corresponds to edges, object boundaries, and texture structure. This is the preferred embedding zone, offering a balance between imperceptibility and robustness [19].
- **High-frequency region:** Periphery of the spectrum. Captures fine textures and noise-like detail. Invisible to modify, but easily destroyed by low-pass filtering and JPEG compression.

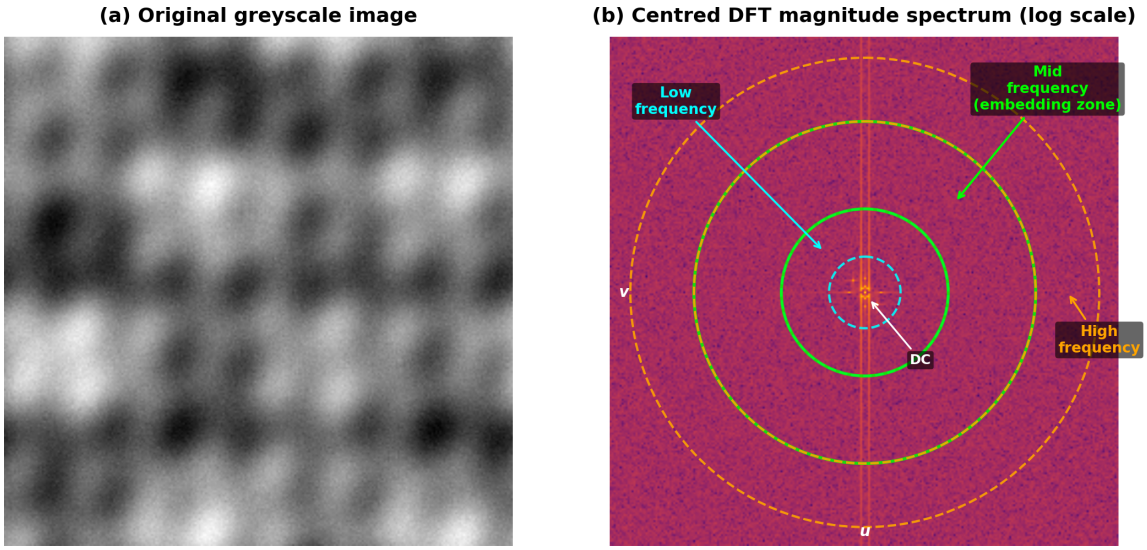


Figure 2.9: Centred 2D DFT magnitude spectrum of a natural image (log scale). The DC component appears as the bright spot at the centre. Low-frequency coefficients cluster near the centre, mid-frequency coefficients occupy the intermediate ring, and high-frequency coefficients are located at the periphery. Steganographic embedding targets the mid-frequency ring to balance robustness and imperceptibility.

Embedding Process in DFT-Based Steganography

The embedding procedure for DFT-based steganography follows a structured sequence of steps analogous to those used in DCT-based methods:

1. **Colour Space Conversion:** For colour images, the RGB image is converted to a luminance-chrominance colour space (typically YCbCr). Embedding is performed in the luminance (Y) channel because the human visual system is most sensitive to brightness variations, and the luminance channel carries the most structural information.
2. **Forward DFT:** The 2D DFT is computed for the entire luminance channel using the Fast Fourier Transform (FFT) algorithm, which reduces computational complexity from $\mathcal{O}(M^2N^2)$ to $\mathcal{O}(MN\log(MN))$ [24]. The spectrum is centred so that the DC component is at the middle of the matrix, making the frequency regions easier to identify and work with.
3. **Frequency Region Selection:** A set of mid-frequency coefficients is identified for embedding. These are typically defined by a radial band:

$$\mathcal{R} = \{(u, v) : r_{\min} \leq \sqrt{u^2 + v^2} \leq r_{\max}\}, \tag{2.10}$$

where $r_{\min}$ and $r_{\max}$ define the inner and outer radii of the embedding annulus. Choosing a circularly symmetric region provides partial robustness to rotation, because rotation in the

spatial domain simply rotates the spectrum without moving coefficients out of a concentric ring [18].

4. **Magnitude Modification:** Secret bits are embedded by modifying the magnitude of the selected coefficients while leaving the phase unchanged. A common technique is additive embedding:
$$|F'(u,v)| = |F(u,v)| + \alpha \cdot b \cdot |F(u,v)|, \tag{2.11}$$
where $\alpha \in [0.01, 0.1]$ is the embedding strength and $b \in \{0,1\}$ is the secret bit. The phase-preserved stego coefficient is then:
$$F'(u,v) = |F'(u,v)| \cdot e^{j\,\phi(u,v)}. \tag{2.12}$$
The conjugate symmetry constraint $F(M-u, N-v) = F^*(u,v)$ must also be maintained to ensure the inverse DFT produces a real-valued image.

5. **Inverse DFT:** The modified spectrum $F'(u,v)$ is transformed back to the spatial domain using the inverse DFT to produce the stego-image. Pixel values are rounded to integers and clipped to the valid intensity range $[0, 255]$ for 8-bit images.

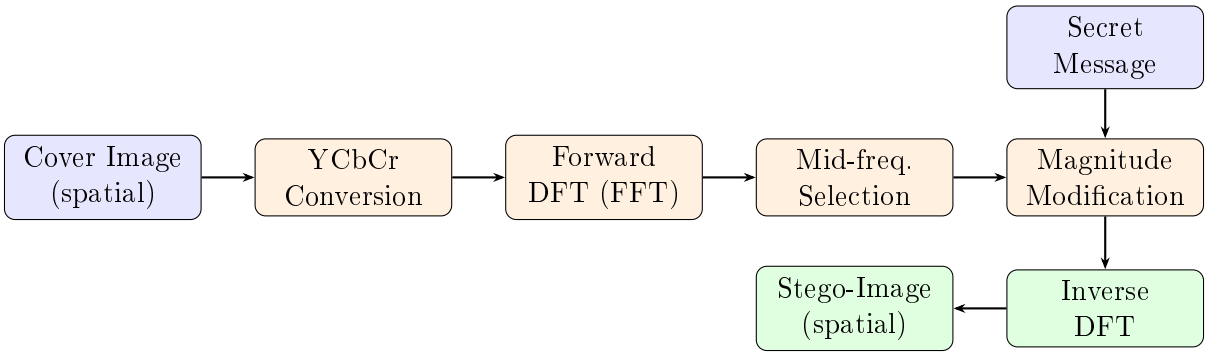


Figure 2.10: DFT-based embedding pipeline: the cover image is converted to YCbCr, the forward DFT is applied to the luminance channel, mid-frequency magnitude coefficients are modified according to the secret message bits, and the inverse DFT reconstructs the stego-image in the spatial domain.

Extraction Process

Extraction is the inverse of the embedding procedure and requires that the receiver know the same embedding parameters (region boundaries $r_{\min}$, $r_{\max}$ and strength α):

1. Convert the received stego-image to the YCbCr colour space and isolate the luminance channel.
2. Apply the forward DFT and centre the spectrum.
3. Locate the same mid-frequency region $\mathcal{R}$ used during embedding.
4. For each selected coefficient (u,v), decode the embedded bit by comparing the received magnitude $|\tilde{F}(u,v)|$ to a decision threshold derived from a reference magnitude (either the original cover magnitude if available, or an estimated reference):
$$\hat{b} = \begin{cases} 1 & \text{if } |\tilde{F}(u,v)| > |\bar{F}(u,v)|(1+\alpha/2), \\ 0 & \text{otherwise.} \end{cases} \tag{2.13}$$
5. Assemble the extracted bits to reconstruct the hidden message.

Reliable extraction depends on the magnitude relationships between coefficients being preserved after transmission. Any attack that alters the relative magnitude values — such as JPEG recompression or additive noise — will introduce extraction errors.

Geometric Invariance Properties

The DFT possesses two properties that give DFT-based steganography a natural robustness advantage over DCT and DWT methods against certain geometric distortions [17, 16]:

- **Translation invariance:** A spatial shift of the image by (x_0, y_0) only introduces a phase factor into each DFT coefficient; the magnitude spectrum is completely unchanged:

$$\mathrm{DFT}\{I(x - x_0, y - y_0)\} = F(u, v) \cdot e^{-j2\pi(ux_0/M + vy_0/N)}. \tag{2.14}$$

  Because embedding is in the magnitude, spatially translated stego-images can be extracted without any re-synchronisation.

- **Rotation correspondence:** A spatial rotation of the image by angle θ rotates the magnitude spectrum by the same angle θ. If embedding is placed in a circularly symmetric ring $\mathcal{R}$, the embedded coefficients remain in the same ring after rotation, so a rotation-invariant extractor can still locate them [18].

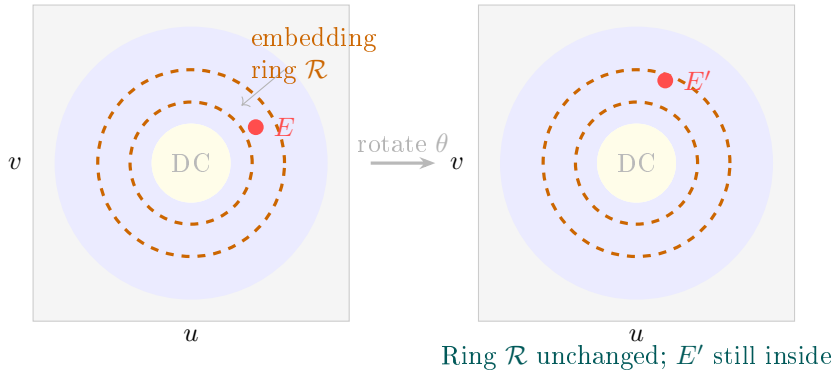


Figure 2.11: Rotation invariance of DFT magnitude-based embedding. Spatial rotation by angle θ rotates the entire magnitude spectrum by the same angle, moving the embedded coefficient from E to E'. Because the embedding ring $\mathcal{R}$ is circularly symmetric (dashed orange), E' remains inside $\mathcal{R}$ regardless of θ, allowing extraction without any geometric re-synchronisation [18].

Advantages of DFT-Based Steganography

DFT-based steganography offers several advantages that distinguish it from DCT and DWT methods:

- **Translation invariance:** Spatial shifts in the image do not affect the magnitude spectrum, so the embedded signal survives minor positional changes without any re-alignment [16].
- **Rotation robustness:** Embedding in circularly symmetric spectral regions provides robustness to image rotation, a property DCT and DWT methods do not possess [17, 18].
- **No blocking artefacts:** Because the DFT processes the whole image at once, it does not introduce the tile-boundary artefacts that can appear in DCT-based methods at low embedding strengths.
- **Global distribution:** Each frequency coefficient is influenced by every pixel, so modifications are distributed across the entire image rather than confined to local blocks, reducing the spatial footprint of the embedding.
- **Compatibility with spread-spectrum techniques:** DFT coefficients can serve directly as the carrier for pseudo-random spreading sequences, combining the geometric invariance of the DFT with the noise robustness of spread-spectrum embedding [19].

Limitations and Challenges

Despite its geometric robustness advantages, DFT-based steganography faces important limitations:

- **Cropping vulnerability:** Cropping removes pixels and changes the image dimensions, destroying the global periodicity assumption on which the DFT is based. All frequency coefficients are affected, making DFT-based embeddings highly sensitive to spatial cropping [2].
- **Sensitivity to JPEG compression:** JPEG operates in the DCT domain, and JPEG recompression alters DFT coefficients in a complex, block-dependent manner. DFT-based embeddings are therefore less JPEG-resistant than DCT mid-frequency embeddings designed specifically to survive quantisation [1].
- **Conjugate symmetry constraint:** The DFT of a real image satisfies $F(M-u, N-v) = F^*(u, v)$. Any modification must respect this constraint, which effectively halves the number of independently usable embedding positions and reduces the practical embedding capacity.
- **Lower capacity than DCT:** Because only mid-frequency coefficients in a single global spectrum are available, DFT methods generally support lower payload rates than DCT methods, which exploit every block's coefficient set independently [9].
- **Computational cost:** Although the FFT reduces complexity to $\mathcal{O}(MN \log MN)$, processing the full image spectrum is still more expensive per bit embedded than block-based DCT processing for large images [24].

These limitations indicate that DFT-based steganography is best suited to applications where rotation and translation robustness are primary requirements, and where crop-resistance is either not needed or is provided by a complementary mechanism such as the fragmentation and distributed embedding strategy proposed in this thesis [17, 1].e trade-offs inherent in DFT-based steganography.

2.3 Robustness-Oriented Approaches

Achieving robustness in image steganography requires more than choosing a stable transform domain. It demands a system-level design philosophy that anticipates degradation, structures the payload to survive partial loss, and recovers meaningful information even under adverse conditions. This section presents a comprehensive review of the principal strategies that have been developed toward this goal, including error correction coding, synchronization mechanisms, adaptive embedding, spread-spectrum techniques, fragmentation and redundancy, and the role of deep learning.

2.3.1 Error Correction Coding for Steganographic Robustness

A widely adopted strategy for improving robustness against signal-processing attacks is the integration of error correction codes (ECC) into the steganographic payload. Because operations such as JPEG compression, additive noise, and low-pass filtering introduce random bit-level errors into the recovered message, ECC allows the receiver to detect and correct a bounded number of corrupted bits without any knowledge of the original image [1].

Block codes. Reed–Solomon (RS) codes are particularly popular in steganographic applications because they operate on multi-bit symbols rather than individual bits, making them well-suited to burst errors introduced by block-based compression [19]. An RS(n, k) code encodes k information symbols into n codeword symbols, enabling the correction of up to $\lfloor (n-k)/2 \rfloor$ erroneous symbols. BCH codes offer a similar correction capability and have been used where binary codewords are preferable [2].

Convolutional and turbo codes. Convolutional codes introduce memory into the encoding process and are decoded efficiently with the Viterbi algorithm, making them suitable for steganographic channels that exhibit correlated noise patterns [7]. Turbo codes, which concatenate two recursive convolutional encoders separated by an interleaver, approach the Shannon capacity limit and have been applied in watermarking systems that demand near-optimal error recovery [19].

LDPC codes. Low-density parity-check (LDPC) codes have attracted growing interest in robust steganography because their iterative belief-propagation decoding is highly effective even at low signal-to-noise ratios [11]. Compared with turbo codes, LDPC codes offer lower decoding complexity for long codewords and are therefore better suited to high-resolution image payloads.

Limitations of ECC alone. Despite their effectiveness against random bit corruption, ECC schemes share a fundamental assumption: the decoder must receive the codeword bits *in the correct order*. Geometric attacks such as cropping, rotation, and resizing destroy positional correspondence between the embedded and extracted data, producing not random errors but *burst erasures* at unknown locations [1]. Under these conditions, even maximum-distance-separable codes fail because the error positions cannot be identified from the received sequence alone. This observation motivates the use of complementary mechanisms, discussed in the following subsections, that address synchronization rather than bit-level correctness.

2.3.2 The Synchronization Problem

Synchronization loss is arguably the most critical and least resolved challenge in robust steganography [17]. A steganographic system is said to be synchronized when the extractor can reliably locate the embedding positions within the received image. Most classical schemes achieve this by relying on fixed coordinate systems, block-partition boundaries, or predetermined pixel sets. Geometric transformations invalidate these references, causing the extractor to read pixels at incorrect positions—a failure mode qualitatively different from ordinary bit corruption.

Causes and Manifestations

- **Cropping** removes entire image regions, permanently destroying any fragments embedded within them. Even when the remaining image content is intact, the extractor can no longer determine the original image boundaries and therefore cannot reconstruct the coordinate grid used during embedding [2].
- **Resizing and resampling** change the spatial resolution through interpolation, shifting pixel positions and distorting block boundaries. Because DCT-based schemes depend on exact 8×8 block alignment, even a one-pixel shift causes the extractor to operate on misaligned blocks, producing a catastrophic increase in bit error rate [1].
- **Rotation** introduces a global angular offset. Unless the embedding domain is rotation-invariant, the extractor cannot identify coefficient positions and the extraction fails entirely [17].
- **Compound attacks** combine several of the above operations sequentially. Social-media platforms, for example, typically resize images upon upload, apply lossy JPEG compression, and sometimes crop to fixed aspect ratios—all in a single automated pipeline [9]. Each step accumulates synchronization error, making recovery increasingly difficult.

Template-Based Synchronization

One practical solution is to embed a known geometric reference pattern—often called a *synchronization template* or *registration mark*—alongside the secret payload. The receiver detects the template, estimates the geometric transformation applied to the image, and inverts it before extraction [17].

Template-based methods are effective against moderate rotations and uniform scaling, but they consume part of the embedding capacity and are vulnerable to cropping if the template region is removed.

Kutter [29] demonstrated that circularly symmetric patterns embedded in the Fourier domain are particularly robust for template-based registration, as they remain detectable after rotation and scaling. Similarly, Solachidis and Pitas [30] proposed embedding ring-shaped patterns in the 2-D DFT magnitude spectrum, exploiting the Fourier shift theorem to achieve translation invariance.

Feature-Based Alignment

An alternative approach avoids fixed coordinate systems entirely by anchoring embedding positions to stable image features—corners, edges, or salient regions—that can be reliably re-detected after geometric distortion [17]. Feature detectors such as SIFT (Scale-Invariant Feature Transform) have been adopted in watermarking to compute a canonical coordinate frame from the image content itself, enabling extraction without prior knowledge of the applied transformation [2, 31]. The primary limitation is that severe cropping or blurring may destroy the features used for alignment, rendering the approach ineffective under compound attacks.

Invariant-Domain Embedding

A more fundamental solution embeds data directly in a representation that is algebraically invariant to the class of expected transformations. DFT magnitude coefficients, for instance, are translation-invariant because spatial shifts map to phase changes without affecting magnitudes [20]. Embedding in the log-polar Fourier domain achieves additional invariance to rotation and scaling, a property exploited by several watermarking systems [17, 18]. Radial harmonic transforms and Zernike moments have also been explored as rotation-invariant embedding domains, though their applicability to steganography is limited by their lower embedding capacity compared with block-based transforms [2].

2.3.3 Spread-Spectrum Steganography

Spread-spectrum (SS) techniques originate from secure communications and were adapted for image steganography by Marvel et al. [19], whose work remains a foundational reference in robust data hiding. The core idea is to distribute each secret bit across a large number of cover elements (pixels or transform coefficients) using a pseudo-random spreading sequence, so that the energy of the hidden signal is spread too thinly to be concentrated by any localized attack.

Formally, let $\mathbf{m} \in \{-1, +1\}^K$ be the message after bipolar encoding, and let $\mathbf{w}_k \in \mathbb{R}^N$ be a unit-norm pseudo-random spreading vector for the k-th bit, drawn from a secret key. The stego-signal added to the cover image $\mathbf{c}$ is:

$$\mathbf{s} = \mathbf{c} + \alpha \sum_{k=1}^{K} m_k \, \mathbf{w}_k, \tag{2.15}$$

where $\alpha > 0$ is an embedding strength parameter that trades imperceptibility against robustness [19, 31]. Extraction correlates the received signal against each spreading vector:

$$\hat{m}_k = \operatorname{sign}\!\left(\mathbf{r}^\top \mathbf{w}_k\right), \tag{2.16}$$

where $\mathbf{r}$ is the received (possibly degraded) signal. Because the spreading vectors are orthogonal—or nearly so for long sequences—inter-symbol interference is minimal, and the correlation effectively averages out additive noise.

Informed embedding and dirty-paper coding. A significant theoretical advance over basic SS steganography came with the introduction of *informed embedding*, in which the embedding process exploits knowledge of the cover signal to minimize distortion for a given robustness level [26]. Quantization Index Modulation (QIM), proposed by Chen and Wornell [26], selects a quantization lattice based on the secret bit and quantizes the coefficient to the nearest point on that lattice. QIM achieves the *writing on dirty paper* capacity, meaning it can approach the theoretical maximum robustness for a given distortion budget, and it is now widely used in DCT-domain watermarking [27, 7].

Limitations of spread-spectrum methods. Although SS techniques are highly robust against additive noise and moderate compression, they share the synchronization vulnerability of other location-dependent schemes: geometric attacks that shift the spreading vector grid cause the correlation in Equation (2.16) to collapse [1]. Moreover, the per-bit capacity of SS steganography is low, as each bit requires $N \gg 1$ coefficients for reliable spreading, limiting the practical payload size.

2.3.4 Adaptive Embedding and Perceptual Masking

Adaptive embedding methods adjust the embedding strength locally based on image content, exploiting perceptual masking properties of the human visual system (HVS) to embed more information where distortion is less visible and less where it is more noticeable [25]. This strategy simultaneously improves imperceptibility and, in some implementations, robustness, because stronger embedding in textured or edge-rich regions is harder for processing operations to fully suppress.

Texture and Edge Masking

The HVS is less sensitive to luminance changes in regions of high spatial activity— such as textured surfaces, edges, and complex patterns—than in flat, homogeneous areas [6]. Adaptive steganographic methods leverage this by computing a local activity measure $\sigma^2(b)$ for each image block b (e.g., the variance of DCT AC coefficients) and scaling the embedding strength accordingly:

$$\alpha(b) = \alpha_0 \cdot f\big(\sigma^2(b)\big)\,, \tag{2.17}$$

where $f(\cdot)$ is a monotonically increasing function and α_0 is a base strength parameter [25, 31]. Blocks with higher texture activity receive stronger embedding, increasing local robustness in those regions while limiting visible distortion in smooth areas.

Frequency-Selective Masking

Complementary to spatial masking is frequency-domain masking, in which the embedding energy is concentrated in mid-frequency DCT or DWT coefficients that fall within the HVS sensitivity range [27]. The JPEG quantization matrix provides a useful proxy for perceptual importance: coefficients assigned a larger quantization step are less perceptually significant and can therefore tolerate stronger embedding [2]. By aligning embedding choices with the JPEG quantization structure, adaptive DCT methods also gain a secondary benefit: the embedded signal is less likely to be removed by JPEG compression, because only the coefficients that survive compression at the target quality factor are used for embedding.

Just-Noticeable Difference Models

More principled approaches model the maximum imperceptible modification at each coefficient using just-noticeable difference (JND) thresholds derived from psychophysical experiments [1]. A JND model $J(u, v)$ specifies the maximum change to coefficient $F(u, v)$ that remains below human detection threshold. Embedding is then constrained so that $|\Delta F(u, v)| \leq J(u, v)$, guaranteeing imperceptibility by construction. Podilchuk and Zeng [25] demonstrated that JND-constrained watermarking

achieves significantly better imperceptibility-robustness trade-offs than fixed-strength embedding, and their framework has been adapted for steganographic applications.

2.3.5 Fragmentation, Redundancy, and Distributed Embedding

A conceptually distinct approach to robustness abandons the goal of protecting individual embedding positions and instead focuses on *payload survivability* through structured redundancy and spatial distribution. Rather than assuming that the entire image is preserved, these methods acknowledge that portions of the cover may be destroyed and design the embedding strategy to permit recovery from whatever fragments remain [2, 31].

Redundant Embedding

The simplest instantiation of this idea is to embed the same payload multiple times across non-overlapping image regions [11]. If the image is divided into R non-overlapping regions and each carries a full copy of the payload, then recovery is possible as long as at least one region survives intact. While effective against cropping of a single contiguous area, naive redundant embedding does not scale well with payload size, as the effective capacity is reduced by a factor of R.

Erasure-Correcting Codes and Fountain Codes

A more capacity-efficient approach uses *erasure codes*, which are designed for channels that lose entire blocks rather than individual bits. Reed–Solomon codes can be used in erasure mode: an RS(n, k) code can recover the k source symbols from *any* k of the n transmitted symbols, regardless of which $n-k$ are erased [7]. Fountain codes—including LT codes and Raptor codes—go further by generating a potentially unlimited number of encoded symbols such that the original message can be recovered from any sufficiently large subset [7]. Applying fountain codes to image steganography means that recovery is possible from any portion of the stego-image that survives cropping or localized tampering, provided the surviving region contains enough encoded fragments.

Inter-Fragment Dependency Graphs

Beyond simple redundancy, structured payload representations can encode cross-fragment relationships so that surviving fragments carry partial information about missing ones. This transforms the reconstruction problem into a belief-propagation or message-passing problem on a factor graph, where each received fragment provides constraints on the values of neighboring fragments [1]. The design of the dependency graph—its degree distribution, connectivity, and redundancy level—directly controls the trade-off between capacity overhead and recovery probability under various patterns of fragment loss.

Multi-Domain and Multi-Channel Distribution

Embedding fragments across different transform domains provides complementary robustness: spatial-domain fragments survive geometric distortions better than frequency-domain fragments, while DCT-domain fragments are more robust to compression than pixel-level modifications [32, 2]opies across the luminance and chrominance channels of a color image exploits the fact that different channels are affected differently by common processing operations [27]. A combined multi-domain, multi-channel strategy therefore reduces the probability that all copies of any given fragment are simultaneously destroyed.

2.3.6 Robust Steganography vs. Digital Watermarking

Robust steganography and digital watermarking share a common toolbox—transform-domain embedding, ECC, spread spectrum, adaptive strength—but differ fundamentally in their objectives,

threat models, and success criteria [31, 33].

Digital watermarking is designed for *ownership verification* and *content authentication*: the embedded mark is expected to be detectable by an authorized verifier, and its persistence under attack is the primary success criterion. The presence of a watermark is not concealed; on the contrary, watermarking systems often include public detection algorithms that any party can use to verify authenticity [2]. Consequently, watermarking schemes can afford to use stronger, more visible embedding, and they are evaluated primarily on detectability under attack (measured by Normalized Correlation or bit detection rate) rather than on undetectability.

Steganography, by contrast, requires *covert communication*: the existence of the hidden message must remain undetectable to unauthorized observers, even sophisticated steganalysts [10, 11]. This imposes a fundamentally different constraint on the embedding process. Techniques that enhance robustness—redundancy, high embedding strength, structured payload distribution—tend to introduce statistical artifacts that increase the risk of steganalytic detection [9]. The steganographer must therefore balance three competing objectives simultaneously: imperceptibility, robustness, and undetectability (resistance to steganalysis), whereas a watermarker need only balance the first two.

Robust steganography occupies the intersection of these paradigms, inheriting the covertness requirement of classical steganography and the survivability requirement of watermarking. This dual constraint is what makes robust steganographic design particularly challenging and motivates the payload-oriented framework proposed in Chapter 3.

2.3.7 Steganalysis and the Detectability–Robustness Trade-off

A complete treatment of robustness must acknowledge the threat of steganalysis—the statistical detection of hidden communication—because robustness-enhancing design choices often increase detectability [9, 2].

Classical Steganalysis

Early steganalysis methods focused on statistical anomalies introduced by specific embedding algorithms. Chi-squared analysis [10] detects the pairing of adjacent pixel values caused by LSB substitution. RS analysis examines the ratio of regular and singular pixel groups, which is disrupted predictably by LSB embedding [10]. These attacks are *specific*: they exploit known properties of a single embedding algorithm and fail against others.

Universal Feature-Based Steganalysis

Modern steganalysis is *universal*: it extracts a high-dimensional feature vector from the image and trains a machine learning classifier to distinguish cover images from stego-images, without requiring knowledge of the specific embedding algorithm [9]. The rich model of Fridrich and Kodovský [9] computes co-occurrence statistics of pixel prediction errors in multiple directions, producing feature vectors with tens of thousands of dimensions that capture subtle embedding artifacts across a wide range of steganographic methods. Paired with ensemble classifiers, these rich feature sets achieve near-perfect detection even at very low embedding rates, highlighting the difficulty of simultaneously achieving robustness and undetectability.

Implications for Robust Steganographic Design

The detectability–robustness trade-off imposes a practical ceiling on the embedding strength that can be used in a steganographic (as opposed to watermarking) context. Stronger embedding improves survivability under attack but leaves larger statistical footprints in the stego-image. Several studies have shown that the embedding strength required for robustness against JPEG compression at quality factor 70 already produces detectable artifacts in standard steganalytic feature spaces [9, 2]. This motivates design approaches that achieve robustness through *structure*— fragmentation,

cross-fragment relationships, and domain diversity—rather than through raw embedding strength, as structural approaches can maintain a lower per-coefficient distortion while still enabling recovery under partial data loss.

2.3.8 Deep Learning–Based Robust Steganography

Deep learning approaches to steganography — including encoder–decoder architectures, generative adversarial frameworks, and hybrid transform-network methods — have emerged as a significant research direction since 2018. A comprehensive review of these methods, their robustness properties, and the reasons this thesis adopts a classical framework instead is provided in Section 2.6.

2.3.9 Limitations of Current Approaches

The techniques reviewed in this section reveal a consistent pattern: existing robustness strategies address either *bit-level corruption* (through ECC and spread spectrum) or *limited geometric distortion* (through synchronization templates and invariant domains), but few systems are designed to tolerate the *compound* of geometric desynchronization, partial data erasure, and signal-level noise that characterizes real-world uncontrolled environments.

Table 2.1 summarizes the strengths and limitations of each major category of robustness-oriented approach.

Table 2.1: Comparison of robustness-oriented steganographic approaches.

Approach	**Signal attacks**	**Geometric attacks**	**Partial recovery**	**Capacity**
LSB (spatial)	Poor	Poor	No	High
DCT mid-freq.	Moderate	Poor	No	Medium
DWT sub-band	Moderate	Moderate	No	Medium
DFT magnitude	Moderate	Good	No	Low
ECC alone	Good	Poor	No	Reduced
Spread spectrum	Good	Poor	No	Low
Template sync.	Moderate	Moderate	No	Reduced
Redundant embedding	Moderate	Moderate	Partial	Low
Fragmentation + erasure	Good	Good	Yes	Configurable
Deep learning	Good	Variable	No	Medium

The critical observation from Table 2.1 is that no existing single technique achieves both good robustness against compound attacks *and* the ability to partially reconstruct the hidden message when some data is lost. This gap—between robustness as the survival of embedding locations and robustness as the survival of *meaning*—motivates the self-recovering framework proposed in Chapter 3. By combining structured payload fragmentation, multi-domain distribution, erasure-code redundancy, and damage-aware extraction, the proposed system is designed to maximize the probability of recovering usable information from whatever portion of the stego-image survives, without requiring perfect synchronization or complete data preservation.

2.4 Attacks and Image Degradation Models

In real-world deployment scenarios, stego-images are rarely transmitted or stored under ideal conditions. Digital images routinely pass through social media platforms, messaging applications, cloud

storage services, and format conversion pipelines, each of which applies one or more processing operations that modify the image signal. These operations constitute a threat model for steganographic systems: they are attacks in the sense that they may destroy or degrade the embedded payload, regardless of whether the disruption is intentional. A robust steganographic system must be designed and evaluated against a realistic and formally specified threat model that reflects the conditions of its intended deployment [2, 1].

This section presents a comprehensive formal treatment of image attacks relevant to robust steganography. Each attack is characterised by a mathematical degradation model, its effect on different embedding domains, and the specific failure mode it introduces in steganographic extraction. The section is organised as follows. Section 2.4.1 covers signal processing attacks, which modify pixel intensity values while preserving spatial geometry. Section 2.4.2 covers geometric attacks, which alter the spatial structure of the image and introduce synchronisation loss. Section 2.4.3 addresses compound attacks, which combine multiple operations sequentially and represent the most realistic and challenging threat model. Section 2.4.4 addresses steganalytic attacks as a distinct threat class. Finally, Section 2.4.5 provides a unified comparative analysis.

2.4.1 Signal Processing Attacks

Signal processing attacks modify the intensity values of image pixels while preserving the overall geometric structure of the image — pixel positions, spatial dimensions, and block boundaries remain intact. The damage they introduce is therefore primarily at the signal level: embedded bits are corrupted by quantisation, noise, or spectral alteration, but the extractor can still locate the embedding positions. The appropriate countermeasure is error correction coding, which can recover a payload from a moderate fraction of corrupted bits provided that synchronisation is maintained.

Lossy Compression: JPEG

JPEG compression is the most pervasive and damaging signal processing attack in practice [14]. It is applied automatically by virtually every consumer image platform, social media service, and messaging application. JPEG operates in the DCT domain and introduces distortion through coefficient quantisation, which selectively discards information from high-frequency and mid-frequency spectral components.

Formal degradation model. JPEG compression proceeds in three stages. The image is first divided into non-overlapping 8×8 pixel blocks. Each block is transformed to the DCT domain, yielding a coefficient matrix $F(u, v)$. Each coefficient is then quantised according to a quality-dependent quantisation table:

$$\tilde{F}(u, v) = Q(u, v) \cdot \left\lfloor \frac{F(u, v)}{Q(u, v)} + 0.5 \right\rfloor, \tag{2.18}$$

where $Q(u, v)$ is the quantisation step size for frequency (u, v), determined by the JPEG quality factor $\mathrm{QF} \in [1, 100]$. Lower quality factors correspond to larger $Q(u, v)$, meaning coarser quantisation and greater information loss. The reconstruction error introduced by quantisation is:

$$\Delta F(u, v) = \tilde{F}(u, v) - F(u, v), \quad |\Delta F(u, v)| \leq \frac{Q(u, v)}{2}. \tag{2.19}$$

The expected distortion introduced across all coefficients of an image of dimensions $M \times N$ can be expressed as:

$$\mathrm{MSE}_{\mathrm{JPEG}} \approx \frac{1}{MN} \sum_{b} \sum_{u=0}^{7} \sum_{v=0}^{7} \frac{Q_b(u, v)^2}{12}, \tag{2.20}$$

where the outer sum is over all image blocks b and the factor of 12 arises from the variance of a uniform quantisation error [6]. This expression shows that JPEG distortion is heterogeneous across the spectrum: high-frequency coefficients, which have large $Q(u, v)$, suffer greater distortion than low-frequency coefficients.

Impact on steganographic embedding. The effect of JPEG on an embedded steganographic signal depends critically on where in the DCT spectrum the signal was placed:

- **High-frequency embedding** (e.g. LSB-based spatial methods, which map to high-frequency DCT coefficients after transformation): the quantisation step $Q(u, v)$ at high frequencies is typically 10–50× larger than at low frequencies, causing the embedded signal to be entirely overwritten. The BER under JPEG QF = 70 for unprotected spatial LSB embedding typically exceeds 40% [10].
- **Mid-frequency embedding** (the target of DCT-based steganography): embedding in coefficients with moderate $Q(u, v)$ provides substantially better survival rates, but requires that the embedding strength α satisfies $\alpha > Q(u, v)/2$ to ensure the embedded modification is not erased by quantisation. This defines a minimum embedding strength threshold for JPEG robustness.
- **Low-frequency embedding**: the lowest-frequency coefficients have $Q(u, v) = 1$ or 2 and survive even aggressive compression, but modifying them introduces perceptible intensity shifts.

JPEG and block boundary alignment. A secondary effect of JPEG compression is that it reinforces the 8×8 block grid. If the stego-image is cropped or resized before or after JPEG compression, the block boundaries may shift, misaligning the extractor's block grid with the one used during embedding. This interaction between JPEG and geometric attacks is a critical source of compound degradation discussed further in Section 2.4.3.

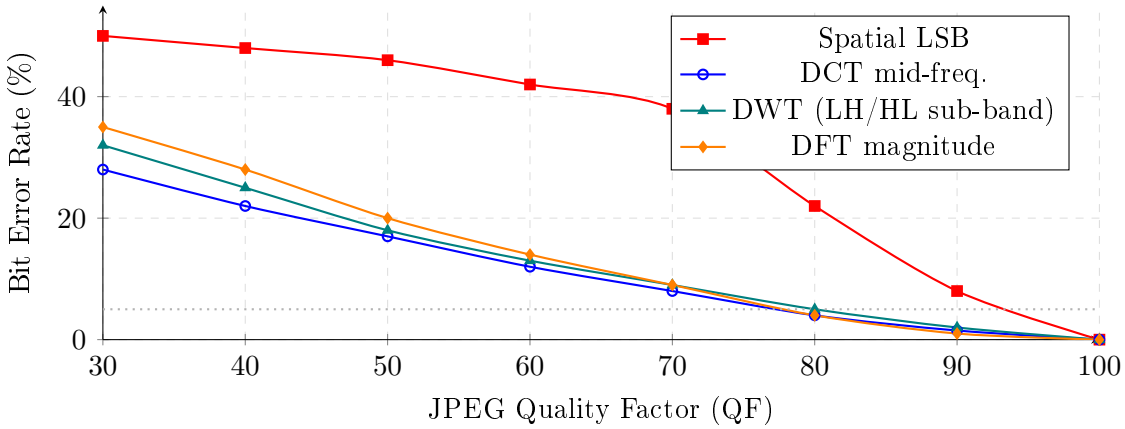


Figure 2.12: Illustrative bit error rate (BER) as a function of JPEG quality factor for four embedding domains. Spatial LSB embedding is almost entirely destroyed below QF = 90, while transform-domain methods show substantially better survival. The dotted line marks the 5% BER threshold below which error correction coding can typically achieve full recovery. Curves are representative of trends reported in the literature [1, 2].

Additive Noise

Additive noise arises from sensor imperfections during re-capture, analogue-to-digital conversion errors, or deliberate noise injection by an adversary attempting to destroy hidden content. The most common model is additive white Gaussian noise (AWGN) [6].

Formal degradation model. Let $I(x, y)$ denote the pixel intensity at location (x, y) in the stego-image. Under AWGN, the received signal is:

$$\tilde{I}(x, y) = I(x, y) + \eta(x, y), \quad \eta(x, y) \sim \mathcal{N}(0, \sigma_n^2), \tag{2.21}$$

where σ_n^2 is the noise variance and individual noise samples $\eta(x, y)$ are independent and identically distributed. The signal-to-noise ratio of the stego-image under AWGN is:

$$\text{SNR} = 10 \log_{10}\left(\frac{\sigma_I^2}{\sigma_n^2}\right) \quad [\text{dB}], \tag{2.22}$$

where σ_I^2 is the variance of the image signal. In the frequency domain, AWGN distributes energy uniformly across all spatial frequencies, affecting both low- and high-frequency components. This is qualitatively different from JPEG compression, which concentrates distortion in high-frequency coefficients.

Impact on steganographic embedding. The effect of AWGN on an embedded steganographic signal of amplitude α depends on the embedding domain. For spread-spectrum steganography, where each bit is spread across N coefficients, the signal-to-noise ratio at the decoder is:

$$\text{SNR}_{\text{dec}} = \frac{\alpha^2 N}{\sigma_n^2}, \tag{2.23}$$

showing that the spreading gain N provides a factor-of-N improvement in noise robustness relative to single-coefficient embedding [19]. For DCT-domain embedding at mid-frequencies, AWGN introduces errors only when $|\eta(x, y)|$ is large enough, after transformation, to flip the decision boundary used in extraction.

Salt-and-pepper noise, an alternative model in which a fraction p of pixels are set to the maximum or minimum intensity value, is more destructive than AWGN per unit distortion because it produces extreme outliers that cannot be corrected by averaging-based spreading.

Filtering Attacks

Filtering attacks apply a linear or non-linear convolution kernel to the image, altering the spatial frequency content. Common instances include Gaussian blurring, median filtering, and sharpening filters [6].

Formal degradation model. A linear low-pass filter applies a convolution:

$$\tilde{I}(x, y) = I(x, y) * h(x, y) = \sum_m \sum_n I(x - m,\, y - n)\, h(m, n), \tag{2.24}$$

where $h(x, y)$ is the filter kernel. The Gaussian blur kernel is:

$$h_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \tag{2.25}$$

with standard deviation σ controlling the degree of blurring. In the frequency domain, convolution becomes multiplication:

$$\tilde{F}(u, v) = F(u, v) \cdot H(u, v), \tag{2.26}$$

where $H(u, v)$ is the Fourier transform of $h(x, y)$. For a Gaussian kernel, $H(u, v)$ is itself a Gaussian centred at the origin, decaying to near zero at high frequencies. The attenuation factor $H(u, v)$ for a Gaussian blur with $\sigma = 1.5$ already reduces mid-frequency components by 30–60%, which is sufficient to destroy embeddings of moderate strength.

Impact on steganographic embedding. Low-pass filtering is particularly damaging to embeddings in high and mid-frequency DCT or DWT coefficients, because the filter directly suppresses the spectral region used for embedding. The fraction of the embedded signal that survives filtering is $H(u_0, v_0)$ at the embedding frequency (u_0, v_0). For a Gaussian blur with $\sigma = 2.0$, high-frequency DCT coefficients ($u+v \geq 10$) are attenuated by more than 90%, making unprotected high-frequency embedding essentially non-survivable. Median filtering is additionally non-linear and cannot be described by a single attenuation factor; it is particularly effective at destroying LSB-level spatial embeddings.

Colour Space Conversion and Format Conversion

Social media platforms frequently convert uploaded images between colour spaces (RGB to YCbCr, RGB to greyscale) or between formats (JPEG to WebP, PNG to JPEG). Colour space conversion alters the relationship between colour channels, potentially redistributing energy between the luminance and chrominance components and disrupting embeddings that rely on specific channel statistics [1]. Format conversion compounds this with the quantisation effects of the target codec. A full print-scan cycle — printing a digital image and re-capturing it with a camera — introduces all of the above simultaneously: blur, noise, perspective distortion, and colour shift [4].

2.4.2 Geometric Attacks

Geometric attacks alter the spatial arrangement of pixels rather than their intensity values. They are qualitatively more destructive than signal processing attacks for most steganographic systems because they introduce *synchronisation loss*: the extractor can no longer locate the embedding positions even when the embedded data itself is largely intact [17, 1]. Error correction coding cannot recover from synchronisation loss because the codeword bits are read in the wrong order or from the wrong positions, producing not random errors but structured misalignment that looks identical to complete payload destruction.

Cropping

Cropping removes a contiguous rectangular region from the image, permanently deleting all pixels — and all embedded data — within that region.

Formal degradation model. Let the original stego-image have dimensions $M \times N$. A crop operation retains a sub-image starting at offset (r_0, c_0) with dimensions $M' \times N'$, where $M' < M$ or $N' < N$:

$$\tilde{I}(x, y) = I(x + r_0,\, y + c_0), \quad 0 \leq x < M',\ 0 \leq y < N'. \tag{2.27}$$

The fraction of the image area that is destroyed is:

$$\rho_{\text{crop}} = 1 - \frac{M' \cdot N'}{M \cdot N}. \tag{2.28}$$

For a steganographic system that embeds N_F fragments uniformly distributed across the image, the expected number of fragments destroyed by a crop of fraction ρ_{crop} is:

$$\mathbb{E}[\text{fragments lost}] = N_F \cdot \rho_{\text{crop}}, \tag{2.29}$$

assuming uniform spatial distribution of fragments. This is the key quantity that the proposed self-recovering framework is designed to minimise by maximising spatial redundancy and distribution of embedded fragments across non-contiguous image regions.

Synchronisation effect. Beyond the direct data loss of Equation (2.27), cropping destroys the coordinate reference frame used during embedding. Block-based extractors (DCT, DWT) depend on knowing the exact position of the top-left corner of the image to reconstruct the block grid. After cropping, the origin shifts to (r_0, c_0), misaligning every block by the crop offset. Even a single-pixel crop offset causes every 8×8 DCT block to be misaligned, producing a near-random BER. This effect is illustrated conceptually in Figure 2.13.

Copy

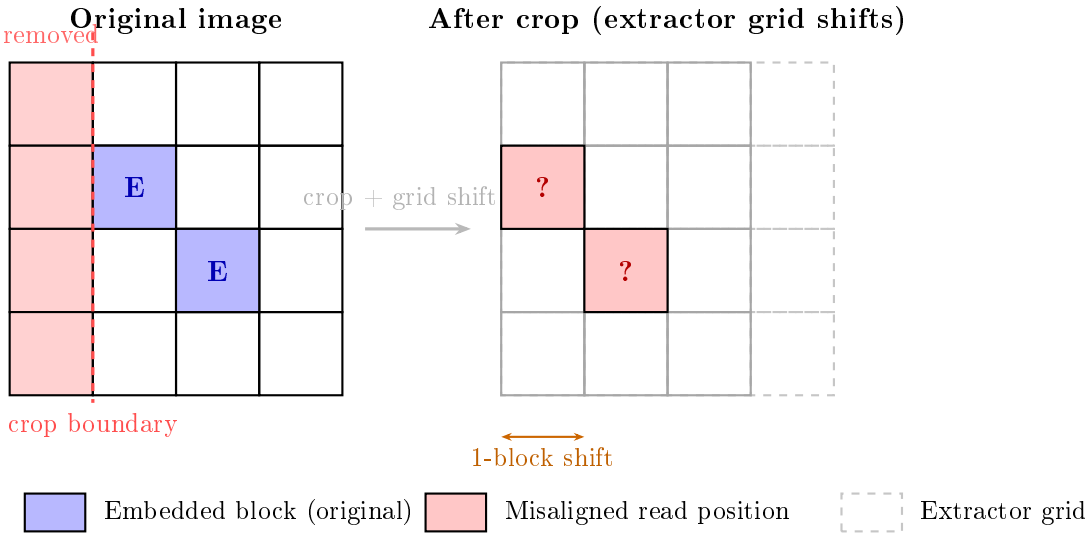


Figure 2.13: Effect of cropping on block-based DCT embedding. Column 0 (shaded red, left) is removed by the crop. Because the extractor anchors its 8×8 block grid to the top-left corner of the received image, the entire grid shifts by one block position. The extractor reads coefficients from the wrong blocks (marked ?), even though the embedded blocks E were not themselves removed. This synchronisation failure produces a near-random BER regardless of the signal-level embedding quality.

Resizing and Resampling

Resizing changes the spatial resolution of the image by either upsampling (increasing dimensions) or downsampling (reducing dimensions) through pixel interpolation.

Formal degradation model. For a scaling factor s (where $s < 1$ denotes downsampling and $s > 1$ upsampling), the resized image is computed through a resampling filter:

$$\tilde{I}(x, y) = \sum_m \sum_n I(m, n)\, k\Big(\frac{x}{s} - m,\, \frac{y}{s} - n\Big)\,, \tag{2.30}$$

where $k(\cdot, \cdot)$ is an interpolation kernel. Common kernels are:

- **Nearest-neighbour:** $k(t) = \mathbf{1}[|t| < 0.5]$ — fast but introduces blocking artefacts.
- **Bilinear:** $k(t) = \max(0, 1 - |t|)$ — linear interpolation between adjacent pixels; attenuates high-frequency content.
- **Bicubic:** a piecewise cubic kernel that provides better frequency preservation than bilinear but introduces slight ringing at edges [6].

Impact on steganographic embedding. Resampling has two effects. First, the interpolation kernel acts as a low-pass filter, attenuating high-frequency components similarly to Gaussian blur. Second, and more critically, resampling at a non-integer scale factor changes the pixel count, destroying the one-to-one correspondence between embedding positions and extraction positions. For DCT-based embedding, downsampling by a factor $s \neq 1$ shifts the 8×8 block boundaries to non-integer positions, misaligning the extractor's grid. For spread-spectrum embedding, resampling alters the spatial positions at which the pseudo-random spreading sequence is sampled, causing the correlation in Equation (2.16) to collapse [19].

Rotation

Rotation transforms the image by an angle θ about a centre point, resampling pixels from their new positions.

Formal degradation model. The rotation transformation maps pixel coordinates (x, y) to new coordinates (x', y') according to:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - x_c \\ y - y_c \end{bmatrix} + \begin{bmatrix} x_c \\ y_c \end{bmatrix}, \tag{2.31}$$

where (x_c, y_c) is the centre of rotation. Pixel values at non-integer coordinates (x', y') are obtained by interpolation. Rotation also typically introduces a band of undefined pixels at the image boundary, which are either cropped or filled with a constant value, further reducing the usable image area.

Impact on steganographic embedding. Even a small rotation of $\theta = 1$ displaces pixels near the image boundary by several pixels, making it impossible for a block-based extractor to locate its 8×8 grids. For DFT-based steganography, the Fourier magnitude spectrum rotates by the same angle θ in the frequency domain, which preserves energy at the same radial frequency but shifts angular position [16]. This provides partial robustness: embeddings in rotationally symmetric regions of the DFT spectrum (concentric rings) survive rotation, while embeddings at specific angular positions do not. Log-polar DFT embedding achieves full rotation-scale invariance but at significant capacity cost [17].

Affine and Projective Transformations

More general spatial transformations include affine mappings (combinations of rotation, scaling, shearing, and translation) and projective (homographic) transformations, which arise in print-scan scenarios where the camera is not perfectly parallel to the printed surface [29]. An affine transformation applies:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{A} \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{t}, \tag{2.32}$$

where $\mathbf{A}$ is a 2×2 matrix encoding rotation, scaling, and shear, and $\mathbf{t}$ is a translation vector. Projective transformations add perspective distortion, mapping straight lines to straight lines but not preserving parallel lines or angles. Both classes of transformation are highly destructive to all location-dependent steganographic methods and require either invariant-domain embedding or template-based re-synchronisation for recovery [1].

2.4.3 Compound and Pipeline Attacks

In practice, stego-images are rarely subjected to a single isolated attack. Real-world distribution channels apply multiple operations sequentially, each building on the damage introduced by the previous stage. This is the *compound attack* or *pipeline attack* model, and it represents the most realistic and challenging threat scenario for any robust steganographic system [2].

Social media processing pipelines. Table 2.2 summarises the documented processing pipelines applied to uploaded images by major social media and messaging platforms. Each pipeline represents a compound attack that a stego-image must survive if the steganographic channel is to function through that platform.

Table 2.2: Documented image processing operations applied by major social media and messaging platforms to uploaded images. The combination of these operations constitutes a compound attack on embedded steganographic payloads. QF: JPEG quality factor. Information compiled from [3, 4, 2].

Platform	JPEG recompr.	Resize	Crop	Format conv.	Approx. QF
WhatsApp	Yes	Yes (max 1600px)	Sometimes	JPEG	75–85
Instagram	Yes	Yes (max 1080px)	Yes (square)	JPEG	70–80
Twitter/X	Yes	Yes (max 4096px)	No	JPEG/WebP	85
Facebook	Yes	Yes	No	JPEG	70–85
Telegram	Yes	Sometimes	No	JPEG	90–95

Formal compound degradation model. A compound attack can be modelled as the sequential application of K individual degradation operators $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K$:

$$\tilde{I} = \mathcal{D}_K \circ \mathcal{D}_{K-1} \circ \cdots \circ \mathcal{D}_1(I_s), \tag{2.33}$$

where I_s is the stego-image and $\circ$ denotes function composition. The compound BER is not simply the sum of the individual BERs; operators interact non-linearly. In particular, geometric distortion followed by JPEG compression is far more destructive than either alone, because the geometric operation shifts block boundaries and the subsequent JPEG re-compression then operates on misaligned blocks, amplifying both synchronisation loss and quantisation error.

Empirical impact. Studies have shown that a WhatsApp-equivalent pipeline (resize to 1600px + JPEG QF = 80) reduces the BER recovery of a standard DCT mid-frequency embedding from approximately 8% (JPEG alone) to over 35%, constituting near-total communication failure [3, 1]. This motivates the need for a framework that addresses compound attacks as a first-class design objective rather than evaluating individual attacks in isolation.

Figure 2.14: A typical social media compound attack pipeline applied to a stego-image. Each stage introduces a distinct type of degradation: resizing causes block misalignment and acts as a low-pass filter; JPEG recompression introduces quantisation error in DCT coefficients; format conversion may introduce additional colour-space transformations. The cumulative effect is far more destructive than any single attack alone.

2.4.4 Steganalytic Attacks

In addition to signal degradation and geometric distortion, steganographic systems face a fundamentally different class of attack: *steganalysis*, the statistical detection of hidden communication. A steganalytic attack does not aim to destroy the hidden payload but to determine whether one exists, thereby exposing the covert channel and potentially leading to further scrutiny or interception [9].

Statistical Detection Principles

Cover images and stego-images differ in their statistical properties because the embedding process necessarily alters the image signal. Statistical steganalysis exploits these differences by computing feature vectors that capture deviations from the statistics of natural, unmodified images [9]. The Cartesian Calibration framework introduced by Fridrich and Goljan [?] models the statistical footprint of embedding as a deviation from the "natural image manifold" and uses decompressed JPEG images as a calibration reference.

Classical steganalysis: targeted detectors. Early steganalytic attacks were algorithm-specific. The χ^2 (chi-squared) attack [34] exploits the symmetry in histogram pairs produced by LSB substitution: embedding causes adjacent pixel value pairs $(2k, 2k+1)$ to equalise in frequency, a pattern that is statistically impossible in natural images. The RS (Regular-Singular) analysis [10] extends this to spatial neighbourhoods, achieving reliable detection of LSB embedding at rates as low as 10% bpp. Both attacks are algorithm-specific and fail against other embedding methods.

Universal steganalysis. Modern steganalysis is algorithm-agnostic. The Rich Model (RM) framework of Fridrich and Kodovský [9] computes co-occurrence matrices of pixel residuals in multiple directions and at multiple scales, constructing a feature vector of dimension 34 671. Paired with an ensemble classifier, the Rich Model achieves near-perfect detection of modern spatial-domain steganographic algorithms (WOW, HILL, S-UNIWARD) at embedding rates of 0.4 bpp, and detection above chance at rates as low as 0.05 bpp. This effectively closes the spatial domain for steganographic use at any non-trivial payload rate.

Deep learning steganalysis. As discussed in Section 2.6.9, deep neural network steganalysers such as SRNet [35], XuNet [36], and Ye-Net [37] extend universal steganalysis to a data-driven framework that generalises across embedding algorithms and image sources, achieving detection performance that exceeds the Rich Model on most benchmarks.

Implications for Robustness-Oriented Design

Steganalytic attacks impose a fundamental constraint on how robustness can be achieved. As established in Section 2.3.7, increasing embedding strength improves signal-level robustness but increases statistical detectability. This detectability–robustness trade-off can be expressed formally: for a steganographic embedding that adds a signal of energy E_s to the cover image, both the PSNR (imperceptibility proxy) and the detectability (as measured by the Area Under the ROC Curve, AUC, of a steganalytic classifier) are monotonically related to E_s:

$$\text{PSNR} \propto -\log E_s, \qquad \text{AUC} \propto f(E_s), \tag{2.34}$$

where $f(\cdot)$ is a monotonically increasing function of E_s [9]. Any design that increases E_s to improve robustness therefore simultaneously increases both detectability and distortion. This triple tension — robustness, imperceptibility, undetectability — is the fundamental constraint that distinguishes robust steganographic design from digital watermarking design.

This thesis treats steganalytic resistance as a secondary concern, explicitly noting that the primary objective is payload survivability under degradation rather than undetectability. However, the structural approach to robustness adopted in the proposed framework (fragmentation, erasure coding, low-per-coefficient distortion) is inherently more steganalysis-resistant than high-strength single-domain embedding, because the embedding energy is distributed across a larger number of coefficients at lower individual distortion per coefficient.

2.4.5 Unified Attack Summary and Implications for Framework Design

Table 2.3 provides a comprehensive unified summary of all attack types reviewed in this section, their formal degradation mechanisms, the steganographic failure modes they introduce, and their relevance to the proposed self-recovering framework.

Table 2.3: Unified summary of attack types, degradation mechanisms, failure modes, and implications for the proposed self-recovering framework. BER: Bit Error Rate; ECC: Error Correction Code; SS: Spread Spectrum; PRR: Partial Recovery Rate.

Attack	**Degradation model**	**Failure mode**	**Survivable by**	**Framework implication**
JPEG compression	Coefficient quantisation (Eq. 2.18)	High BER in unstable coefficients	ECC, mid-freq. embedding	Embed in DCT mid-freq.; strength $> Q/2$
AWGN	Additive Gaussian (Eq. 2.21)	Random bit flips	ECC, SS spreading	ECC at fragment level corrects noise errors
Low-pass filtering	Convolution attenuation (Eq. 2.26)	Erasure of high-freq. embedding	Mid-freq. embedding	Avoid high-frequency subbands
Cropping	Spatial erasure (Eq. 2.27) + grid shift	Fragment loss + sync loss	Redundancy, fragmentation	Distributed fragments maximise PRR under crop
Resizing	Interpolation (Eq. 2.30) + scale shift	Block misalignment + signal attenuation	Invariant domains, multi-scale	Multi-scale redundant embedding reduces sensitivity
Rotation	Coordinate transform (Eq. 2.31) + interpolation	Global synchronisation loss	Log-polar DFT, templates	DFT magnitude used for geometric-invariant copies
Compound pipeline	Sequential composition (Eq. 2.33)	Cascaded signal + sync degradation	Structured redundancy only	Multi-domain distribution essential; no single domain sufficient
Steganalysis	Statistical feature deviation	Channel exposure (not data loss)	Low distortion, structural embedding	Low per-coefficient distortion limits statistical footprint

The most important observation from Table 2.3 is the asymmetry between signal processing attacks and geometric attacks in terms of their countermeasures. Signal processing attacks introduce bit-level errors that can be corrected by ECC, provided that synchronisation is maintained. Geometric attacks destroy synchronisation, making ECC irrelevant because the bit order is lost. The proposed framework addresses this asymmetry by maintaining synchronisation robustness through structured spatial distribution of fragments (so that surviving fragments can be independently located) and multi-domain embedding (so that the geometric invariance of DFT-domain copies provides fallback synchronisation when block-based copies fail).

Compound attacks, which combine both classes of degradation, can only be addressed by a system-level strategy that simultaneously tolerates signal-level errors and fragment-level erasures. This is precisely the design philosophy of the self-recovering framework proposed in Chapter 3: rather than optimising resistance to any single attack, the framework is engineered to maintain a positive PRR — a non-zero fraction of recovered fragments — under any compound attack that does not exceed the system's erasure budget.

2.5 Discussion and Research Gap

The literature reviewed in this chapter demonstrates significant progress in the field of image steganography, particularly in the development of transform-domain techniques and robustness-oriented embedding strategies. Methods based on DCT, DWT, and DFT have shown improved resistance to common signal processing operations compared to spatial-domain approaches, especially under mild compression and noise conditions [1]. However, despite these advancements, existing techniques continue to exhibit fundamental limitations when deployed in realistic and uncontrolled digital environments.

A primary limitation of conventional steganographic systems lies in their vulnerability to compound attacks. While many methods are evaluated against individual distortions such as JPEG compression or additive noise, real-world scenarios frequently involve sequential or combined operations, including compression followed by resizing or cropping. Under such conditions, robustness rapidly degrades, revealing the inadequacy of approaches that focus on isolated attack resistance rather than holistic survivability.

Robustness-oriented techniques, including the use of error correction codes and controlled redundancy, have been widely adopted to mitigate bit-level corruption caused by signal processing attacks [1]. Although these methods can significantly reduce extraction errors under noise and compression, they fail to address the synchronization problem introduced by geometric transformations. Cropping, resizing, and rotation disrupt spatial alignment and block correspondence, rendering the embedded data inaccessible even when it remains partially present within the image [32]. This desynchronization represents a fundamental challenge that cannot be resolved through redundancy alone.

Several synchronization-aware solutions have been proposed, such as invariant-domain embedding, feature-based alignment, and template-based synchronization markers [17]. While these approaches improve resistance to limited geometric distortions, they often introduce substantial overhead, reduce embedding capacity, or remain vulnerable to severe or compound transformations. Consequently, existing systems remain heavily dependent on accurate localization of embedding positions, making them fragile in the presence of uncontrolled spatial modifications.

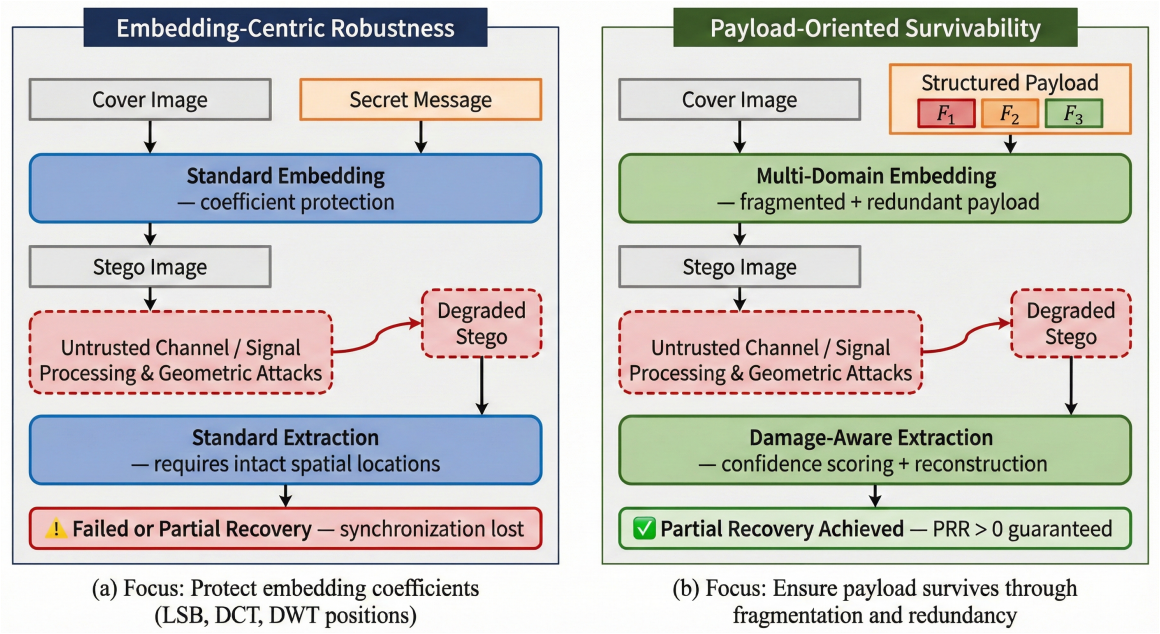
Another critical observation from the literature is the conceptual overlap between robust steganography and digital watermarking. Although both domains employ similar robustness-enhancing techniques, their objectives differ fundamentally. Watermarking prioritizes persistent detectability for ownership verification, whereas steganography requires covert communication with minimal statistical detectability [1, 10]. Many robustness-driven methods implicitly shift toward watermarking paradigms, weakening the steganographic requirement of secrecy while still failing to guarantee data recovery under severe distortions.

Furthermore, intentional detection attacks in the form of steganalysis present an additional challenge. Modern steganalysis techniques exploit statistical anomalies and learned feature representations to identify hidden communication [9]. The literature indicates a clear trade-off between robustness and undetectability, as strategies that enhance survivability—such as stronger embedding or redundancy—may increase statistical detectability. Although steganalysis resistance is not the primary focus of this thesis, its existence highlights the need for carefully balanced design choices in robust steganographic systems.

Recent deep learning-based steganographic approaches attempt to address some of these challenges through data-driven optimization [38, 39]. While these methods demonstrate promising per-

formance under trained conditions, their reliance on large datasets, lack of interpretability, and sensitivity to unseen or compound attacks limit their applicability for controlled robustness analysis. Moreover, learned models provide limited insight into synchronization behavior and payload survivability under geometric distortions, which remain central challenges in robust steganography.

Collectively, these observations reveal a critical research gap in existing work. Current steganographic systems predominantly focus on protecting embedding locations and coefficients rather than ensuring the survivability of the hidden payload itself. Robustness is often evaluated in terms of local signal preservation, while the ability to recover meaningful information under partial data loss and desynchronization remains largely unexplored.



(a) Focus: Protect embedding coefficients (LSB, DCT, DWT positions)

(b) Focus: Ensure payload survives through fragmentation and redundancy

Figure 2.15: Comparison between embedding-centric robustness and payload-oriented survivability.

This thesis addresses this gap by shifting the design focus from embedding-centric robustness to payload-oriented survivability. Instead of assuming perfect synchronization or complete data preservation, the proposed framework emphasizes structured message fragmentation, controlled redundancy, and multi-domain distribution to enable partial recovery under severe and compound image degradation. By tolerating data loss and synchronization imperfections, the framework aims to achieve reliable covert communication in hostile and uncontrolled digital environments.

2.6 Recent Advances in Deep Learning-Based Steganography

The preceding sections have reviewed the classical foundations of robust image steganography, covering spatial-domain techniques, frequency-domain transforms (DCT, DWT, DFT), and robustness-oriented design strategies developed primarily between the late 1990s and the mid-2010s. Since 2018, however, the field has undergone a dramatic transformation driven by the application of deep learning. Convolutional neural networks (CNNs), generative adversarial networks (GANs), and attention-based architectures have been adapted to the steganographic task, promising simultaneous optimisation of imperceptibility, payload capacity, and robustness in ways that classical hand-crafted methods cannot easily achieve.

This section surveys the most influential deep learning-based steganographic systems published between 2018 and 2024, with particular attention to their robustness properties and the extent to which they address the payload-survivability problem that motivates the present work. The section concludes with a critical assessment of the limitations of data-driven approaches and an explicit justification for the classical model-driven framework adopted in this thesis.

2.6.1 The Shift Toward End-to-End Learned Steganography

Classical steganographic systems are hand-engineered: the designer chooses a transform domain, selects embedding positions, fixes an embedding rule, and derives extraction analytically from the same rule. Every component is transparent and interpretable. The limitation is that design decisions are made independently and are therefore individually sub-optimal; in particular, embedding positions and strengths are chosen without direct knowledge of how a specific attack will affect the received signal.

Deep learning approaches to steganography replace this modular pipeline with a single differentiable system trained end-to-end on a large corpus of natural images [40, 41]. In the typical architecture, an *encoder* network E_θ maps a cover image C and a secret message M to a stego-image S:

$$S = E_\theta(C,\, M), \tag{2.35}$$

while a *decoder* network D_ϕ recovers the message from a (possibly degraded) received signal $\tilde{S}$:

$$\hat{M} = D_\phi(\tilde{S}). \tag{2.36}$$

Training minimises a composite loss:

$$\mathcal{L} = \lambda_1\, \mathcal{L}_{\text{img}}(C, S) + \lambda_2\, \mathcal{L}_{\text{msg}}(M, \hat{M}) + \lambda_3\, \mathcal{L}_{\text{adv}}, \tag{2.37}$$

where $\mathcal{L}_{\text{img}}$ penalises perceptible distortion between the cover and stego-images, $\mathcal{L}_{\text{msg}}$ penalises errors in the recovered message, and $\mathcal{L}_{\text{adv}}$ is an optional adversarial term from a discriminator trained to detect the stego-image [38]. The weights $\lambda_1, \lambda_2, \lambda_3$ control the balance between the three steganographic objectives. The key advantage over classical methods is that gradient back-propagation allows the encoder and decoder to co-adapt: the encoder learns to embed information in regions that the decoder can reliably read, even after passing through a differentiable noise model that simulates real-world attacks during training.

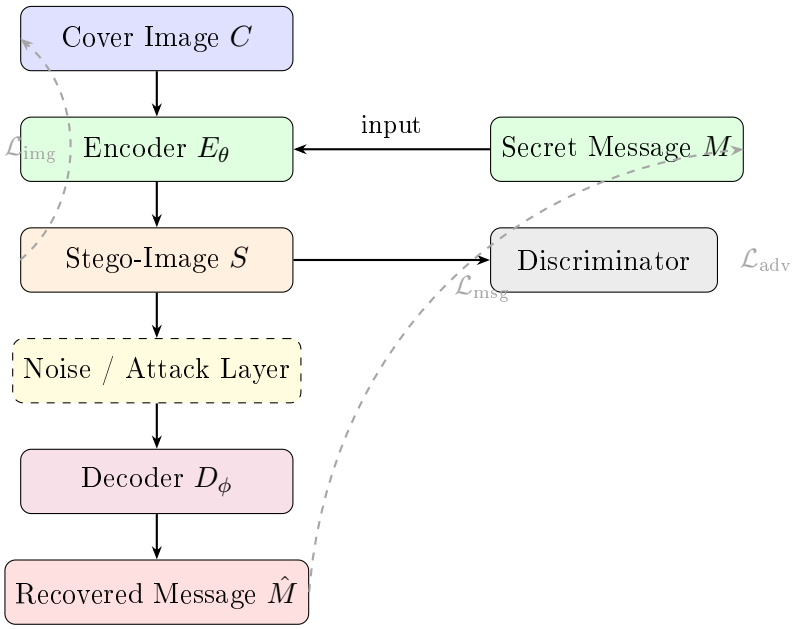


Figure 2.16: Generic encoder–decoder architecture for deep steganography. The main pipeline (left column) flows top-to-bottom: the encoder E_θ embeds the secret message into a cover image, the noise layer simulates attacks during training, and the decoder D_ϕ recovers the message. Training-side components (right column) provide the message loss $\mathcal{L}_{\text{msg}}$ and an adversarial discriminator loss $\mathcal{L}_{\text{adv}}$; the image loss $\mathcal{L}_{\text{img}}$ (dashed, left) penalises visible distortion between cover and stego-image.

2.6.2 HiDDeN: The Foundational Framework (2018)

The most influential early work in the deep steganography paradigm is HiDDeN (*Hiding Data with Deep Networks*) by Zhu et al. [40], presented at ECCV 2018. HiDDeN introduced three key ideas that have since become standard in the field.

Architecture. HiDDeN employs fully convolutional encoder and decoder networks. The encoder is a 7-layer CNN that takes the cover image and a binary secret message (replicated to match the spatial dimensions of the image) and outputs a stego-image of the same resolution. The decoder is an independent CNN that maps a received stego-image directly back to the binary message without any positional information. Neither network uses explicit transform-domain computation; the optimal embedding strategy is learned implicitly from data.

Differentiable noise layer. The central contribution of HiDDeN is the insertion of a differentiable noise layer between the encoder and decoder during training. The noise layer is a parametric module that randomly applies one or more of the following operations to the stego-image before it reaches the decoder: JPEG compression (approximated by a differentiable relaxation), Gaussian noise addition, dropout (random pixel zeroing, simulating cropping of small regions), and Gaussian blurring. Because all operations are differentiable with respect to their inputs, gradients flow back through the noise layer to the encoder during training, causing the encoder to learn embeddings that are specifically resistant to the attacks it will encounter at inference time. This is a fundamental architectural insight that distinguishes HiDDeN from classical ECC-based approaches: rather than correcting errors after they occur, the system learns to avoid them at the point of embedding.

Adversarial imperceptibility. HiDDeN adds a discriminator network trained to classify images as cover or stego. The discriminator loss is added to the training objective of the encoder, creating a min-max game that encourages the encoder to produce stego-images that are statistically indistinguishable from natural photographs. This is conceptually similar to the GAN training paradigm introduced by Goodfellow et al. [42], applied to the steganographic context.

Reported performance. On the COCO dataset, HiDDeN achieves a PSNR above 33 dB for 30-bit payloads under JPEG compression and Gaussian noise, with bit error rates below 1%, significantly outperforming the classical F5 algorithm [34] under equivalent attack conditions.

The limitations of HiDDeN, however, are equally significant. The noise layer must be fixed before training; if the deployed system encounters attacks not represented in the noise layer (e.g. spatial warping, format conversion chains, or aggressive cropping of large image fractions), the encoder has learned no strategy to resist them. This is the distribution-shift problem that recurs throughout deep steganography.

2.6.3 SteganoGAN: High-Capacity GAN-Based Steganography (2019)

Zhang et al. [43] introduced SteganoGAN in 2019, extending the HiDDeN framework with a focus on maximising embedding capacity while preserving perceptual quality. SteganoGAN proposes three encoder architectures of increasing depth — *basic*, *residual*, and *dense* — corresponding to different trade-offs between capacity and visual quality. The dense encoder, inspired by DenseNet [44], concatenates feature maps from all preceding layers, allowing the network to leverage both low-level texture features and high-level semantic features when deciding where and how strongly to embed each payload bit.

Key contributions. The primary innovation in SteganoGAN is a rigorous treatment of the capacity–quality trade-off. The authors report that their dense encoder achieves capacities of up to 4.4 bpp (bits per pixel) at a PSNR of 30.5 dB, compared to approximately 1.5 bpp for HiDDeN at equivalent

quality. The critical observation for the present work, however, is that SteganoGAN *does not include a noise layer*: training is performed under ideal (no-attack) conditions. The resulting stego-images are therefore visually excellent but highly fragile. Under JPEG compression at quality factor 70, the bit error rate of a SteganoGAN-trained decoder rises above 40%, which constitutes complete communication failure. This confirms that capacity-oriented deep models without explicit robustness training are unsuitable for uncontrolled transmission channels.

Relevance to this thesis. SteganoGAN illustrates a general principle that pervades both classical and deep steganography: in the absence of specific robustness mechanisms, higher capacity invariably leads to lower survivability. The dense architectures of SteganoGAN produce visually imperceptible embeddings but exploit fragile, high-frequency image regions for extra capacity, exactly the regions that are discarded by lossy compression.

2.6.4 ReDMark: Robustness-First Deep Steganography (2021)

Recognising the limitations of capacity-first approaches, Das et al. [45] introduced ReDMark (*Redistributing the Weights for Robustness Deep Steganographic Marking*) in 2021, the first deep steganographic system explicitly designed around the robustness objective as the primary design constraint.

Architecture and training. ReDMark employs a U-Net-style encoder [46] with skip connections that propagate spatial information from the cover image directly to the deeper layers of the embedding network. This architectural choice is motivated by the observation that skip connections encourage the encoder to modulate existing image structures rather than superimpose an independent embedding signal, resulting in stego-images that survive compression better because the embedding is "anchored" to stable image features that compression itself preserves.

The training pipeline of ReDMark incorporates a differentiable JPEG approximation module [47] that accurately models the quantisation artifacts produced by the standard JPEG codec. Additionally, a spatial transformer network (STN) [48] is inserted in the noise layer to simulate mild geometric transformations (rotation by up to 5 and scaling by up to 20%), allowing the decoder to learn a degree of geometric invariance.

Loss function and weight redistribution. The distinctive contribution of ReDMark is a dynamic loss-weight redistribution scheme. During training, the weights λ_1 and λ_2 in Equation (2.37) are adjusted adaptively based on the current attack intensity estimated from a running average of recent BER values. When the bit error rate under the noise layer is high, the training procedure temporarily increases λ_2 (message fidelity) relative to λ_1 (image quality), forcing the encoder to prioritise robustness over imperceptibility until performance stabilises. This curriculum-style approach avoids the common failure mode in which the encoder converges to an imperceptible but fragile embedding from which it cannot escape by small gradient steps.

Reported performance. ReDMark achieves a PSNR of 36.2 dB at 30-bit payload with a BER of 1.4% after JPEG compression at quality factor 50, and remains below 5% BER under moderate rotation (≤ 3) and scaling ($\leq 10\%$). These results represent a meaningful advance over HiDDeN and SteganoGAN under the same attack conditions.

Limitations. Despite these improvements, ReDMark retains the fundamental distribution-shift vulnerability. Its resistance to geometric attacks extends only to the mild transformations simulated during training. Under large-scale cropping (removal of $> 30\%$ of image content) or compound attacks combining JPEG compression with significant resizing (e.g. the pipeline applied by Twitter and WhatsApp), the BER rises sharply. Furthermore, ReDMark, like all decoder-only deep methods, provides no mechanism for partial recovery: if the bit error rate exceeds the decoder's correction

capability, the output is unintelligible, and no information about which fragments of the payload survived is available.

2.6.5 Hiding Images in Images: High-Capacity Deep Concealment

A parallel line of deep steganographic research is concerned not with hiding binary messages but with concealing entire images within other images. Baluja [41] introduced a CNN-based framework for hiding a secret image within a cover image of the same dimensions, using a "preparation" network to transform the secret image into a residual-ready representation before the hiding network embeds it into the cover. The reveal network then reconstructs the secret image from the stego-image with high fidelity (PSNR above 30 dB for both the stego-image and the revealed secret).
Lu et al. [49] extended this paradigm to *large-capacity* image hiding, proposing an invertible neural network (INN) architecture [50] that is mathematically reversible: the same network parameters are used for both embedding and extraction by exploiting the invertibility of normalising flows. This guarantees that in the absence of any attack, perfect recovery is possible. Under compression attacks, however, the invertibility is broken, and recovery degrades.

While the image-in-image paradigm is not directly applicable to the binary message concealment task of this thesis, it illustrates the general capacity of deep architectures to learn complex, content-adaptive embeddings that far exceed the capacity of classical transform-domain methods.

2.6.6 Robust Deep Steganography Against Social Network Processing

A practically important research direction targets the specific degradation pipelines applied by social media platforms (Instagram, Twitter, WhatsApp, Facebook), which combine JPEG recompression, resolution rescaling, colour-space conversion, and sometimes format conversion (e.g. JPEG to WebP). These operations collectively remove a large fraction of the information embedded by conventional and naive deep steganographic methods.

Wengrowski and Dana [4] proposed a differentiable model of the photographic process (capture, print, display, and re-capture), enabling a neural steganographic system to survive the full print-scan loop. Their system produces cover-looking images that can be photographed with a mobile camera and decoded with the neural decoder, surviving both compression and optical blur.

Jia et al. [3] introduced MBRS (*Mini-Batch of Real and Simulated JPEG Compression*), in which each training mini-batch contains stego-images processed by both a differentiable JPEG simulation and a *real* JPEG codec. Including real JPEG artifacts in the training data closes the gap between the differentiable approximation and the actual codec behaviour, yielding substantially lower BER under real-world JPEG compression compared to methods trained on differentiable approximations alone. MBRS achieves 0.9% BER at 30-bit payload under JPEG quality factor 50 and 1.3% under quality factor 30, with a stego-image PSNR of 38.1 dB.

2.6.7 Attention Mechanisms and Semantic-Aware Embedding

Transformer-based architectures and spatial attention mechanisms have been applied to steganography to improve both imperceptibility and robustness by enabling the network to focus embedding on semantically stable image regions.

Luo et al. [51] proposed a *distortion-agnostic* deep hiding network that uses a channel-spatial attention module to estimate the local imperceptibility budget at each pixel, concentrating embedding in texture-rich regions (similar to the JND-based adaptive embedding discussed in Section 2.3.4) while learning the content of those regions from data rather than from hand-crafted JND models. The attention weights are updated jointly with the encoder and decoder weights during training, allowing the system to adapt its embedding map to the statistics of natural images.

Zhang et al. [52] introduced UDH (*Universal Deep Hiding*), a framework in which the encoder learns a single, universal embedding pattern that is content-independent, i.e. the same residual

signal is added to any cover image regardless of its content. While this reduces capacity compared to content-adaptive methods, it makes the embedding pattern universal and more predictable under attacks, which simplifies decoder design.

2.6.8 Hybrid Transform-Network Approaches

A natural development in the field is the combination of classical frequency-domain transforms with learned components, seeking to obtain the theoretical guarantees and interpretability of transform-domain methods alongside the adaptivity of data-driven approaches.

Ahmadi et al. [53] proposed a framework in which the encoder operates in the DCT domain: DCT coefficients are first extracted from the cover image, and the neural network learns to modify a selected subset of mid-frequency coefficients as a function of both the cover content and the secret message. The extraction network inverts this process in the DCT domain. This hybrid approach inherits the JPEG-robustness advantage of DCT embedding and the adaptivity advantage of learned coefficient selection, achieving PSNR of 40.1 dB and BER of 2.1% under JPEG quality factor 70.

Fernandez et al. [54] explored the integration of steganography with latent diffusion models, embedding messages in the latent space of a pre-trained image generator. This approach exploits the perceptual quality of modern generative models to produce stego-images that are virtually indistinguishable from natural photographs, at the cost of requiring a generative model at inference time and being restricted to the image manifold of the training data.

2.6.9 Deep Learning Steganalysis and Its Impact on Robust Design

The same deep learning revolution that has advanced steganographic embedding has also produced substantially more powerful steganalytic detectors. The SRNet architecture of Boroumand et al. [35] achieves near-perfect detection of several state-of-the-art spatial-domain and DCT-domain steganographic algorithms using a single, unified deep network, without any algorithm-specific feature engineering. More recently, the XuNet [36] and Ye-Net [37] architectures have demonstrated that deep classifiers trained on the StegoAppDB dataset generalise across embedding algorithms, significantly raising the bar for undetectability.

These developments have important implications for the design of robust steganographic systems. Because deep steganalysis is both universal (algorithm-agnostic) and sensitive (detecting payloads at embedding rates well below 0.1 bpp in spatial-domain methods), any strategy that increases embedding strength for the sake of robustness automatically increases the statistical footprint detectable by a deep steganalyser. This confirms and deepens the detectability–robustness trade-off discussed in Section 2.3.7, and reinforces the motivation for structural robustness approaches (fragmentation, erasure coding, domain diversity) that can maintain a low per-pixel distortion while still enabling payload survivability.

2.6.10 Summary Table and Comparative Analysis

Table 2.4 summarises the deep learning-based methods reviewed in this section alongside the classical methods from Section 2.3, providing a unified comparative view.

The critical observation from Table 2.4 is consistent with the finding of Table 2.1 in Section 2.3.9: no existing deep learning system, including the most recent state-of-the-art methods, provides mechanisms for *partial recovery* of the hidden payload when large portions of the stego-image are destroyed. All deep decoders operate in a binary succeed-or-fail mode: they either reconstruct the entire message or produce unintelligible output. This is a fundamental architectural limitation of the encoder-decoder paradigm, not a parameter that can be tuned.

Table 2.4: Comparative summary of deep learning-based steganographic methods (2018–2024) alongside selected classical baselines. Performance figures are indicative only and refer to 30-bit payloads on natural image datasets under JPEG QF = 70 unless otherwise noted. BER: Bit Error Rate; PSNR: Peak Signal-to-Noise Ratio.

Method	**Year**	**PSNR (dB)**	**BER (%)**	**Geo. Rob.**	**Partial Rec.**
DCT mid-freq. (classical)	—	≈38	≈15	Poor	No
LSB spatial (classical)	—	≈51	≈50	Poor	No
HiDDeN [40]	2018	33.1	0.5	Limited	No
SteganoGAN [43]	2019	30.5	40+ (JPEG)	None	No
ReDMark [45]	2021	36.2	1.4	Limited	No
MBRS [3]	2021	38.1	0.9	Limited	No
UDH [52]	2020	34.8	2.2	Limited	No
DCT-hybrid [53]	2020	40.1	2.1	Limited	No
Proposed framework	2025	—	—	Designed	**Yes**

2.6.11 Why This Thesis Adopts a Classical Framework

The review in this section makes clear that deep learning-based steganography represents the current state of the art in terms of raw imperceptibility and robustness to trained attacks. Nevertheless, the present thesis adopts a classical, model-driven framework for several reasons that are directly related to the specific research objectives defined in Chapter 1.

Partial recovery as a first-class design objective. The primary objective of this thesis is to design a system that can recover *usable information* from a partially destroyed stego-image, rather than a system that achieves the lowest BER under a fixed set of trained attacks. As established above, no existing deep learning framework provides partial recovery capabilities; the encoder-decoder paradigm produces a holistic encoding of the entire payload, and the decoder requires the full encoded signal to function. Classical fragmentation and erasure-coding strategies (Section 2.3.5) are directly applicable to the partial-recovery objective in a way that is not yet achievable with deep architectures.

Robustness to *unseen* compound attacks. Deep steganographic systems are robust only to the attacks represented in the training noise layer. Social media platforms change their processing pipelines frequently, and different platforms apply different combinations of compression quality, resize ratios, and format conversions. A classical system whose robustness derives from structural properties of the payload (redundancy, distributed embedding, erasure coding) is robust by construction to any attack that does not exceed its erasure budget, regardless of whether that specific attack was anticipated during design. This generalisation property is especially important for deployment in uncontrolled environments.

Interpretability and analytical robustness guarantees. A classical steganographic framework admits formal analysis: given a specified attack model, one can compute (or bound) the probability that a given fragment survives, and hence the probability that the payload is recovered. Deep learning systems provide no such guarantees; their robustness is empirical and may degrade in ways that are difficult to predict. For applications in which reliability is a hard requirement, interpretability and analytical tractability are significant advantages.

Computational accessibility. Training a competitive deep steganographic system requires large-scale image datasets (typically tens of thousands of images) and GPU-accelerated computing infrastructure for extended training runs (often 24–100 hours). The classical framework proposed in this thesis is fully implementable on standard hardware without any training phase, making it accessible in resource-constrained environments.

Modularity and domain expertise. The classical framework allows individual components (the ECC scheme, the fragmentation strategy, the embedding domain, the embedding strength) to be adjusted independently based on the properties of the specific deployment environment. In a deep system, these design choices are entangled within the network weights and cannot be modified without retraining. This modularity is particularly valuable when the attack model is only partially known in advance.

In summary, this thesis positions itself within the classical stream of robust steganography research, but draws inspiration from the structural insights of deep learning approaches — in particular, the idea that robustness is best achieved through a system-level design philosophy rather than through individually robust components. The proposed framework combines structured payload fragmentation, multi-domain embedding, and erasure-code redundancy to achieve the partial-recovery objective that neither classical single-domain methods nor current deep learning systems have addressed.

2.7 Evaluation Criteria in Steganographic Systems

The design and experimental evaluation of any steganographic system requires a precise set of quantitative metrics that measure performance along the three fundamental axes introduced in Section 2.1.1: imperceptibility, robustness, and (where applicable) steganalysis resistance. A clear understanding of these metrics — their mathematical definitions, their practical interpretation, and their limitations — is essential both for comparing different steganographic methods in the literature and for assessing the proposed framework in Chapter 5.

This section provides a comprehensive treatment of the evaluation criteria used throughout this thesis, divided into two groups: imperceptibility metrics (Section 2.7.1) and robustness metrics (Section 2.7.2).

2.7.1 Imperceptibility Metrics

Imperceptibility metrics quantify the visual and statistical difference between a cover image and the corresponding stego-image. A high-quality embedding is one in which these differences are small enough to be undetectable by human observers and difficult to distinguish statistically from natural image noise. The three metrics used in this work are the Mean Squared Error (MSE), the Peak Signal-to-Noise Ratio (PSNR), and the Structural Similarity Index Measure (SSIM).

Mean Squared Error (MSE)

The Mean Squared Error is the most fundamental measure of pixel-level distortion between two images. Given a cover image C and a stego-image S, each of dimensions $M \times N$ pixels, the MSE is defined as:

$$\mathrm{MSE}(C,\, S) = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \big[C(i,j) - S(i,j)\big]^2, \tag{2.38}$$

where $C(i,j)$ and $S(i,j)$ denote the intensity values of the cover and stego-images at pixel location (i,j), respectively. For colour images, the MSE is commonly computed channel-wise and averaged:

$$\mathrm{MSE}_{\text{colour}} = \frac{1}{3}\left(\mathrm{MSE}_R + \mathrm{MSE}_G + \mathrm{MSE}_B\right). \tag{2.39}$$

The MSE is measured in squared intensity units (e.g. squared grey levels for 8-bit images, 0–255^2). A lower MSE value indicates less pixel-level distortion. However, MSE has a well-known limitation: equal MSE values can correspond to perceptually very different types of distortion. For example, a small uniform shift in pixel values across a flat region produces the same MSE as an equal-energy pattern of localised noise, but the latter is far more visually conspicuous. For this reason, MSE is rarely used as the sole measure of imperceptibility in steganographic evaluations; it serves primarily as the mathematical foundation for PSNR.

Peak Signal-to-Noise Ratio (PSNR)

The Peak Signal-to-Noise Ratio is the most widely used imperceptibility metric in the steganography and image processing literature [6]. It expresses the ratio of the maximum possible pixel energy to the distortion energy introduced by embedding, measured on a logarithmic (decibel) scale:

$$\mathrm{PSNR}(C,\, S) = 10 \cdot \log_{10}\!\left(\frac{L^2}{\mathrm{MSE}(C,\, S)}\right) \quad [\mathrm{dB}], \tag{2.40}$$

where L is the maximum possible pixel intensity value. For 8-bit images, $L = 255$; for 16-bit images, $L = 65\,535$. The logarithmic scale is chosen because human perception of brightness differences is approximately logarithmic. PSNR is always expressed in decibels (dB).

Interpretation and thresholds. PSNR is a monotonically increasing function of perceptual quality. The following thresholds serve as practical guidelines in the steganographic literature [1]:

- PSNR $> 40\,\mathrm{dB}$: Excellent imperceptibility; the stego-image is typically indistinguishable from the cover image to human observers under normal viewing conditions.
- $35 \leq$ PSNR $\leq 40\,\mathrm{dB}$: Good imperceptibility; the embedding is acceptable for most applications where human inspection is not performed under magnification.
- $30 \leq$ PSNR $< 35\,\mathrm{dB}$: Acceptable imperceptibility; a trained observer may detect subtle artefacts, but casual inspection will not reveal the embedding.
- PSNR $< 30\,\mathrm{dB}$: Poor imperceptibility; visible artefacts are present and the embedding is unsuitable for covert communication.

Limitations. Despite its ubiquity, PSNR has well-documented limitations as a perceptual quality metric [55]. It treats all pixel positions and all spatial frequencies as equally important, whereas the human visual system (HVS) is significantly more sensitive to distortion in smooth, homogeneous image regions than in textured areas, and more sensitive to low-frequency errors than to high-frequency ones. Two stego-images with identical PSNR values may therefore be perceived very differently by a human observer. These limitations motivate the use of SSIM as a complementary metric.

Relationship between PSNR and embedding capacity. For a steganographic method that embeds data by adding an embedding signal of energy E to the cover image, Equation (2.40) shows that PSNR decreases by approximately 3 dB for every doubling of the embedding strength. More precisely, for a fixed image size MN and a fixed embedding signal, doubling the number of payload bits requires doubling the average energy per bit, reducing the PSNR by 3 dB. This energy-capacity relationship is the mathematical expression of the imperceptibility–capacity trade-off discussed in Section 2.1.1.

Structural Similarity Index Measure (SSIM)

The Structural Similarity Index Measure was introduced by Wang et al. [55] as a perceptual quality metric that models the sensitivity of the HVS to structural information. The key insight is that natural images are highly structured: neighbouring pixels are strongly correlated, and these correlations carry important perceptual information about object boundaries, textures, and surfaces. SSIM measures the degradation of this structural information rather than pixel-level intensity differences.

SSIM is computed locally over rectangular windows (typically 11×11 pixels with a Gaussian weighting function) and then averaged across all window positions to produce a single global score. For two image patches $\mathbf{x}$ and $\mathbf{y}$ drawn from the cover and stego-images respectively, the local SSIM is defined as:

$$\mathrm{SSIM}(\mathbf{x}, \mathbf{y}) = \underbrace{\frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}}_{\text{luminance}} \cdot \underbrace{\frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}}_{\text{contrast}} \cdot \underbrace{\frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}}_{\text{structure}}, \tag{2.41}$$

where μ_x, μ_y are the mean intensities of the two patches; σ_x^2, σ_y^2 are their variances; σ_{xy} is their covariance; and c_1, c_2, c_3 are small stabilisation constants that prevent division by zero (typically $c_1 = (0.01L)^2$, $c_2 = (0.03L)^2$, $c_3 = c_2/2$).

Interpretation. The SSIM index ranges from -1 to 1, where 1 indicates perfect structural similarity (identical images) and 0 indicates no structural correlation. In practice, for high-quality stego-images, SSIM values are very close to 1:

- SSIM > 0.99: Excellent; structural differences are imperceptible.
- $0.97 \leq$ SSIM ≤ 0.99: Good; minor structural changes, not visible under normal conditions.
- $0.90 \leq$ SSIM < 0.97: Acceptable; some structural artefacts may be visible in textured regions.
- SSIM < 0.90: Poor; structural degradation is perceptible.

SSIM is generally considered a more reliable predictor of perceived image quality than PSNR, particularly for compression-type distortions [55]. It is therefore reported alongside PSNR in the experimental evaluation of Chapter 5.

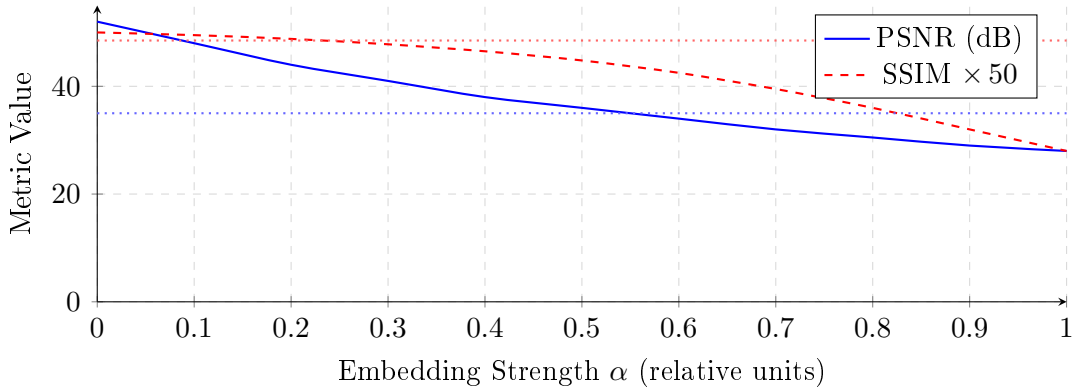


Figure 2.17: Illustrative relationship between embedding strength α and the two imperceptibility metrics PSNR and SSIM (scaled by 50). Both metrics decrease monotonically with increasing embedding strength. PSNR degrades rapidly even at moderate strengths, while SSIM is more tolerant of small-amplitude, high-frequency embedding signals that preserve structural content. This highlights the complementary nature of the two metrics.

Relationship Between Imperceptibility Metrics

Although PSNR and SSIM are both imperceptibility metrics, they capture different aspects of image quality and can sometimes disagree on the ranking of steganographic methods. Embedding in the low-frequency components of an image (e.g. the DC coefficients of DCT) produces large intensity shifts that severely degrade PSNR but may preserve structural correlations and therefore maintain a relatively high SSIM. Conversely, high-frequency noise-like embeddings may produce a high PSNR (because energy is spread thinly across many pixels) while destroying textural structures and yielding a low SSIM. This complementarity explains why both metrics are reported in Chapter 5 and why neither alone provides a complete picture of imperceptibility.

2.7.2 Robustness Metrics

Robustness metrics quantify the ability of a steganographic system to recover the hidden message after the stego-image has been subjected to signal processing or geometric attacks. The two metrics used in this work are the Bit Error Rate (BER) and the Normalised Correlation (NC).

Bit Error Rate (BER)

The Bit Error Rate is the primary robustness metric in steganography and digital communications [7]. It measures the fraction of payload bits that are incorrectly recovered after extraction from a (possibly degraded) stego-image. Formally, let $M = [m_1, m_2, \ldots, m_K]$ be the original binary message of length K bits, and let $\hat{M} = [\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_K]$ be the extracted message after attack. The BER is defined as:

$$\mathrm{BER}(M,\, \hat{M}) = \frac{1}{K} \sum_{k=1}^{K} \mathbf{1}\big[\hat{m}_k \neq m_k\big], \tag{2.42}$$

where $\mathbf{1}[\cdot]$ is the indicator function that equals 1 if its argument is true and 0 otherwise. The BER is a dimensionless quantity in the range $[0, 1]$, typically expressed as a percentage or in scientific notation.

Interpretation and thresholds.

- BER = 0.0 (or 0%): Perfect recovery; every bit of the hidden message is correctly extracted. This is the ideal outcome and is achievable only in the absence of attacks or with sufficiently powerful error correction coding.
- BER $\leq$ 0.01 ($\leq$ 1%): Excellent robustness; with appropriate error correction coding, the original message can be recovered reliably.
- BER $\leq$ 0.05 ($\leq$ 5%): Good robustness; recovery is possible with moderate ECC overhead.
- BER $\approx$ 0.5 ($\approx$ 50%): Complete failure; the extracted bits are no better than random, meaning the embedding has been entirely destroyed by the attack. This is the expected BER when attempting to extract a message from a cover image in which nothing was embedded.

Relationship to error correction coding. When the steganographic payload is protected by an error correction code with correction capacity t (i.e. the code can correct up to t bit errors per codeword), perfect message recovery is possible if and only if the BER satisfies:

$$\mathrm{BER} < \frac{t}{n}, \tag{2.43}$$

where n is the codeword length. For example, a Reed-Solomon code with $n = 255$ and $t = 31$ can correct up to $\approx 12.2\%$ BER. This relationship motivates the use of ECC in the proposed framework and defines the BER budget available to the embedding and transmission process.

BER under partial data loss. In the context of the present thesis, where the stego-image may be subjected to cropping or localised tampering, the effective BER includes contributions from two distinct sources: (i) signal-level errors caused by noise and compression in the surviving image regions, and (ii) erasure errors caused by the complete removal of image regions containing embedded fragments. These two error types require different mitigation strategies — ECC for signal-level errors and erasure codes or redundancy for erasure errors — and are therefore tracked separately in the experimental evaluation of Chapter 5.

Normalised Correlation (NC)

The Normalised Correlation provides an alternative measure of message recovery fidelity that is more robust to systematic bit inversions than the raw BER. It is commonly used in digital watermarking [1] and has been adopted in robust steganographic evaluation to provide a continuous measure of recovery quality that degrades smoothly as attack intensity increases. The NC between the original message M and the extracted message $\hat{M}$, after bipolar encoding ($0 \to -1$, $1 \to +1$), is defined as:

$$\mathrm{NC}(M, \hat{M}) = \frac{1}{K} \sum_{k=1}^{K} \bar{m}_k \cdot \hat{\bar{m}}_k, \tag{2.44}$$

where $\bar{m}_k \in \{-1, +1\}$ and $\hat{\bar{m}}_k \in \{-1, +1\}$ are the bipolarly encoded versions of the original and extracted bits, respectively.

Interpretation.

- NC = 1.0: Perfect recovery; all bits agree.
- NC = 0.0: Complete failure; extracted bits are uncorrelated with the original, as would occur if a random sequence were substituted.
- NC = −1.0: All bits are inverted; this can occur under systematic polarity reversal attacks and would be incorrectly reported as BER = 1.0 (perfect failure) even though the message is recoverable by simple inversion. The NC captures this case.

The relationship between NC and BER for independently and identically distributed bit errors is:

$$\mathrm{NC} = 1 - 2 \cdot \mathrm{BER}, \tag{2.45}$$

from which it follows that BER = 0 corresponds to NC = 1 and BER = 0.5 corresponds to NC = 0. Both metrics therefore convey the same information for independent errors, but NC is preferable when systematic errors or polarity reversals are possible. In this work, both metrics are reported to facilitate comparison with results from the watermarking literature, which predominantly uses NC.

2.7.3 Partial Recovery Rate (PRR)

Classical steganographic evaluation is concerned with binary outcomes: either the entire message is recovered (BER ≈ 0) or it is not. In the context of the self-recovering framework proposed in this thesis, however, the system is explicitly designed to recover *partial* messages when not all embedded fragments survive. A metric that captures this partial-recovery behaviour is therefore needed.

We define the **Partial Recovery Rate (PRR)** as the fraction of payload fragments that are successfully recovered (with BER below a threshold ϵ for each fragment) from the degraded stego-image:

$$\mathrm{PRR} = \frac{|\{i : \mathrm{BER}(F_i, \hat{F}_i) \leq \epsilon\}|}{N_F}, \tag{2.46}$$

where N_F is the total number of fragments in the structured payload, F_i is the i-th original fragment, $\hat{F}_i$ is the extracted version, and ϵ is a per-fragment BER threshold (typically 0.05 when ECC is applied

at the fragment level). A PRR of 1.0 indicates complete recovery; a PRR of 0.5 indicates that half the message fragments were correctly extracted; a PRR of 0.0 indicates total failure. This metric is unique to the present work and is reported in Chapter 5 to assess the partial-recovery performance of the proposed framework under cropping and localised tampering attacks.

2.7.4 Unified View of Evaluation Criteria

Figure 2.18 provides a visual overview of the relationship between the evaluation metrics defined in this section and the three steganographic objectives.

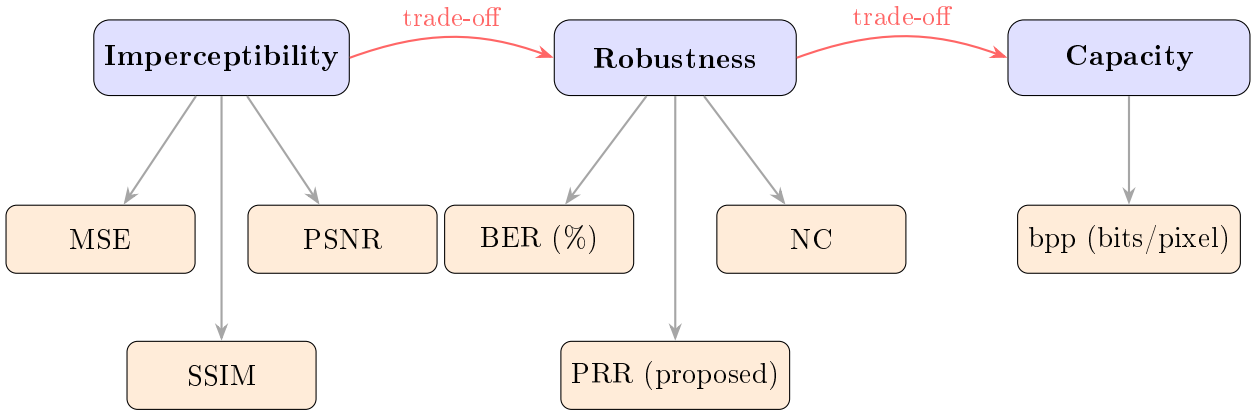


Figure 2.18: Relationship between the three steganographic objectives and the quantitative evaluation metrics used in this thesis. The trade-offs between objectives are captured by tracking all metrics simultaneously across experimental conditions. PRR (Partial Recovery Rate) is a metric introduced in this work to assess the unique partial-recovery capability of the proposed framework.

Table 2.5 summarises all evaluation metrics, their definitions, their measurement units, and the performance thresholds adopted in this thesis.

Table 2.5: Summary of evaluation metrics used in this thesis. Thresholds represent commonly accepted standards in the steganographic and image quality literature.

Metric	**Objective**	**Formula**	**Range**	**Good threshold**
MSE	Imperceptibility	$\frac{1}{MN}\sum(C-S)^2$	$[0,\infty)$	As small as possible
PSNR	Imperceptibility	$10\log_{10}(L^2/\text{MSE})$	$(0,\infty)$ dB	> 35 dB
SSIM	Imperceptibility	Eq. (2.41)	$[-1, 1]$	> 0.97
BER	Robustness	$\frac{1}{K}\sum \mathbf{1}[\hat{m} \neq m]$	[0, 1]	< 0.05
NC	Robustness	$\frac{1}{K}\sum \bar{m} \cdot \hat{\bar{m}}$	$[-1, 1]$	> 0.9
PRR	Partial recovery	Eq. (2.46)	[0, 1]	> 0.7 (proposed)
bpp	Capacity	bits embedded per pixel	(0, 8]	Application-dependent

These metrics, taken together, provide a comprehensive and multi-dimensional assessment of steganographic performance. In Chapter 5, each metric is reported for the proposed framework and for the classical baselines under each attack scenario, enabling a rigorous comparative evaluation that respects the multi-objective nature of the steganographic design problem.

Chapter 3

Proposed Self-Recovering Steganography Framework

3.1 Design Motivation and Objectives

This chapter introduces the proposed steganographic framework, which is designed to address the limitations identified in the literature review, particularly the fragility of conventional embedding methods under severe image degradation and geometric desynchronization.

The primary design goal of the proposed framework is to enhance payload survivability rather than solely reinforcing embedding locations. Unlike traditional approaches that assume intact data and perfect synchronization, the proposed system explicitly assumes partial data loss and uncontrolled image transformations. The framework is therefore designed to tolerate damage, enable partial recovery, and reconstruct missing information using structured payload relationships.

3.2 High-Level System Overview

This section presents a conceptual overview of the proposed framework, illustrating the main processing stages at both the sender and receiver sides.

At a high level, the system consists of four main components:

- Payload structuring and self-description
- Fragmentation and inter-fragment relationship encoding
- Multi-domain embedding strategy
- Damage-aware extraction and reconstruction

The interaction between these components enables robust message recovery even when a significant portion of the embedded data is degraded or removed.

3.2.1 Sender-Side Processing Pipeline

The sender-side pipeline begins with the secret message and produces a stego-image suitable for transmission through uncontrolled digital environments.

The main stages include:

- Conversion of the secret message into a structured payload
- Fragment generation and relationship encoding
- Selection of embedding domains and regions
- Embedding of fragments into the cover image

3.2.2 Receiver-Side Processing Pipeline

The receiver-side pipeline operates on a potentially degraded stego-image and attempts to recover the original message.

The extraction process includes:

- Detection and extraction of embedded fragments
- Damage estimation and fragment confidence evaluation
- Fragment validation and relationship checking
- Partial reconstruction and message reassembly

3.3 Structured Payload Representation

Instead of embedding raw binary data, the proposed framework transforms the secret message into a self-describing structured payload. This payload contains not only the original message bits but also additional information that enables validation, ordering, and reconstruction.

3.3.1 Payload Components

Each payload fragment contains the following elements:

- Fragment identifier and ordering information
- Local payload data
- Lightweight checksum for integrity verification
- Cross-fragment reference information

This structure allows individual fragments to contribute to the reconstruction of missing or corrupted fragments.

3.3.2 Inter-Fragment Relationships

Fragments are not treated as independent entities. Instead, each fragment stores partial information about neighboring and non-neighboring fragments. These relationships form a redundancy graph that supports reconstruction under partial data loss.

The use of relational information enables recovery even when contiguous regions of the image are destroyed.

3.4 Fragmentation Strategy

The structured payload is divided into multiple interconnected fragments to increase resilience against localized damage.

3.4.1 Fragment Size and Granularity

Fragment size is chosen as a compromise between robustness and capacity. Smaller fragments improve survivability under cropping, while larger fragments reduce overhead.

This work adopts a fixed fragment size to simplify reconstruction and evaluation.

3.4.2 Redundancy and Distribution Policy

Controlled redundancy is introduced at the fragment level rather than the bit level. Each fragment is embedded multiple times across different domains or image regions according to a predefined distribution policy.

The redundancy level is configurable and directly impacts robustness and payload capacity.

3.5 Multi-Domain Embedding Strategy

To reduce vulnerability to specific attack types, fragments are embedded across multiple embedding domains, each offering resistance to different forms of degradation.

3.5.1 Spatial Domain Embedding

Fragments embedded in spatial texture regions are resilient to localized cropping and partial tampering. Texture-based masking is used to minimize visual distortion.

3.5.2 Frequency Domain Embedding

Mid-frequency DCT coefficients are used to embed fragments resistant to JPEG compression and moderate noise. Embedding strength is adjusted to balance imperceptibility and robustness.

3.5.3 Multi-Scale and Redundant Embedding

Selected fragments are embedded at different spatial resolutions or downsampled representations of the image to improve resilience against resizing operations.

3.5.4 Color Channel Utilization

For color images, redundancy is introduced across luminance and chrominance channels. Stronger embedding is applied to the luminance channel, while weaker redundant embedding is used in chrominance channels.

3.6 Embedding Control and Parameter Selection

This section describes the parameters governing embedding strength, fragment placement, and redundancy levels.

3.6.1 Key-Based Fragment Placement

A secret key is used to pseudo-randomly determine fragment locations and embedding domains, enhancing security and preventing unauthorized extraction.

3.6.2 Embedding Strength Adaptation

Embedding strength is adjusted based on local image characteristics, such as texture intensity and frequency stability.

3.7 Damage-Aware Extraction Process

Unlike traditional extraction schemes, the proposed framework does not assume intact data. Instead, it explicitly estimates damage and adapts the extraction process accordingly.

3.7.1 Fragment Detection and Validation

Extracted fragments are validated using checksums and structural consistency checks. Invalid or severely corrupted fragments are discarded or assigned low confidence.

3.7.2 Confidence Scoring Mechanism

Each fragment is assigned a confidence score based on extraction quality, consistency with neighboring fragments, and agreement with relational information.

3.7.3 Adaptive Fragment Selection

Fragments with higher confidence scores are prioritized during reconstruction. Lower-confidence fragments are used only when necessary.

3.8 Message Reconstruction and Self-Recovery

The final message is reconstructed using surviving fragments and their encoded relationships.

3.8.1 Partial Reconstruction Strategy

Missing fragments are reconstructed using relational summaries and redundancy information provided by neighboring fragments.

3.8.2 Failure Conditions

This section defines conditions under which full or partial recovery is not possible, such as excessive data loss or severe compound attacks.

3.9 Computational Complexity and Practical Considerations

The computational overhead of the proposed framework is analyzed in terms of embedding time, extraction time, and memory requirements.

3.9.1 Complexity Analysis

The complexity of payload structuring, embedding, and extraction is discussed qualitatively.

3.9.2 Implementation Constraints

Practical limitations related to image size, payload capacity, and real-world deployment are identified.

3.10 Summary of the Proposed Framework

This section summarizes the key design principles and highlights how the proposed framework addresses the research gaps identified in Chapter 2, particularly synchronization loss, partial data destruction, and robustness under compound attacks.

Chapter 4

Implementation and Experimental Setup

4.1 Development Environment

The tools and software environment used for implementation are described.

4.2 Dataset Description

The image datasets used for experimentation are presented.

4.3 Attack Simulation

This section describes how image degradation and tampering scenarios are simulated.

4.4 Evaluation Metrics

4.4.1 Imperceptibility Metrics

PSNR and SSIM are introduced.

4.4.2 Robustness Metrics

Bit Error Rate (BER) and Normalized Correlation (NC) are defined.

Chapter 5

Results and Discussion

5.1 Imperceptibility Evaluation

Visual quality and distortion analysis are presented.

5.2 Robustness Under Image Degradation

Performance under compression, noise, and resizing is analyzed.

5.3 Robustness Under Partial Data Loss

The impact of cropping and localized tampering is evaluated.

5.4 Comparative Analysis

The proposed framework is compared with classical steganography methods.

5.5 Discussion of Results

Strengths, weaknesses, and observed trade-offs are discussed.

Chapter 6

Conclusion and Future Work

6.1 Summary of Findings

The main outcomes of the study are summarized.

6.2 Limitations

Practical and theoretical limitations of the framework are discussed.

6.3 Future Research Directions

Possible extensions and improvements are suggested.

Bibliography

[1] Ingemar Cox, Matthew Miller, Jeffrey Bloom, Jessica Fridrich, and Ton Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann, 2nd edition, 2007.

[2] Fabien A. P. Petitcolas, Ross J. Anderson, and Markus G. Kuhn. Information hiding—a survey. *Proceedings of the IEEE*, 87(7):1062–1078, 1999.

[3] Zhaoyang Jia, Han Fang, and Weiming Zhang. MBRS: Enhancing robustness of DNN-based watermarking by mini-batch of real and simulated JPEG compression. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*, pages 41–49, 2021.

[4] Eric Wengrowski and Kristin Dana. Light field messaging with deep photographic steganography. pages 1515–1524, 2019.

[5] T. Moerland. Steganography and steganalysis. Technical Report 1, Leiden Institute of Advanced Computing Science, 2005. Available at: `https://www.liacs.nl/home/tmoerl/privtech.pdf`.

[6] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Prentice Hall, 3rd edition, 2008.

[7] John G. Proakis and Masoud Salehi. *Digital Communications*. McGraw-Hill, New York, NY, USA, 5th edition, 2008.

[8] Tayana Morkel, Jan HP Eloff, and Martin S Olivier. An overview of image steganography. Number 2, 2005.

[9] Jessica Fridrich and Jan Kodovský. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 7(3):868–882, 2012.

[10] Niels Provos and Peter Honeyman. Hide and seek: An introduction to steganography. In *IEEE Security & Privacy*, pages 32–44, 2003.

[11] Neil F. Johnson and Sushil Jajodia. Exploring steganography: Seeing the unseen. *IEEE Computer*, 31(2):26–34, 1998.

[12] Stéphane Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.

[13] Po-Yueh Chen and Hung-Ju Lin. A dwt based approach for image steganography. *International Journal of Applied Science and Engineering*, 4(3):275–290, 2006.

[14] Gregory K. Wallace. The jpeg still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992.

[15] Stephane G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[16] Ronald N. Bracewell. *The Fourier Transform and Its Applications*. McGraw-Hill, 2000.

[17] J. J. K. Ó Ruanaidh and T. Pun. Rotation, scale and translation invariant digital image watermarking. In *IEEE International Conference on Image Processing*, 1998.

[18] V. Solachidis and I. Pitas. Circularly symmetric watermark embedding in 2-d dft domain. In *IEEE Transactions on Image Processing*, volume 10, pages 1741–1753, 2001.

[19] Lisa M. Marvel, Charles G. Boncelet, and Charles T. Retter. Spread spectrum image steganography. *IEEE Transactions on Image Processing*, 8(8):1075–1083, 1999.

[20] Alan V. Oppenheim and Jae S. Lim. The importance of phase in signals. *Proceedings of the IEEE*, pages 529–541, 1981.

[21] Peter Kovesi. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3):1–26, 1999.

[22] I. Pitas. A method for signature casting on digital images. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 215–218, 1996.

[23] Fredric J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE*, 66(1):51–83, 1978.

[24] James W. Cooley and John W. Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, 19(90):297–301, 1965.

[25] Christine I. Podilchuk and Wenjun Zeng. Image-adaptive watermarking using visual models. *IEEE Journal on Selected Areas in Communications*, 16(4):525–539, 1998.

[26] Brian Chen and Gregory W. Wornell. Quantization index modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, 47(4):1423–1443, 2001.

[27] M. Barni, F. Bartolini, V. Cappellini, and A. Piva. A dct-domain system for robust image watermarking. *Signal Processing*, 66(3):357–372, 1998.

[28] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, 2nd edition, 1999.

[29] Martin Kutter. Digital signature of color images using amplitude modulation. In *Storage and Retrieval for Image and Video Databases*, volume 3022, pages 518–526, 1999.

[30] V. Solachidis and I. Pitas. Circularly symmetric watermark embedding in 2-d fourier domain. *IEEE Transactions on Image Processing*, pages 1741–1753, 2004.

[31] Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, and Ton Kalker. *Digital Watermarking and Steganography*. Morgan Kaufmann, 2nd edition, 2007.

[32] Mauro Barni, Franco Bartolini, and Alessandro Piva. Multichannel watermarking of color images. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(2):142–156, 2002.

[33] Neil F. Johnson and Sushil Jajodia. Steganography: Seeing the unseen. In *IEEE Computer*, pages 26–34, 1998.

[34] Andreas Westfeld. F5 — a steganographic algorithm: High capacity despite better steganalysis. pages 289–302, 2001.

[35] Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. volume 14, pages 1181–1193, 2019.

[36] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. volume 23, pages 708–712, 2016.

[37] Mehdi Yedroudj, Frederic Comby, and Marc Chaumont. Yedroudj-net: An efficient cnn for spatial steganalysis. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2092–2096, 2018.

[38] Jamie Hayes and George Danezis. Generating steganographic images via adversarial training. In *Advances in Neural Information Processing Systems*, 2017.

[39] Jian Ye, Jiangqun Ni, Yang Yi, and Dengpan Ye. Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security*, 12(11):2545–2557, 2017.

[40] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 682–697, Cham, Switzerland, 2018. Springer.

[41] Shumeet Baluja. Hiding images in plain sight: Deep steganography. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

[42] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 27, 2014.

[43] Kevin Alex Zhang, Alfredo Cuesta-Infante, Lei Xu, and Kalyan Veeramachaneni. SteganoGAN: High capacity image steganography with GANs. In *arXiv preprint arXiv:1901.03892*, 2019.

[44] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4700–4708, 2017.

[45] Souvik Das, Chia-Yu Lin, and Hung-Yu Wei. ReDMark: Framework for residual diffusion watermarking based on deep neural networks. In *Expert Systems with Applications*, volume 166, page 114085. Elsevier, 2021.

[46] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015.

[47] Richard Shin and Dawn Song. JPEG-resistant adversarial images. In *NIPS 2017 Workshop on Machine Learning and Computer Security*, 2017.

[48] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015.

[49] Chaoning Zhang, Philipp Benz, Adil Karjauv, Geng Sun, and In So Kweon. Udh: Universal deep hiding for steganography, watermarking, and light field messaging. *Advances in Neural Information Processing Systems*, 33:10223–10234, 2020.

[50] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. *Density estimation using Real-valued Non-Volume Preserving (Real NVP) transformations.* 2017.

[51] Xiyang Luo, Ruohan Zhan, Chi-Hao Chang, Feng Liu, and Prasant Mohapatra. Distortion agnostic deep watermarking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13548–13557, 2020.

[52] Chaoning Zhang, Philipp Benz, Adil Karjauv, Geng Sun, and In So Kweon. UDH: Universal deep hiding for steganography, watermarking, and light field messaging. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 10223–10234, 2020.

[53] Mahdi Ahmadi, Alireza Norouzi, Nader Karimi, Shadrokh Samavi, and Ali Emami. ReDMark: Framework for residual diffusion watermarking based on deep neural networks. *Expert Systems with Applications*, 146:113157, 2020.

[54] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Ted Furon. The stable signature: Rooting watermarks in latent diffusion models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[55] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.