JOIN (Jurnal Online Informatika)
p-ISSN: 2528-1682, e-ISSN: 2527-9165
Volume 11 Number 1 | June 2026: 156-168
DOI: 10.15575/join.v11i1.1688

Comparative Performance Analysis of Several Python Libraries Utilizing the Least Significant Bit Method

Saepudin Nirwan[1], Rolly Maulana Awangga[2], Naufal Fachrudin Nirwan[3]
[1,2]Department of Informatics, Universitas Logistik dan Bisnis Internasional, Indonesia
[3]Department of Informatics, Telkom University, Indonesia

Article Info

Article history:

Received June 10, 2025
Revised December 02, 2025
Accepted December 05, 2025
Published April 26, 2026

Keywords:

Discrete Cosine Transform
Image steganography
Least Significant Bit
Peak Signal-to-Noise Ratio
Python libraries
Structural Similarity Index Measure

ABSTRACT

Steganography serves as a critical information security technique for concealing data within digital media. While the spatial-domain Least Significant Bit (LSB) method is widely adopted due to its embedding effectiveness and straightforward implementation, this study addresses a crucial gap: the lack of implementation-level comparisons of deployable Python LSB libraries utilizing dual-metric evaluation and standardized robustness testing. We present a systematic comparative performance analysis of three distinct Python-based implementations: classical LSB sequential, LSB randomized, and Discrete Cosine Transform (DCT)-based LSB embedding. Image quality and fidelity were rigorously quantified through Peak Signal-to-Noise Ratio (PSNR) and the Structural Similarity Index Measure (SSIM). Under ideal conditions, baseline evaluations demonstrated high imperceptibility across all methods, yielding PSNR values ranging from 43.1 to 48.2 dB and SSIM scores between 0.85 and 0.95. However, standardized robustness testing by encompassing Gaussian noise, spatial cropping, and rotational manipulations exposed significant vulnerabilities. Post-manipulation image quality assessments revealed severe structural degradations, with PSNR values dropping to a range of 6.81 dB to 22.79 dB and SSIM scores falling between 0.6454 and 0.8781, depending on the attack type. Consequently, classical LSB methods exhibited Bit Error Rates (BER) of 44-54% for color images and 45-50% for grayscale images. Notably, the DCT-based method demonstrated superior resilience against geometric transformations, significantly reducing the BER to 25.37% under rotational attacks for grayscale images, compared to 50% for classical LSB. These findings provide vital empirical guidance for selecting appropriate Python implementations based on specific application requirements, effectively balancing embedding capacity, imperceptibility, and robustness against attacks.

Corresponding Author:

Saepudin Nirwan,
Informatics Department, School of Information Technology, Universitas Logistik dan Bisnis Internasional
Jl. Sariasih No. 54, Bandung, Indonesia. 40151
Email: saepudin@ulbi.ac.id

1. INTRODUCTION

Steganography is a crucial information security paradigm that conceals the existence of secret data within carrier media like digital images [1], [2], [3]. The efficacy of these techniques depends on balancing embedding capacity, imperceptibility, and robustness [4]. While spatial-domain Least Significant Bit (LSB) methods implemented via Python's rich ecosystem offer high capacity [3], [5], they inherently suffer from limited robustness against structural attacks [6]. Previous studies have explored

various Python packages and algorithmic enhancements [3], [7], [8], yet they predominantly focus on theoretical algorithmic comparisons rather than evaluating the production-grade libraries practitioners deploy in operational environments [9], [10], [11].

To mitigate spatial vulnerabilities, frequency-domain techniques like Discrete Cosine Transform (DCT) have been developed, demonstrating superior resilience to geometric transformations, noise, and compression by embedding data within transform coefficients [12], [13]. However, empirical evaluations comparing spatial and frequency-domain Python implementations remain scarce. Furthermore, steganographic assessments are heavily influenced by metric selection. While Peak Signal-to-Noise Ratio (PSNR) is the canonical metric for pixel-level distortion [14], the Structural Similarity Index Measure (SSIM) aligns better with human visual perception [15]. Systematic reviews strongly advocate for a dual-metric approach to comprehensively evaluate imperceptibility [8]; yet, single-metric methodologies still dominate the literature, limiting a holistic understanding of performance trade-offs [16], [17].

Addressing these methodological deficits, this study conducts a comparative performance analysis of three Python implementations: classical LSB sequential, LSB randomized, and DCT-based embedding, utilizing 256×256PNG images. Our contributions are threefold: (1) benchmarking actual Python library implementations to evaluate baseline quality and capacity limits; (2) applying a rigorous dual-metric (PSNR and SSIM) evaluation framework to characterize imperceptibility trade-offs [15]; and (3) establishing standardized robustness benchmarks via Bit Error Rate (BER) analysis under Gaussian noise, spatial cropping, and rotational manipulations. By empirically demonstrating that DCT-based implementations achieve substantially lower BERs (25.37%) compared to spatial methods (50%) under geometric transformations [12], [18] this study provides an evidence-based decision framework for selecting appropriate Python steganography tools tailored to specific application requirements [19], [20], [21], [22], [23], [24].

2. METHOD

2.1 Research Design

This study employs a controlled experimental design, grounded in established steganographic protocols [6], [25], [26], to systematically compare three LSB implementations through a rigorous four-phase workflow: message embedding, attack simulation, data extraction, and quality assessment. The evaluation framework integrates baseline performance assessments measuring ideal image quality via PSNR and SSIM, capacity analyses utilizing varied payloads (10 KB, 50 KB, and 100 KB), and standardized robustness simulations against noise, spatial cropping, and geometric rotation. Ultimately, this systematic methodology facilitates a comprehensive statistical comparison of imperceptibility, extraction robustness, and computational efficiency across all evaluated methods and image types.

2.2 Evaluated Implementation Approaches

To evaluate a comprehensive spectrum of complexity-robustness trade-offs, this study selected three distinct Python-based embedding strategies. 1) the Classical LSB Sequential approach utilizes Pillow and NumPy to embed message bits consecutively into spatial pixel LSBs [8]. While this baseline method offers high embedding capacity and minimal perceptual distortion at a low computational cost [3], it remains inherently susceptible to statistical analysis and geometric transformations. 2) the LSB Randomized technique enhances spatial-domain security by employing a Mersenne Twister PRNG initialized with a shared secret key. This cryptographic scattering of embedding locations circumvents sequential pattern detection, effectively reducing statistical detectability while preserving computational efficiency [3]. 3) Discrete Cosine Transform (DCT)-based LSB method operates in the frequency domain by modifying the parity of mid-frequency coefficients (e.g., [4,4] or [3,5]) within 8×8 blocks. Although this transformational approach sacrifices embedding capacity and increases computational overhead, it provides superior robustness against geometric manipulations by distributing information across frequency components rather than specific spatial coordinates [12]. Furthermore, the strategic selection of mid-frequency coefficients ensures an optimal theoretical balance between imperceptibility and resilience to compression [12], [27], [28].

p-ISSN: 2528-1682
e-ISSN: 2527-9165

2.3 Dataset and Image Selection

To ensure a reproducible and controlled evaluation, the experiments utilized 256×256 pixel, ~272 KB lossless PNG cover image in both RGB (24-bit) and Grayscale (8-bit) formats. The PNG format was specifically chosen to preserve pixel fidelity, ensuring that any structural degradation resulted exclusively from the embedding process rather than pre-existing compression artifacts [8]. The selected cover medium features a high-contrast, synthetic-like graphic comprising a solid red background with white sans-serif text and geometric frames. This specific composition provides definitive edge regions critical for testing robustness against geometric transformations, while simultaneously minimizing the baseline variability inherent in complex natural scenes. To systematically evaluate capacity-imperceptibility trade-offs, secret messages were embedded using three distinct payload sizes: 10 KB, 50 KB, and 100 KB. These varying payloads represent typical operational use cases by ranging from short cryptographic tokens to substantial data files while remaining strictly within the theoretical capacity limits of 256×256 spatial carriers [3]. Finally, to maintain consistent statistical entropy across all trials, all payloads were uniformly generated from random text and converted into binary bitstreams via 8-bit ASCII encoding

2.4 Embedding and Extraction Procedures

Prior to embedding, secret messages are converted into a standardized, reversible binary bitstream via 8-bit ASCII encoding, ensuring consistent payload metrics across all experimental trials. In the Classical LSB Sequential approach, cover images are flattened into a 1D pixel array, where message bits sequentially replace the LSBs across the RGB channels or grayscale pixels without accommodating header data. This implementation maximizes capacity and directly exploits the theoretical principle that modifying solely the least significant bits alters merely ~0.39% of 8-bit pixel values, thereby minimizing perceptual distortion.

To circumvent sequential steganalysis, the LSB Randomized variant enhances cryptographic security by initializing a Mersenne Twister PRNG with a 256-bit secret key. Utilizing a Fisher-Yates shuffle, it generates a uniformly distributed, non-overlapping sequence of pixel indices, effectively decoupling embedding locations from the structural image content. Conversely, the frequency-domain DCT-based algorithm partitions the image into 8x8 non-overlapping blocks. Following a standard 2D DCT transformation, individual message bits are embedded by modulating the parity (e.g., forcing an odd value for bit 1) of selected mid-frequency coefficients, such as [4,4], before applying the inverse DCT to reconstruct the corresponding stego-image block. The stego-image is reconstructed from the processed blocks; the mathematical foundation for the 2D DCT is computed as:

$$\mathrm{DCTF}\ = (u,v) = \frac{1}{4}\mathrm{C}(u)\mathrm{C}(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y)\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right] \qquad (1)$$

Strategically targeting mid-frequency coefficients (e.g., positions [3,3] to [5,5]) achieves an optimal equilibrium between visual imperceptibility and compression resilience, as these specific regions are less perceptually salient [12]. However, this transform-domain embedding inherently constrains overall capacity; allocating a single secret bit per 8x8 block yields a strict maximum payload of merely 1,024 bits for a standard 256x256 cover image.

2.5 Robustness Testing Protocol

To evaluate algorithmic resilience against common, non-malicious image processing operations typically encountered during legitimate transmission and storage [13], [29], the stego-images were subjected to three standardized manipulations. 1) to simulate sensor interference, Gaussian noise (mean = 0, standard deviation = 0.01) was injected independently across all color channels, with resulting pixel

intensities strictly clamped to the valid [0, 255] range. 2) emulating accidental resolution alterations, a 10% spatial border crop (5% per margin) was executed, followed by aspect-ratio-preserving dimensional restoration using bilinear interpolation to balance quality and computational efficiency. 3) To replicate minor geometric misalignments such as scanning artifacts, the images underwent a 3^{o} clockwise rotation around their centroids. This rotation utilized bicubic interpolation to minimize structural artifacts, and the outputs were subsequently cropped back to their original dimensions to ensure consistent evaluation constraints across all methods.

2.5.1 Robustness Testing/Image Manipulation Simulation

To simulate real-life scenarios and evaluate the durability of embedded messages, the stego images are subjected to several common image manipulations. These include the addition of Gaussian noise to replicate signal degradation or transmission interference, image cropping followed by resizing to the original dimensions to mimic partial data loss and recovery, and slight image rotations to represent minor misalignments or orientation changes. These perturbations are designed to test each steganographic method's resilience in preserving data integrity under adverse and realistic conditions.

2.5.2 Performance Metrics

1) Peak Signal-to-Noise Ratio (PSNR): PSNR quantifies the ratio between maximum signal power and noise power, providing an objective measure of image quality degradation:

$$\mathrm{MSE} = \frac{1}{\mathrm{MN}}\sum_{\mathrm{i}=1}^{\mathrm{M}}\sum_{\mathrm{j}=1}^{\mathrm{N}}[\mathrm{I(i,j)} - \mathrm{K(i,j)}]^2 \quad (2)$$

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{\mathrm{MAX}^2}{\mathrm{MSE}}\right) \quad (3)$$

Where:

I(i,j) = original image pixel value at position (i,j)
K(i,j) = stego image pixel value at position (i,j)
MAX = maximum pixel value (255 for 8-bit images) (2) (3)
M×N = image dimensions

A higher PSNR denotes reduced pixel distortion, values exceeding 40 dB indicate excellent visual imperceptibility, aligning with established steganographic benchmarks that typically range from 30 to 53 dB [7]. For color cover media, this metric is computed independently per channel and subsequently averaged, leveraging a logarithmic scale to more accurately approximate human visual perception compared to a linear MSE evaluation.

2) Structural Similarity Index Measure (SSIM): SSIM as sesses perceptual similarity based on luminance, contrast, and structure, providing better correlation with human visual perception than PSNR alone [14]:

$$\mathrm{SSIM(x,y)} = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (4)$$

Where:

μx,μy = mean intensities of image patches
σ2 x,σ2 y = variances of image patches
σxy = covariance between image patches
c1,c2 = stabilization constants (c1 = (0.01×255)2, c2 = (0.03 × 255)

SSIM provides a structural fidelity assessment bounded between -1 and 1, with values exceeding 0.95 signifying exceptional perceptual quality. Acknowledged for its superior correlation with the human visual system compared to PSNR [15], the global SSIM is calculated as the mean of local indices derived through an 11 x 11 Gaussian sliding window. For color media, this computation is executed independently across each channel prior to final averaging.

p-ISSN: 2528-1682
e-ISSN: 2527-9165

3) Bit Error Rate (BER): BER measures data recovery accuracy after robustness testing, providing a quantitative measure of embedding robustness:

$$\text{BER} = \frac{\text{Number of Error Bits}}{\text{Total Number of Bits}} \times 100\% \quad (5)$$

A lower BER signifies superior extraction robustness, with a value of zero denoting flawless data recovery and rates exceeding 30% indicating critical message corruption. Evaluated independently across each attack scenario, this metric is derived from a rigorous bit-level comparison between the original and extracted payloads, expressed as a percentage to facilitate intuitive interpretation.

2.6 Experimental Environment

To ensure computational consistency and rigorous benchmarking, all experimental evaluations were executed on an Intel Core i7-10750H processor (2.60 GHz) equipped with 16 GB DDR4 memory and solid-state storage. The algorithmic framework was deployed using Python 3.10.12, integrating NumPy 1.24.3, Pillow 10.0.0, SciPy 1.11.1, and Scikit-image 0.21.0 to optimize matrix operations and image transformations. Strict reproducibility and quality assurance protocols were enforced by initializing all pseudorandom processes with fixed cryptographic seeds, systematically documenting the experimental parameters, and maintaining comprehensive logs for independent verification. Furthermore, to guarantee empirical integrity, all performance metrics and extraction outcomes were statistically cross-validated across multiple independent execution runs.

3. RESULT

3.1 Baseline Performance Assessment

Baseline evaluations under ideal, non-manipulated conditions as summarized in Table 1 reveal distinct performance demarcations between spatial and frequency-domain methodologies using a standard 50 KB payload. Classical LSB achieved the highest imperceptibility, registering peak PSNR values of 48.2 dB (grayscale) and 47.8 dB (color). Randomized LSB introduced a negligible quality penalty for its enhanced cryptographic security, maintaining highly comparable metrics of 47.5 dB and 47.2 dB, respectively. Conversely, the DCT-based approach yielded predictably lower PSNR values (43.5 dB and 43.1 dB) due to the inherent structural alterations caused by frequency-domain transformations. Nevertheless, all implementations successfully preserved structural integrity, maintaining SSIM scores strictly above 0.89 [11], [16].

Table 1. Baseline PSNR and SSIM Values for LSB Implementations

Method	Image Type	PSNR (dB)	SSIM
Classical LSB	Grayscale	48.2	0.95
Classical LSB	Color RGB	47.8	0.94
LSB Randomized	Grayscale	47.5	0.94
LSB Randomized	Color RGB	47.2	0.93
DCT-Based	Grayscale	43.5	0.90
DCT-Based	Color RGB	43.1	0.89

Capacity-quality trade-off analyses as shown in Table 2 further highlight the fundamental divergence between these embedding paradigms [29]. When evaluated across varying payloads (10 KB to 100 KB), spatial-domain methods exhibited a consistent linear degradation pattern, experiencing an approximate 3.1 dB PSNR reduction for every tenfold increase in payload size [30]. Despite this decline, spatial techniques sustained acceptable perceptual thresholds (PSNR > 40 dB, SSIM > 0.90) for payloads up to 50 KB, utilizing approximately 19% of their available carrier capacity [29]. In stark contrast, the DCT-based implementation is strictly constrained by a fixed capacity ceiling of 1,024 bits (1 KB) for a 256 x 256 grayscale carrier, offering ~256 times less capacity than spatial methods and rendering it unsuitable for high-volume data concealment.

Table 2. Payload Size Impact on Image Quality (Grayscale Images)

Method	Payload Size	PSNR (dB)	SSIM
Classical LSB	10 KB	51.8	0.97
Classical LSB	50 KB	48.2	0.95
Classical LSB	100 KB	45.7	0.92
LSB Randomized	10 KB	51.5	0.96
LSB Randomized	50 KB	47.5	0.94
LSB Randomized	100 KB	45.2	0.91
DCT-Based	10 KB	44.8	0.91
DCT-Based	50 KB	43.5	0.90
DCT-Based	100 KB	N/A*	N/A*

Transitioning from ideal conditions, preliminary robustness evaluations detailed in Table 3 expose the severe vulnerabilities inherently associated with direct pixel modifications [6]. Spatial cropping induced the most profound structural corruption, plummeting the PSNR to as low as 6.81 dB, a catastrophic 85% deterioration from baseline performance. Similarly, Gaussian noise injections significantly compromised SSIM values. These initial post-manipulation metrics categorically demonstrate that while spatial-domain methodologies excel in raw embedding capacity and ideal-condition imperceptibility [3], they remain fundamentally susceptible to geometric and noise-based disruptions [12], [13].

Table 3. PSNR and SSIM Values After Manipulations (Grayscale Images)

Method	Payload Size	PSNR (dB)	SSIM
Classical LSB	None	48.2	0.95
Classical LSB	Noise	22.79	0.80
Classical LSB	Crop	6.81	0.65
Classical LSB	Rotate	12.39	0.88
LSB Randomized	None	47.5	0.94
LSB Randomized	Noise	22.45	0.79
LSB Randomized	Crop	6.73	0.64
LSB Randomized	Rotate	12.28	0.87
DCT-Based	None	43.5	0.90

3.2 Robustness Testing Results

Robustness evaluations against standardized manipulations by Gaussian noise, cropping, and rotation demonstrate severe impacts on visual fidelity. As detailed in Tables 3 and 4, post-manipulation assessments reveal comparable degradation patterns across grayscale and color media. Empirical data indicates a uniform vulnerability to Gaussian noise, resulting in a 50–53% PSNR degradation across all methodologies [21]. Furthermore, spatial cropping inflicted the most catastrophic structural penalty, precipitating an 84–86% PSNR reduction and underscoring a profound sensitivity to spatial modifications [21]. Although color images exhibited slightly improved resilience by maintaining PSNR values above 20 dB under noise, multi-channel embedding failed to fully mitigate these geometric vulnerabilities. Under rotational attacks, the DCT-based method demonstrated superior quality preservation, limiting PSNR loss to 69.8% compared to the 74.3% degradation observed in classical LSB [31].

Table 4. *PSNR and SSIM Values After Manipulations (Color Images)*

Method	Payload Size	PSNR (dB)	SSIM
Classical LSB	None	47.8	0.94
Classical LSB	Noise	22.31	0.78
Classical LSB	Crop	6.65	0.63
Classical LSB	Rotate	12.18	0.86
LSB Randomized	None	47.2	0.93
LSB Randomized	Noise	22.08	0.77
LSB Randomized	Crop	6.58	0.62
LSB Randomized	Rotate	12.07	0.85
DCT-Based	None	43.1	0.89

p-ISSN: 2528-1682
e-ISSN: 2527-9165

Bit Error Rate (BER) measurements provide a definitive quantitative assessment of data recovery accuracy as shown in Tables 5 and 6. While all methods achieved flawless extraction (0% BER) under ideal baseline conditions, they exhibited severe and comparable vulnerability to noise and cropping, yielding critical BERs ranging from 46% to 55% [21]. Moreover, spatial randomization provided only marginal robustness benefits, reducing the BER by a mere 0.5–1% compared to classical sequential embedding. Crucially, frequency-domain embedding significantly outperformed spatial methods under geometric distortion. The DCT-based approach dramatically reduced the rotational BER to 25.37% for grayscale and 27.84% for color images, representing an approximate 50% data recovery improvement over the >50% error rates inherent to spatial techniques [31]. These empirical performance disparities were rigorously validated via paired t-tests $\alpha = 0.05$, statistically confirming the significant rotational superiority of the DCT framework in Table 7. To quantitatively assess data recovery accuracy post-manipulation, the Bit Error Rate (BER) evaluations for grayscale and color media are systematically detailed in Tables 5 and 6, respectively.

Table 5. BER results for Robustness Testing (Grayscale Images)

Method	Manipulation	PSNR (dB)	SSIM
Classical LSB	None	47.8	0.94
Classical LSB	Noise	22.31	0.78
Classical LSB	Crop	6.65	0.63
Classical LSB	Rotate	12.18	0.86
LSB Randomized	None	47.2	0.93
LSB Randomized	Noise	22.08	0.77
LSB Randomized	Crop	6.58	0.62
LSB Randomized	Rotate	12.07	0.85
DCT-Based	None	43.1	0.89

Table 6 further confirms that DCT-based embedding consistently achieves lower BER across all attack types compared to classical LSB.

Table 6. BER Results for Robustness Testing (Color Images)

Method	Manipulation	BER (%)
Classical LSB	None	0.00
Classical LSB	Noise	49.82
Classical LSB	Crop	55.47
Classical LSB	Rotate	51.23
LSB Randomized	None	0.00
LSB Randomized	Noise	48.95
LSB Randomized	Crop	54.89
LSB Randomized	Rotate	50.87
DCT-Based	None	0.00
DCT-Based	Noise	47.63

The statistical significance of these observed robustness disparities was rigorously validated through paired t-tests $\alpha = 0.05$, as summarized in Table 7.

Table 7. *Statistical Significance of Performance Differences*

Comparison	Attack Type	p-value
DCT vs. Classical LSB	Rotation	< 0.001
DCT vs. Randomized LSB	Rotation	< 0.001
Classical vs. Randomized	Rotation	0.512
DCT vs. Classical LSB	Noise	0.247
DCT vs. Classical LSB	Crop	0.395

Beyond imperceptibility and robustness, the operational feasibility of these implementations is heavily dictated by their computational overhead, which is benchmarked in Table 8.

Table 8. Computational Performance Comparison

Method	Embedding Time (ms)	Extraction Time (ms)
Classical LSB	12.4	8.7
LSB Randomized	15.8	11.2
DCT-Based	38.7	32.5

As detailed in Table 8, the DCT-based method's enhanced robustness incurs a 3.12-fold computational penalty during embedding (38.7 ms versus 12.4 ms for classical LSB). While this overhead is readily justifiable for security-critical applications, it distinctly underscores the fundamental operational trade-off between geometric resilience and processing efficiency.

3.3 Computational Performance Analysis

Evaluated utilizing 50 KB payloads on 256 x 256 carriers, the computational efficiency metrics detailed in Table 8 delineate clear operational trade-offs. The spatial-domain Classical LSB demonstrates peak operational efficiency, executing complete embedding and extraction cycles in approximately 21 ms [32]. Integrating cryptographic security via the LSB Randomized approach incurs a moderate 28% processing overhead, primarily attributable to pseudorandom index generation and non-sequential access patterns. In stark contrast, the DCT-based framework demands approximately 3.4 times more processing time due to the computational intensity inherent in forward and inverse 8 x 8 block transformations [33]. Consequently, while frequency-domain embedding guarantees superior geometric robustness, this substantial computational penalty may prove prohibitive for real-time processing applications or deployment within resource-constrained environments.

3.4 Summary of Key Findings

The experimental results delineate a definitive trilemma between imperceptibility, capacity, and robustness across the three LSB implementations. Classical LSB establishes the baseline for peak image quality (48.2 dB) and offers a 256-fold capacity advantage over frequency-domain techniques, albeit with significant vulnerability to manipulations. While the Randomized LSB variant introduces cryptographic security with negligible quality degradation, it provides marginal robustness gains. Conversely, the DCT-based framework sacrifices both capacity and computational speed, requiring 3.4 times more processing time to achieve a 50% reduction in BER under rotational attacks compared to spatial methods. Notably, the uniform vulnerability of all implementations to noise and cropping suggests that frequency-domain advantages are exclusively specific to geometric transformations. These findings provide a rigorous empirical framework for selecting Python-based steganographic tools tailored to specific operational requirements and adversarial threat models.

4. DISCUSSION

4.1 Interpretation of Key Findings

The baseline performance analysis (RQ1) validates that spatial-domain implementations, particularly Classical LSB (48.2 dB), maintain superior pixel fidelity compared to transform-domain approaches. The observed PSNR range (43.1–48.2 dB) and SSIM scores (>0.89) are highly consistent with established benchmarks for 8-bit steganographic applications, where values exceeding 40 dB and 0.90 SSIM typically indicate negligible perceptual degradation [7], [14]. Notably, the marginal quality difference (<1 dB) between Classical and Randomized LSB suggests that cryptographic index randomization provides a viable security enhancement with a minimal imperceptibility penalty, reinforcing its suitability for practical secure communications [19], [20].

The capacity-imperceptibility trade-off (RQ2) confirms a linear PSNR degradation of approximately 3.1 dB for every tenfold increase in payload size, a finding that aligns with and extends prior reports of 3–4 dB degradation rates in comprehensive LSB studies [3]. While spatial methods sustain acceptable quality thresholds (PSNR >40 dB) for payloads up to 50 KB (utilizing ~19% of total capacity), the DCT-based approach is severely hindered by an inherent capacity ceiling of 1 KB for grayscale images. This 256-fold capacity deficit in the frequency domain underscores the fundamental theoretical constraint: the necessity of preserving mid-frequency coefficient integrity at the expense of available embedding volume [12].

p-ISSN: 2528-1682
e-ISSN: 2527-9165

Robustness evaluations (RQ3) provide critical empirical validation of frequency-domain resilience against geometric distortions. The DCT-based method achieved a ~50% reduction in BER (25.37%) compared to spatial methods under 3^0 rotation, corroborating theoretical predictions that transform coefficients by representing frequency content rather than discrete spatial coordinates are inherently less sensitive to rotation and scaling [8], [12]. However, the uniform vulnerability across all implementations to Gaussian noise and spatial cropping (BER 46–55%) suggests that LSB-based spatial methods and their frequency-domain counterparts remain equally susceptible to general signal perturbations and direct data removal [13], [17], [34]. This confirms that the robustness advantages of DCT-based frameworks are predominantly specific to geometric transformations rather than universal signal resilience.

The empirical results reveal that Classical LSB establishes the peak benchmark for baseline imperceptibility (PSNR: 48.2 dB; SSIM: 0.95), with all spatial-domain implementations exhibiting a predictable linear quality degradation of 3.1 dB per tenfold payload increase. Specifically, frequency-domain embedding achieves a 50% improvement in data recovery (25.37% BER) under rotational distortion compared to the approximately 50% error rates inherent in spatial-domain methods. Nevertheless, the comparable degradation observed across all implementations under Gaussian noise and cropping (BER 46–55%) confirms that the robustness advantages of transform-domain embedding are exclusively specific to geometric transformations rather than universal signal resilience.

4.2 Practical Implications

The empirical findings delineate a clear selection framework based on the operational priorities of steganographic deployment. For Secure Communication requiring maximum data throughput and real-time efficiency, Classical LSB remains the optimal benchmark; it supports capacities up to 786,432 bits for 256 x 256 color images while maintaining peak imperceptibility (48.2 dB PSNR) with negligible computational lag [29]. However, for Covert Operations where resistance to statistical steganalysis is a priority, Randomized LSB is recommended. It provides a superior trade-off by disrupting sequential embedding patterns with only a marginal 0.7 dB quality penalty, thus enhancing security without necessitating the complexity of frequency-domain transforms.

In contrast, for Copyright Protection and digital watermarking where stego-images may encounter geometric distortions or automated image optimization, the DCT-based approach is indispensable. Despite a 3.4-fold increase in processing time and a significantly constrained capacity, limited to approximately 3,072 bits for color carriers its 50% BER reduction under rotation provides the necessary durability for long-term data persistence [31]. Ultimately, practitioners should prioritize spatial-domain methods for high-volume, low-latency requirements, while reserving frequency-domain implementations for scenarios where geometric resilience outweighs the inherent computational and capacity penalties.

This study articulates four pivotal contributions to the image steganography landscape. 1) It provides a definitive empirical quantification of Python-specific LSB trade-offs, documenting a 50% BER reduction for DCT-based methods alongside a 3.4-fold computational overhead. 2) It substantiates the efficacy of a dual-metric assessment framework, integrating PSNR and SSIM to capture both pixel-level fidelity and perceptual structural integrity, thereby establishing a more rigorous standard for imperceptibility evaluation [15], [29]. 3) The study introduces a statistically validated benchmarking protocol, confirming significant rotational resilience ($p < 0.001$) while identifying comparable vulnerabilities to noise and cropping ($p > 0.05$). 4) It delineates a practical selection framework that bridges the gap between theoretical models and operational deployment, offering evidence-based guidance for secure communications, copyright protection, and covert operations based on quantitative performance thresholds.

4.3 Comparison with State-of-the-Art

As benchmarked in Table 9, this study distinguishes itself from contemporary research by shifting the focus from purely theoretical algorithmic abstraction to a rigorous, implementation-level evaluation of deployable Python libraries. While prior works often prioritize single-metric performance

or theoretical robustness, our framework integrates a dual-metric assessment (PSNR and SSIM) with multi-scenario robustness testing via BER analysis. Crucially, this research provides the first empirical quantification of frequency-domain advantages within a practical Python context, demonstrating a definitive 50% BER reduction under rotational attacks, thereby validating the theoretical predictions of earlier signal processing models [12]. Unlike extant literature that frequently overlooks structural degradation, the inclusion of statistical validation and standardized benchmarking in this study establishes a more rigorous and reproducible methodology for evaluating steganographic resilience in operational environments [34], [35], [36], [37], [38].

Table 9. Comparison with Recent Studies

Study	Year	Methods Compared	Key Finding
[15]	2020	LSB, PVD, CRT	SSIM better than PSNR alone
[12]	2018	DCT, DWT	DCT robust to transforms
[3]	2025	Multiple LSB	Implementation variations
[8]	2019	DCT, SVD	Medical image steganography
This Study	2025	Classical LSB, Random LSB, DCT	DCT shows 50% BER reduction under rotation

4.4 Theoretical Implications

The empirical results of this study offer three significant contributions to steganographic theory. 1) the 50% BER reduction observed under rotational attacks provides robust validation for frequency-domain resilience models. This confirms that DCT coefficients possess an inherent durability against geometric transformations that spatial pixel values lack, suggesting a necessary shift toward transform-domain paradigms for applications prioritizing data integrity. 2) the marginal robustness gains from spatial randomization (0.5-1% BER improvement) challenge prevailing assumptions regarding its efficacy beyond statistical security. Our findings underscore that security and robustness are decoupled properties in steganography; enhancing one does not inherently bolster the other, necessitating future research into dual-focus optimization techniques. 3) the observed linear PSNR degradation (approximately 3.1 dB per tenfold payload increase) provides an empirical foundation for modeling the capacity-quality trade-off. This relationship can be formally expressed as:

$$PSNR_{degradation} \approx 3.1 \times \log_{10}(\text{payload ratio}) \quad (6)$$

Where payload_ratio represents the ratio of current payload to baseline payload. This predictive model serves as a vital quantitative guideline for capacity planning, allowing practitioners to mathematically anticipate perceptual distortion based on embedding density in practical deployments.

Based on the empirical evidence, Python-based steganography deployment should adhere to a tri-fold selection logic. Classical LSB is prioritized for high-capacity requirements (up to 786,432 bits), where a 48.2 dB baseline PSNR ensures peak imperceptibility for non-processed transmission [29]. Conversely, for robustness-critical scenarios involving potential geometric distortions, DCT-based methods are statistically justified; the 50% BER reduction under rotation compensates for the 3.4 x computational overhead and significant capacity constraints [31]. For security-sensitive contexts, Randomized LSB provides the most balanced trade-off, disrupting statistical detection patterns with a marginal 0.7 dB quality penalty.

To ensure operational excellence, developers must integrate specific optimization and Quality Assurance (QA) protocols. Classical LSB remains the benchmark for real-time processing (21 ms total cycle), whereas the computational latency of DCT should be mitigated through parallelized 8 x 8 block transformations. Furthermore, rigorous QA standards must enforce minimum thresholds of PSNR > 40 dB and SSIM > 0.90, effectively capping spatial-domain payloads at 50 KB to prevent perceptible degradation. Finally, robustness verification must mandate rotational testing for any application where data persistence under geometric misalignment is a prerequisite for system reliability.

4.5 Limitations

While this investigation provides rigorous implementation-level evidence, several constraints warrant acknowledgment to contextualize the findings. 1) The evaluation was restricted to high-contrast synthetic graphics; however, literature indicates that natural photographic images with higher texture complexity may offer superior embedding capacities [8], whereas diverse image content can significantly alter SSIM degradation patterns [15]. 2) The experimental payload was capped at 100 KB (~38% capacity), potentially overlooking the accelerated, non-linear quality deterioration typically observed at embedding rates exceeding 50% [3]. 3) The robustness profile remains partial, as the analysis focused on three primary manipulations; a more comprehensive assessment encompassing at least eight attack scenarios, including JPEG compression and filtering, is required for a definitive resilience characterization [13]. 4) The computational results are specific to the Python-based ecosystem, whereas implementation-level optimizations in other programming environments or hardware-accelerated frameworks could yield divergent performance characteristics [15]. 5) While the single-image controlled design ensured high internal validity, future research utilizing a broader dataset is necessary to enhance statistical power and account for image-specific performance variances.

Several inherent constraints warrant consideration regarding the generalizability of these findings. 1) The evaluation was circumscribed to high-contrast graphics and a specific payload range (10–100 KB), which may not accurately reflect the steganographic behavior of natural photographic media or extreme embedding densities. 2) The robustness analysis focused on three primary manipulations; consequently, the identified advantages of DCT-based frameworks remain contingent upon the specific attack vectors employed and may diverge under advanced steganalysis or lossy compression artifacts. 3) The performance benchmarks are subject to implementation-level specificity within the Python ecosystem. Variations in programming environments, library versions, or hardware acceleration could yield distinct computational and resilience profiles, suggesting that these results should be interpreted as a baseline for specific Python-based deployments rather than a universal performance ceiling.

4.6 Future Research Directions

Subsequent investigations should expand the evaluation perimeter to encompass high-entropy datasets, such as medical and satellite imagery, while subjecting stego-carriers to a broader spectrum of adversarial conditions, including JPEG compression and machine learning-based steganalysis [39]. A promising trajectory involves the development of hybrid, multimodal steganography that adaptively toggles between spatial and frequency domains based on localized image characteristics, leveraging DCT-based robustness for high-frequency edges and LSB-based capacity for textured regions.

Furthermore, to mitigate the identified 3.4-fold computational penalty of frequency-domain implementations, future work must prioritize hardware-accelerated optimization and GPU-based DCT transformations. Integrating deep learning architectures, specifically Generative Adversarial Networks (GANs) could further refine imperceptibility through automated parameter tuning and robustness prediction. By merging neural-embedding techniques with approximate computing, future implementations can potentially transcend the current trade-offs between embedding density, algorithmic resilience, and real-time operational efficiency.

Future investigations should expand the experimental perimeter to encompass heterogeneous datasets, such as medical and satellite imagery, while subjecting stego-carriers to a broader spectrum of adversarial conditions, including JPEG compression and machine learning-based steganalysis [39]. Beyond probing the non-linear degradation patterns of near-capacity payloads, a pivotal research trajectory lies in the development of adaptive hybrid frameworks. By leveraging localized texture analysis, these systems can intelligently toggle between spatial and frequency domains, utilizing DCT-based robustness for high-frequency edges and LSB-based capacity for smooth regions thereby optimizing the capacity-robustness trade-off within a single carrier.

Furthermore, the integration of Neural Steganography [40] and deep-learning-based robustness prediction [39] offers a sophisticated pathway to transcend traditional algorithmic limitations through automated, content-aware parameter tuning. To mitigate the identified

computational latencies, subsequent work must prioritize GPU-accelerated DCT implementations [32] and specialized hardware architectures, such as FPGA or ASIC, for high-performance embedded systems [41]. Such optimizations are essential for advancing steganographic viability within resource-constrained environments and real-time streaming applications where temporal consistency is paramount.

5. CONCLUSION

This investigation established a systematic performance framework for Python-based LSB steganography through rigorous dual-metric assessment and robustness benchmarking. By delineating the operational boundaries of each steganographic paradigm, this study provides an empirical foundation for implementation selection in practical, security-critical applications.

This study demonstrates that no single Python LSB implementation is universally superior. Instead, selection should be guided by specific application requirements and threat models. Classical methods excel in capacity and computational efficiency, while frequency-domain approaches provide superior robustness at the cost of complexity and capacity. Randomized methods offer enhanced security with minimal quality penalty. The empirical data and decision framework provided in this study enable informed implementation choices for secure data hiding applications. By quantifying the specific trade-offs between capacity, imperceptibility, robustness, and computational efficiency, this work bridges the gap between theoretical steganography research and practical Python implementation.

The findings confirm theoretical predictions about frequency domain robustness while providing concrete performance metrics for practical decision-making. The 50% BER reduction achieved by DCT-based methods under rotation validates the theoretical advantages of frequency domain embedding, while the comparable performance under noise and cropping attacks highlights the attack specific nature of these advantages.

This comprehensive evaluation establishes a foundation for future research in adaptive steganography systems that can dynamically select optimal embedding strategies based on application requirements and environmental constraints. The empirical framework and benchmarking protocols developed in this study provide reproducible methodology for continued advancement in the field of image steganography.

REFERENCES

[1] J. A. Mazumder and K. Hemachandran, "Study of Image Steganography using LSB, DFT and DWT," *Int J Comput Technol*, vol. 11, pp. 2618–2627, 2013.

[2] S. K. Behera and M. Mishra, "Steganography--A Game of Hide and Seek in Information Communication," *arXiv preprint arXiv:1604.00493*, 2016.

[3] S. Rahman *et al.*, "A novel and efficient digital image steganography technique using least significant bit substitution," *Sci. Rep.*, vol. 15, no. 1, p. 107, 2025.

[4] A. K. Sahu and M. Sahu, "Digital image steganography and steganalysis: A journey of the past three decades," *Open Computer Science*, vol. 10, no. 1, pp. 296–342, 2020.

[5] A. Lavanya, S. Sindhuja, L. Gaurav, and W. Ali, "A Comprehensive Review of Data Visualization Tools: Features," *Strengths, and Weaknesses*, 2023.

[6] D. R. I. M. Setiadi, S. Rustad, P. N. Andono, and G. F. Shidik, "Digital Image Steganography Survey and Investigation (Goal, Assessment, Method, Development, and Dataset)," *Signal Processing*, vol. 206, p. 108908, 2023, doi: 10.1016/j.sigpro.2022.108908.

[7] R. R. Asaad, R. I. Ali, Z. A. Ali, and A. A. Shaaban, "Image processing with Python libraries," *Academic Journal of Nawroz University (AJNU)*, vol. 12, no. 2, 2023.

[8] D. Raghuvanshi, K. Joshi, R. Nandal, S. Singh, and D. Kumari, "Advancing image steganography: PRISMA-ScR based analysis of spatial domain techniques," *Multimed. Tools Appl.*, vol. 84, no. 40, pp. 48475–48509, 2025, doi: 10.1007/s11042-025-21040-5.

[9] G. Li, S. Li, Z. Qian, and X. Zhang, *Cover-separable Fixed Neural Network Steganography via Deep Generative Models*, vol. 1, no. 1. Association for Computing Machinery, 2024. doi: 10.1145/3664647.3680824.

[10] R.-G. Zhou, J. Luo, X. Liu, C. Zhu, L. Wei, and X. Zhang, "A Novel Quantum Image Steganography Scheme Based on LSB," *International Journal of Theoretical Physics*, vol. 57, no. 6, pp. 1848–1863, 2018, doi: 10.1007/s10773-018-3710-x.

[11] Y. JinaChanu, Kh. Manglem Singh, and T. Tuithung, "Image Steganography and Steganalysis: A Survey," *Int. J. Comput. Appl.*, vol. 52, no. 2, pp. 1–11, 2012, doi: 10.5120/8171-1484.

[12] X. Zhang, F. Peng, and M. Long, "Robust Coverless Image Steganography Based on DCT and LDA Topic Classification," *IEEE Trans. Multimedia*, vol. 20, no. 12, pp. 3223–3238, 2018, doi: 10.1109/TMM.2018.2838334.

[13] K. Zeng, K. Chen, W. Zhang, Y. Wang, and N. Yu, "Improving robust adaptive steganography via minimizing channel errors," *Signal Processing*, vol. 195, p. 108498, 2022, doi: 10.1016/j.sigpro.2022.108498.

[14] A. Munshi, "Randomly-based Stepwise Multi-Level Distributed Medical Image Steganography," *Engineering, Technology and Applied Science Research*, vol. 13, no. 3, pp. 10922–10930, 2023, doi: 10.48084/etasr.5935.

[15] D. R. I. M. Setiadi, "PSNR vs SSIM: imperceptibility quality assessment for image steganography," *Multimed. Tools Appl.*, vol. 80, no. 6, pp. 8423–8444, 2021, doi: 10.1007/s11042-020-10035-z.

[16] Y. M. Younis, R. J. Mstafa, and S. AL-Dohuki, "AttenHideNet: A novel deep learning-based image steganography method using a lightweight U-net with soft attention," *Appl. Soft Comput.*, vol. 182, no. February, p. 113583, 2025, doi: 10.1016/j.asoc.2025.113583.

[17] D. A. Shehab and M. J. Alhaddad, "SS symmetry Comprehensive Survey of Multimedia Steganalysis :," *Symmetry 2022, MDPI*, vol. 14, no. 117, pp. 1–26, 2022.

[18] S. Arunkumar, V. Subramaniyaswamy, V. Vijayakumar, N. Chilamkurti, and R. Logesh, "SVD-based robust image steganographic scheme using RIWT and DCT for secure transmission of medical images," *Measurement (Lond).*, vol. 139, pp. 426–437, 2019, doi: 10.1016/j.measurement.2019.02.069.

[19] V. Himthani, V. S. Dhaka, M. Kaur, G. Rani, M. Oza, and H. N. Lee, "Comparative performance assessment of deep learning based image steganography techniques," *Sci. Rep.*, vol. 12, no. 1, pp. 1–16, 2022, doi: 10.1038/s41598-022-17362-1.

[20] F. Li, L. Li, Y. Zeng, J. Yu, and C. Qin, "Adversarial multi-image steganography via texture evaluation and multi-scale image enhancement," *Multimed. Tools Appl.*, vol. 84, no. 9, pp. 5793–5823, 2025, doi: 10.1007/s11042-024-18920-7.

[21] P. Fan, H. Zhang, and X. Zhao, "Robust video steganography for social media sharing based on principal component analysis," *EURASIP J. Inf. Secur.*, vol. 2022, no. 1, 2022, doi: 10.1186/s13635-022-00130-z.

[22] R. Apau, M. Asante, F. Twum, J. Ben Hayfron-Acquah, and K. O. Peasah, *Image steganography techniques for resisting statistical steganalysis attacks: A systematic literature review*, vol. 19, no. 9 September. 2024. doi: 10.1371/journal.pone.0308807.

[23] Y. Sanjalawe, S. Al-E'mari, S. Fraihat, M. Abualhaj, and E. Alzubi, "A deep learning-driven multi-layered steganographic approach for enhanced data security," *Sci. Rep.*, vol. 15, no. 1, pp. 1–30, 2025, doi: 10.1038/s41598-025-89189-5.

[24] M. Taleby Ahvanooey, Q. Li, J. Hou, H. Dana Mazraeh, and J. Zhang, "AITSteg: An innovative text steganography technique for hidden transmission of text message via social media," *IEEE Access*, vol. 6, pp. 65981–65995, 2018, doi: 10.1109/ACCESS.2018.2866063.

[25] A. Mohammadi, "A general framework for reversible data hiding in encrypted images by reserving room before encryption," *J. Vis. Commun. Image Represent.*, vol. 85, no. December 2021, p. 103478, 2022, doi: 10.1016/j.jvcir.2022.103478.

[26] R. S. Hameed, S. S. Mokri, M. S. Taha, and M. M. Taher, "High Capacity Image Steganography System based on Multi-layer Security and LSB Exchanging Method," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 8, pp. 108–115, 2022, doi: 10.14569/IJACSA.2022.0130814.

[27] G. Gao, L. Zhang, Y. Lin, S. Tong, and C. Yuan, "High-performance reversible data hiding in encrypted images with adaptive Huffman code," *Digital Signal Processing: A Review Journal*, vol. 133, p. 103870, 2023, doi: 10.1016/j.dsp.2022.103870.

[28] S. Ahmad, J. O. Ogala, F. Ikpotokin, M. Arif, J. Ahmad, and S. Mehfuz, "Enhanced CNN-DCT Steganography: Deep Learning-Based Image Steganography Over Cloud," *SN Comput. Sci.*, vol. 5, no. 4, 2024, doi: 10.1007/s42979-024-02756-x.

[29] Y. Peng, C. Fu, Y. Zheng, Y. Tian, G. Cao, and J. Chen, "Medical steganography: Enhanced security and image quality, and new S-Q assessment," *Signal Processing*, vol. 223, no. May, p. 109546, 2024, doi: 10.1016/j.sigpro.2024.109546.

[30] W. Rehman, "A Novel Approach to Image Steganography Using Generative Adversarial Networks," pp. 1–17, 2024, [Online]. Available: http://arxiv.org/abs/2412.00094

[31] N. N. Kumar, R. Viswanathan, and P. S. Kumar, "An Efficient Approach on Image Encryption Steganography based on 2D SWT with Chaotic Techniques," in *2024 4th International Conference on Soft Computing for Security Applications (ICSCSA)*, 2024, pp. 479–486. doi: 10.1109/ICSCSA64454.2024.00083.

[32] S. Agha, F. Jan, H. A. Khan, M. Kaleem, and M. Khan, *Efficient motion estimation and discrete cosine transform implementation using the graphics processing units*, vol. 19, no. 8. 2024. doi: 10.1371/journal.pone.0307217.

[33] L. Widyawati, I. Riadi, and Y. Prayudi, "Comparative Analysis of Image Steganography using SLT, DCT and SLT-DCT Algorithm," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 20, no. 1, pp. 169–182, 2020, doi: 10.30812/matrik.v20i1.701.

[34] W. Tang, B. Li, M. Barni, J. Li, and J. Huang, "Improving Cost Learning for JPEG Steganography by Exploiting JPEG Domain Knowledge," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 4081–4095, 2022, doi: 10.1109/TCSVT.2021.3115600.

[35] B. A. Y. Alqaralleh, T. Vaiyapuri, V. S. Parvathy, D. Gupta, A. Khanna, and K. Shankar, "Blockchain-assisted secure image transmission and diagnosis model on Internet of Medical Things Environment," *Pers. Ubiquitous Comput.*, vol. 28, no. 1, pp. 17–27, 2024, doi: 10.1007/s00779-021-01543-2.

[36] M. Xu and Y. Lin, "FedSteg: Coverless Steganography-Based Privacy-Preserving Decentralized Federated Learning," *Ieej Transactions on Electrical and Electronic Engineering*, vol. 19, no. 8, pp. 1345–1359, 2024, doi: 10.1002/tee.24085.

[37] E. Kuchumova, S. M. M. Monterrubio, and J. A. Recio-García, "STEG-XAI: explainable steganalysis in images using neural networks.," *Multim. Tools Appl.*, vol. 83, no. 17, pp. 50601–50618, 2024, doi: 10.1007/S11042-023-17483-3.

[38] G. Han, D. J. Lee, J. Hur, J. Choi, and J. Kim, "Deep Cross-Modal Steganography Using Neural Representations," *Proceedings - International Conference on Image Processing, ICIP*, pp. 1205–1209, 2023, doi: 10.1109/ICIP49359.2023.10222113.

[39] N. Farooq and A. K. Selwal, "Image steganalysis using deep learning: a systematic review and open research challenges," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, pp. 7761–7793, 2023, [Online]. Available: https://api.semanticscholar.org/CorpusID:257883581

[40] J. Ye, J. Ni, and Y. Yi, "Deep Learning Hierarchical Representations for Image Steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2545–2557, 2017, doi: 10.1109/TIFS.2017.2710946.

[41] X. Mo, S. Tan, B. Li, and J. Huang, "MCTSteg: A Monte Carlo Tree Search-Based Reinforcement Learning Framework for Universal Non-Additive Steganography," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4306–4320, 2021, doi: 10.1109/TIFS.2021.3104140.