Intelligent Automation & Soft Computing

Tech Science Press

https://doi.org/10.32604/iasc.2026.087652

ARTICLE

A Dual-Layer Framework for Secure Image Sharing: Integrating Adversarial Perturbation and Reversible DCT-QIM Steganography with a Secret Embedding Key

Hartawan Bahari Mulyadi and Tohari Ahmad*

Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

*Corresponding Author: Tohari Ahmad. Email: tohari@its.ac.id

Received: 20 June 2026; Accepted: 28 September 2026; Published: 02 November 2026

ABSTRACT: Publicly distributing digital images while preserving privacy and copyright ownership has become an ongoing challenge, yet current image-sharing methods face two unresolved problems: embedded metadata is vulnerable to deletion or tampering by automated systems, and existing information-hiding techniques permanently alter the original image, preventing its recovery. This paper proposes a unified dual-layer framework that integrates generative adversarial network (GAN) based momentum iterative method projected gradient descent (MIM-PGD) adversarial perturbation with discrete cosine transform quantization index modulation (DCT-QIM) steganographic embedding governed by a secret embedding key (SEK). The adversarial perturbation layer generates imperceptible noise that degrades unauthorized AI classifier confidence, while the steganographic layer embeds a hidden message, authentication data, or recovery information. The perturbation affected DCT coefficient regions are reused as deterministic embedding locations for metadata insertion, minimizing additional distortion. The SEK coordinates both layers, enabling authorized users to extract embedded information and restore the image to its original state with near-lossless fidelity. Evaluation on ImageNet, CIFAR-100, and USC-SIPI datasets using peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), defense success rate (DSR), and reversibility accuracy (RA) demonstrate two top configurations at BPP = 0.10; at $\epsilon = 16/255$, the method achieves PSNR of 44.28 dB, SSIM of 0.972, mean DSR of 86.1% across five classifiers, and RA above 0.998; at $\epsilon = 8/255$, the method achieves PSNR above 35 dB, SSIM above 0.90, mean DSR of 80.0%, and RA above 0.998. The framework simultaneously provides adversarial resistance against unauthorized AI-driven analysis, covert data embedding, near-lossless reversibility (RA $\geq$ 0.998) to the original cover image, and visual imperceptibility, applicable to privacy-preserving image sharing, digital rights protection, authentication, and secure cloud-based storage within a single pipeline.

KEYWORDS: Adversarial machine learning; information security; secure image sharing; image steganography; digital rights protection; image authentication; ICT infrastructure

1 Introduction

Secure and privacy-preserving image sharing has become a growing concern in modern cloud-based multimedia storage and social networking, where users publicly distribute images while attempting to protect personal privacy and digital rights [1,2]. The scale of contemporary image data generation is large, with billions of images uploaded and shared each day across personal, medical, and commercial repositories [3]. In parallel with this growth, deep neural networks (DNNs) have achieved levels of visual understanding that enable automated, unauthorized profiling, surveillance, and classification of public images. Residual

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

networks [4], Vision Transformers [5], and their variants routinely achieve top-1 accuracy above 80% on ImageNet [6] and comparable results on CIFAR-100 [7]. Generative Adversarial Networks (GANs) [8] and diffusion-based synthesis models have further expanded the threat landscape: GAN fingerprint based data poisoning can inject targeted backdoors into DNN classifiers [9], while adversarial patch techniques merged with federated learning produce natural looking patches that achieve near-perfect attack success rates [10]. Image synthesis tools such as Stable Diffusion enable high fidelity manipulation that can bypass conventional content authentication [11,12]. Recent studies confirm that automated AI classifiers remain highly sensitive to adversarial perturbations [13,14], offering a potential pathway to protect public images from unauthorized AI-driven analysis [15,16].

Existing image protection mechanisms each address only part of the problem. Access control and encryption prevent unauthorized file retrieval, yet they offer no protection once pixel content becomes accessible to an AI model [1,3]. Adversarial perturbation methods, including FGSM [17], PGD [18], universal adversarial perturbations [19], and the C&W attack [20], can cause DNN classifiers to misclassify images, but the perturbations are strictly irreversible and carry no embeddable payload [19,21]. User-facing adversarial tools such as Fawkes [22], Glaze [11], and Nightshade [23] improve visual naturalism, but they too cannot embed retrievable secret information or reconstruct the original image [24]. On the information hiding side, classical transform domain watermarking using the discrete cosine transform (DCT) [25] provides moderate robustness, as surveyed by Boenisch [26], yet capacity is limited and deep learning based removal attacks are gaining effectiveness [27]. End-to-end deep watermarking methods such as HiDDeN [28] increase robustness through joint training but remain irreversible [24]. Least significant bit (LSB) steganography [29] offers high embedding capacity at the cost of trivial detectability by steganalysis [30,31] and fragility under compression. While deep learning based steganography [32] improves visual imperceptibility and security, it does not support reversibility. RDH [33] enables lossless image recovery, which is essential for applications where the original content must be restored exactly, such as secured medical data transfer and digital rights verification [34], but RDH methods are not designed to resist AI classification [35]. Recent dual strategy steganography frameworks [36] and robust steganographic networks [37] have started combining hiding with adversarial resilience, but none supports full cover image reversibility. No single paradigm in the reviewed literature simultaneously addresses AI resistance, visual imperceptibility, covert payload embedding, and original-cover reversibility.

A structured review of the literature reveals the specific and complementary nature of these shortcomings, as summarized in Table 1. Gradient based attacks such as FGSM [17] and PGD [18] achieve high misclassification rates in white box settings but exhibit limited transferability across architectures [19,21]. GAN-based generators like AdvGAN [38] and related frameworks [24] yield more natural perturbations, but they remain irreversible and lack steganographic integration [39]. Classical frequency domain watermarking, covered comprehensively by Boenisch [26], survives mild JPEG compression but offers insufficient capacity and is vulnerable to modern removal networks [27]. Adversarial watermarking for model ownership [40,41] targets a different problem entirely and does not guarantee reversibility. Deep watermarking [28] delivers high robustness at the expense of irreversibility. Coverless image steganography methods based on text-to-image generation [12] and generative steganography using diffusion models [42] represent emerging alternatives, but their reliance on generative model reproducibility limits applicability to preexisting photographs. Adversarial steganography frameworks for generated images [43] share this limitation. LSB methods [29] achieve the highest raw capacity but collapse under both steganalysis [31,44] and mild lossy compression. Prediction error expansion [45] and histogram shifting [46] guarantee perfect reconstruction but have never been evaluated under adversarial perturbation conditions [33,47]. Deep learning-augmented RDH [48] improves adaptive capacity but remains defenseless against AI classifiers. Recent proactive image

protection methods such as Dual Defense [49] and EditGuard [50] combine adversarial perturbation with watermarking for copyright traceability but do not provide reversible image recovery, which is required when the user needs to restore the image to its original clean state [2]. Based on this review, no method in the surveyed literature unifies adversarial protection, reversible frequency domain steganographic embedding, and key-controlled deterministic extraction within a single pipeline.

In practice, secure image sharing and privacy preservation require the simultaneous satisfaction of four properties: (1) *AI resistance*, whereby unauthorized classifiers cannot extract meaningful labels from marked images; (2) *data embedding*, whereby owner credentials, authentication tags, or integrity checksums are conveyed within the image; (3) *reversibility*, whereby an authorized receiver possessing the protected image and the SEK can restore the image to its original cover state with exact or near-lossless fidelity (RA ≥ 0.998); and (4) *visual imperceptibility*, while the marked image appears unaltered to human observers. As shown in Table 1, none of the surveyed methods satisfies all four properties. This gap motivates the present work.

This paper addresses that gap by proposing a unified dual layer framework that integrates GAN-based MIM-PGD adversarial perturbation with DCT-QIM steganographic embedding, governed by a Secret Embedding Key (SEK). The framework operates in three coordinated phases. In the Protection Phase, a GAN generator combined with MIM and PGD [8,18] produces adversarial perturbations bounded by $\|\boldsymbol{\delta}\|_\infty \leq \epsilon$ that cause diverse classifiers, including ResNet [4], InceptionV3, VGG-16, Vision Transformer [5], and the commercial Google Cloud Vision API, to misclassify the input. The momentum accumulation [51] promotes cross architecture transferability [19]. In the Embedding Phase, the system identifies perturbation-affected regions by comparing DCT coefficient parities between the cover and adversarial images in the YCbCr luminance channel [52], then embeds secret metadata into those regions via QIM [53]. Because embedding targets regions already modified by the adversarial attack, the additional visual distortion is minimal [54]. In the Recovery Phase, the SEK enables authorized users to extract the embedded data and reconstruct the original cover image with near-lossless fidelity (RA ≥ 0.998), verified by RA [48]. DSR [19], PSNR, and SSIM [54,55] serve as the primary evaluation criteria. The contributions are:

1. A unified pipeline integrating GAN-based adversarial poisoning with reversible DCT-QIM steganographic embedding, combining capabilities that prior work treats separately: AdvGAN [38] for perturbation and RDH [48,56] for recovery. The combined system achieves mean DSR of 86.1% at $\epsilon = 16/255$ (BPP = 0.10) with PSNR 44.28 dB on ImageNet-100, and mean DSR of 80.0% at $\epsilon = 8/255$ (BPP = 0.10) with PSNR above 35 dB.
2. A UNet-guided perturbation strategy restricting adversarial noise to semantically relevant foreground regions, concentrating perturbation energy on areas that affect classifier decisions.
3. A perturbation-guided embedding strategy that reuses already-altered DCT coefficient positions as steganographic targets. QIM modifications blend into existing perturbation distortion rather than introducing new artifacts, reducing additional quality loss compared to methods [29] that embed without reference to prior modifications.
4. An SEK mechanism extending Shi et al. [33] with cryptographic block permutation, SHA-256 verification, and DCT coefficient backups, validated on USC-SIPI [57].

The remainder of this paper surveys related work (Section 2), presents the proposed method (Section 3), describes experiments (Section 4), reports and discusses results (Section 5), and concludes with future directions (Section 6).

2 Related Works

Research on digital image protection spans adversarial perturbation, watermarking, steganography, and RDH. Each approach tries to either prevent unauthorized AI analysis or hide information inside an image while keeping visual quality acceptable [16,26,33]. Studies of deep-learning-based image profiling and classification [14,15] confirm that online platforms increasingly rely on AI classifiers for automated image analysis, making adversarial vulnerability a direct concern for privacy-preserving image sharing. Section 2.1 first provides a conceptual background of the three core mechanisms combined in this work. The literature review in Section 2.2 then examines each area, and Section 2.3 identifies the research gap that motivates the proposed framework.

2.1 Background

Three independent research areas form the conceptual foundation of the proposed framework: adversarial image perturbation, frequency domain steganography, and reversible data hiding. Understanding each area in isolation is necessary before examining how the proposed method integrates them.

Adversarial perturbation modifies a digital image by adding pixel level noise that is imperceptible to humans but causes a DNN classifier to produce an incorrect label [17,18]. The perturbation is designed so that the protected image stays within a bounded distortion range, commonly expressed as $\|\boldsymbol{\delta}\|_\infty \leq \epsilon$, while the classifier's confidence in the true label collapses. Frequency domain steganography hides a secret binary message by modifying transform coefficients of an image, typically in the DCT domain [25,53]. The key advantage over spatial-domain methods such as LSB substitution is that DCT-based hiding is less detectable by steganalysis and allows a formal trade-off between payload capacity and image quality. RDH extends this by ensuring that the embedding step is completely reversible: given a key and the stego image, the receiver can extract the hidden message and then restore the image to its exact prior state with no loss of pixel values [33,56].

Fig. 1 shows how these three mechanisms combine in the proposed framework at a conceptual level. The left side of the diagram represents the protection path: a cover image first receives an adversarial perturbation to become a protected image, and a secret message is then embedded into perturbation-affected frequency coefficients to produce the final protected stego image. The right side represents the authorized recovery path: the SEK enables the receiver to extract the hidden message from the stego image and independently restore the image to its pre-embedding state. The two paths are symmetric in structure, and the SEK is the only information that bridges them. This architecture means that anyone without the SEK sees only a visually normal image and cannot tell that a hidden message is present.

Fig. 2 expands on this concept by showing the concrete pipeline stages. Starting from the cover image, a UNet segmentation model separates the foreground from the background so that adversarial noise is applied only to semantically relevant regions. The GAN generator, guided by MIM-PGD, then produces a bounded perturbation $\boldsymbol{\delta}$ that pushes the image across the classifier's decision boundary. After the PI is formed, candidate embedding positions are detected by comparing DCT coefficient parities between the cover and the PI in the luminance channel, and the SEK is assembled from those positions, their backed up coefficient values, and the cryptographic seed. The QIM embedding step then writes the secret message bits into those positions to produce the PSI. On the authorized recovery side, the SEK drives message extraction and lossless image restoration.



Figure 1: General workflow of the proposed dual layer framework: adversarial perturbation and hidden-message embedding protect images from AI analysis while enabling SEK-based message extraction and reversible recovery.

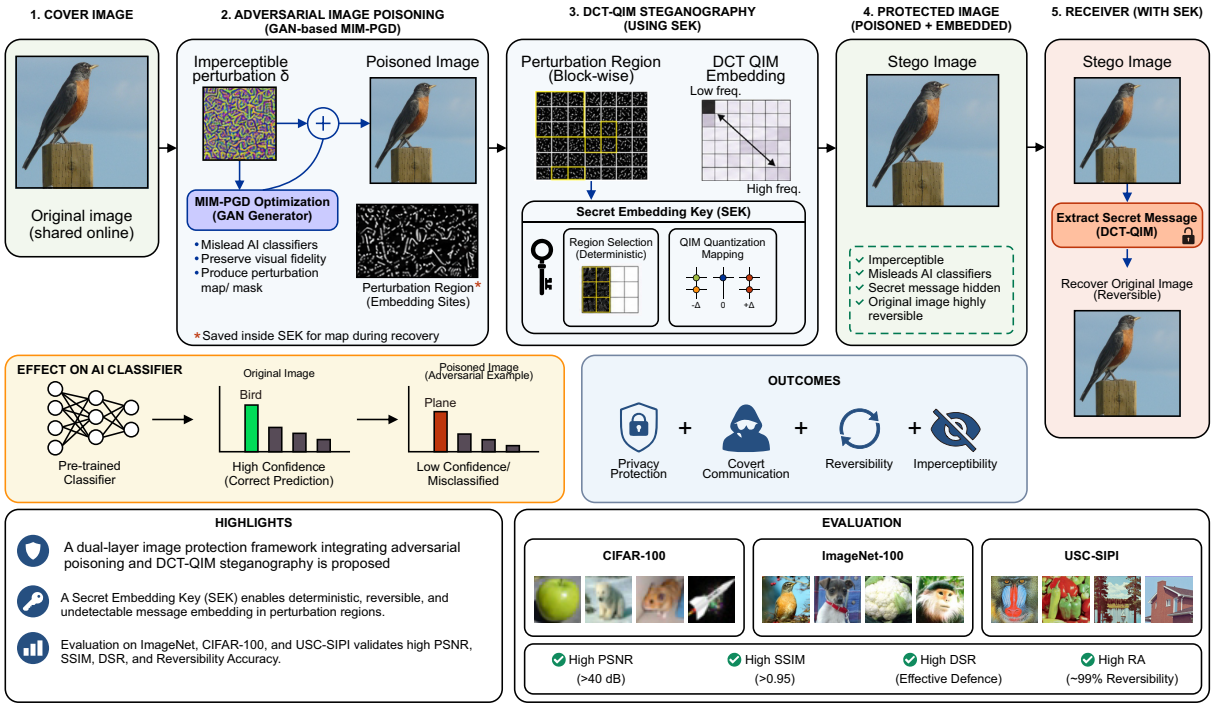


Figure 2: Overview of the proposed dual layer image protection framework. The pipeline applies UNet segmentation, GAN-MIM-PGD adversarial perturbation, and DCT-QIM steganographic embedding, all governed by the SEK.

Fig. 3 details the embedding path. The cover image is first converted from RGB to the YCbCr color space so that all modifications are confined to the luminance channel, where the adversarial perturbation has already altered DCT coefficient parities. The SEK identifies which 8 × 8 DCT blocks have perturbation-affected coefficients, shuffles their order using a cryptographic random seed, and stores the original coefficient values as backups. Each bit of the secret message is then embedded by adjusting the quantized parity of the corresponding coefficient according to the QIM rule. After all bits are placed, an inverse DCT rebuilds the spatial-domain blocks and the image is converted back to RGB. The result is a protected stego image that simultaneously misleads classifiers and carries the hidden message.

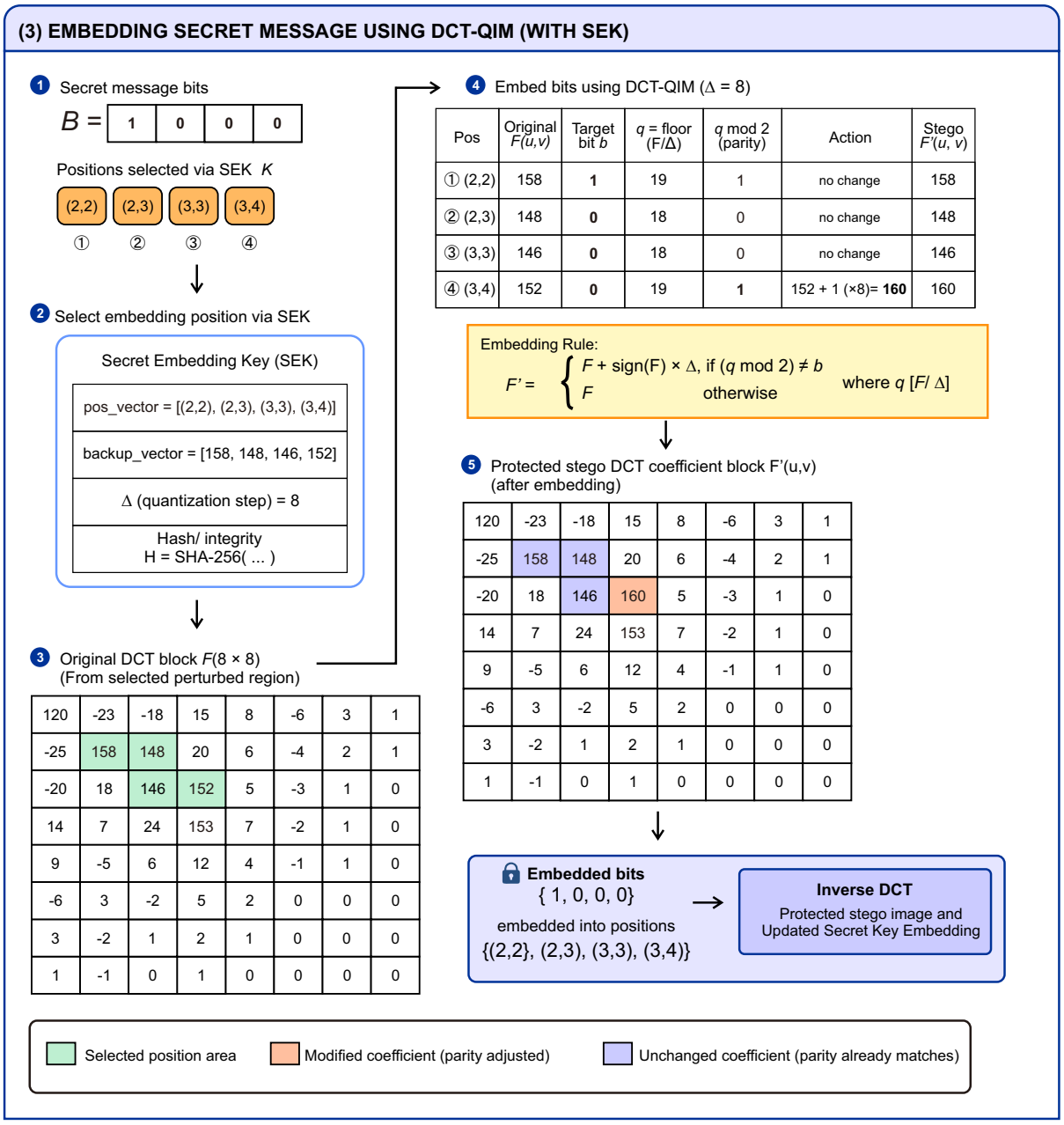


Figure 3: DCT-QIM embedding process: secret bits are embedded into perturbation-affected DCT coefficients selected through the SEK.

Fig. 4 details the recovery path. Given the protected stego image and the SEK, the receiver computes the DCT of each flagged block and reads the quantized parity of each coefficient to recover the embedded bits. Majority voting over the replicated header and payload bits corrects any minor coefficient distortions. Once the message has been extracted and its CRC32 checksum verified, the receiver replaces each stego DCT coefficient with the original value stored in the SEK's backup field and applies an inverse DCT to reconstruct the luminance channel. Converting back to RGB yields an image that is pixel accurate relative to the pre-embedding state. The separation between the extraction step (reading parities) and the restoration step (replacing backed up coefficients) is what makes lossless recovery possible without needing to reverse the adversarial perturbation.

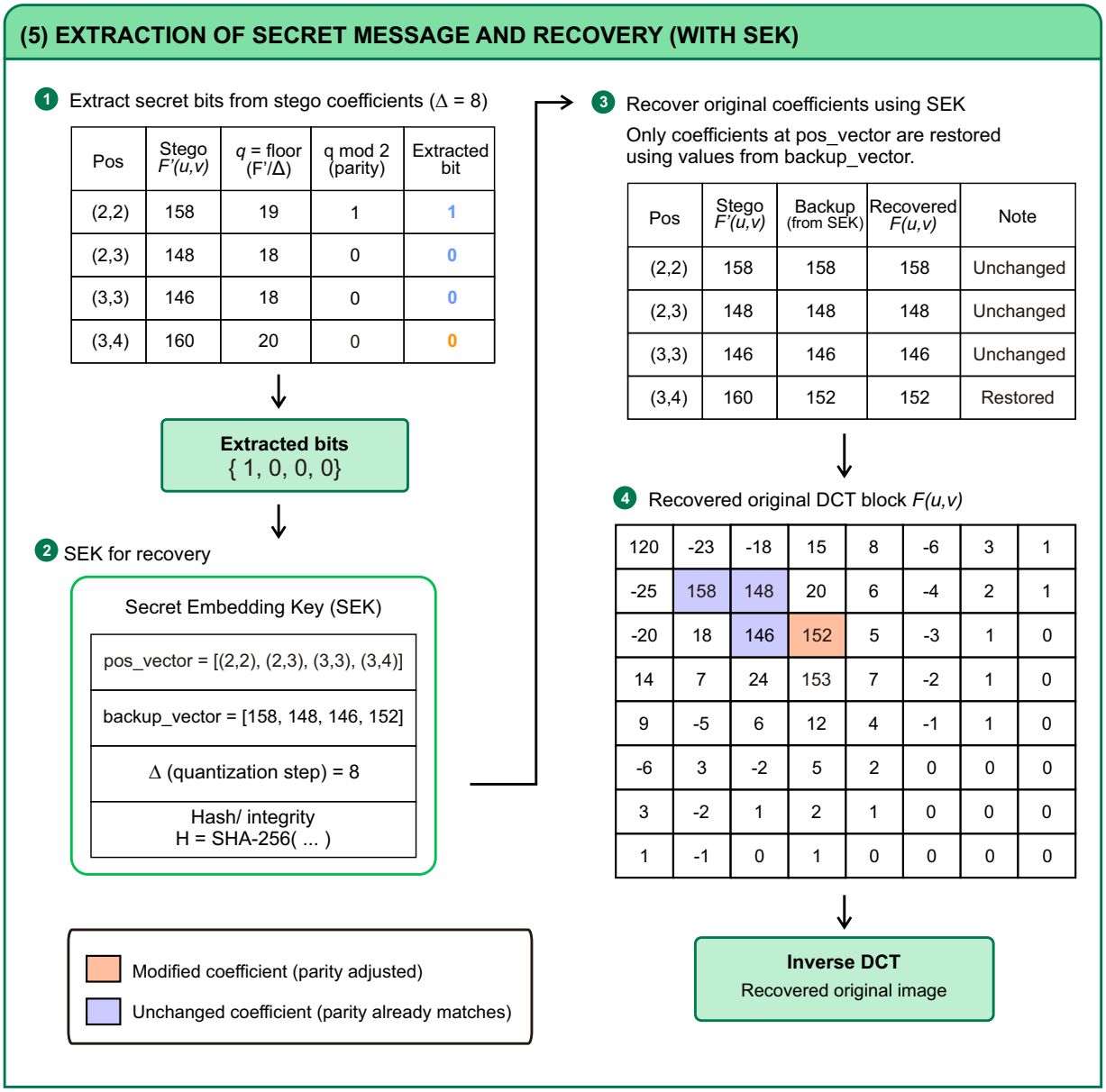


Figure 4: Extraction and recovery process: hidden bits are read from stego DCT coefficients and original coefficients are restored for reversible reconstruction.

2.2 Literature Review

2.2.1 Adversarial Protection Methods

The idea behind adversarial perturbation is straightforward: add small, carefully computed changes to an image so that a DNN misclassifies it, while humans still see the original content. Early methods proved this was practical. FGSM [17] computes a single gradient step. PGD [18] iterates multiple steps within an ℓ_∞ budget. DeepFool [58] finds the minimum perturbation needed to cross a decision boundary. The C&W attack [20] optimizes for both misclassification and low distortion.

Later work moved toward more realistic and transferable perturbations. AdvGAN [38] trains a generative network to produce adversarial images that look natural. The Momentum Iterative Attack (MI-FGSM) [51] accumulates gradient momentum across iterations, which makes perturbations transfer better to models the attacker has never seen. Universal adversarial perturbations [19] go further by computing a single noise pattern that fools a classifier on most inputs. On the application side, Fawkes [22], Glaze [11], and Nightshade [23] apply adversarial techniques specifically to protect users from unauthorized facial recognition and artistic style theft. Recent surveys [19,59] confirm these methods work well against a range of classifiers. Forensic AI detectors are also susceptible: De Rosa et al. [13] demonstrated that CLIP-based forensic detectors can be deceived by targeted adversarial perturbations at the IEEE Workshop on Information Forensics and Security. The common weakness, however, is that all of these methods are irreversible. Once the perturbation is applied, there is no way to get the original image back, and none of them can carry a hidden message [16,21].

2.2.2 Information Hiding Methods

Watermarking and steganography both embed information into images, but for different purposes. Watermarks typically prove ownership or track distribution. Steganography hides a secret message so that nobody knows it is there. Classical transform domain watermarking works in the frequency representation of an image, usually via the DCT [25] or DWT, and can survive common processing like JPEG compression [26]. The trade-off is limited capacity, and deep learning based removal tools are getting better at removing these watermarks out [27]. Deep learning watermarking methods like HiDDeN [28] improve both imperceptibility and robustness through end-to-end training. Similarly, deep neural network architectures have been applied to optimize steganographic security and embedding capacity [32]. Methods that watermark DNN models for ownership verification [26,40] solve a different problem entirely and do not protect image content.

In the spatial domain, LSB substitution [29] can embed a lot of data with minimal computation. The downside is well documented: modern steganalysis [30,31] catches LSB patterns easily, and even mild JPEG compression destroys the hidden bits. Adaptive schemes that combine steganography with adversarial perturbation [39,60] have been proposed, but they remain irreversible. In contrast, QIM [53] takes a different approach by quantizing transform coefficients to encode bits deterministically. When used with the DCT, QIM offers a clean capacity-distortion trade-off and reliable extraction [54], making it a strong foundation for steganographic systems that need formal performance guarantees.

The common limitation across all information hiding methods is that they provide *no defense against AI-based image classification*. A classifier processes a watermarked or stego image with the same accuracy as the unprotected original. Recent proactive image protection methods have begun addressing this limitation: Dual Defense [49] combines adversarial perturbation with robust watermarking for face swapping protection, and EditGuard [50] uses image-bit steganography for tamper localization and copyright verification. However, neither provides reversible image recovery, which is required when the original image must be retrieved in its unaltered form [2].

2.2.3 RDH

RDH solves a problem that watermarking and steganography typically does not satisfy: how to perfectly reconstruct the original cover image after the hidden data has been extracted. The foundational techniques are Difference Expansion [56], Prediction-Error Expansion [45], and Histogram Shifting [46]. All three allow lossless recovery of both the payload and the original pixels. RDH has found real use in privacy-preserving sharing, digital rights management, and medical imaging, where any alteration to the original content is unacceptable [2,33]. For instance, reversible data hiding systems have been developed to ensure secured medical data transfer through steganography [34]. Recent extensions cover multi image RDH [61], deep learning-assisted methods [48], and robust reversible watermarking that survives geometric attacks [35,47,62]. None of these, however, were designed to resist adversarial perturbation or defend images against AI classification [33,47].

2.3 Research Gap

The analysis above reveals that existing methods are fragmented: adversarial perturbation resists AI analysis but is irreversible and payload-free; steganography embeds data but offers no classifier defense; and RDH recovers images losslessly but provides no adversarial resistance. Privacy-preserving image sharing requires all four properties simultaneously (AI resistance, data embedding, reversibility, and imperceptibility), because applying separate tools sequentially would compound distortion and risk one process destroying the other. Table 1 summarizes the capability coverage of existing methods. Recent proactive image protection methods [49,50] have advanced toward combining adversarial defense with watermarking but omit reversibility. No existing method simultaneously satisfies all four requirements as defined here: adversarial defense against AI classifiers, user-controlled secret payload embedding, recovery of the original cover image, and imperceptible distortion. Yin et al. [63] combine adversarial perturbation with data embedding, but their recovery mechanism does not restore the original cover image. Lyu et al. [64] provide reversible adversarial perturbation but do not support user-controlled steganographic payload embedding. This motivates the proposed framework. A detailed quantitative comparison with prior methods is provided in Section 5.6.

3 Proposed Method

To satisfy the requirements identified in Section 2.3, the proposed framework integrates two complementary protection mechanisms into a single reversible pipeline: GAN-based MIM-PGD adversarial image poisoning and DCT-QIM steganographic embedding, coordinated through an SEK. Prior adversarial methods such as AdvGAN [38] and Fawkes [22] are strictly irreversible and carry no hidden payload; prior RDH approaches [48,56] provide reconstruction but no adversarial defense. This framework extends the generation-based perturbation paradigm with near lossless recoverability to the post-perturbation state and data embedding.

The system operates in three coordinated phases, illustrated in Fig. 2: (1) the Protection Phase, which generates an imperceptible adversarial perturbation that causes target classifiers to misclassify the image; (2) the Embedding Phase, which hides a secret binary message inside the perturbation-affected frequency coefficients; and (3) the Recovery Phase, which allows authorized parties to extract the message and restore the image to the post-perturbation state with near lossless fidelity (RA $\geq$ 0.998).

Table 1: Capability coverage of related image protection methods. ✓ = fully supported; × = not supported; Partial = limited or dependent on conditions.

Reference	AI	Emb.	Rev.	Imp.
Goodfellow et al. [17]	✓	×	×	Partial
Madry et al. [18]	✓	×	×	Partial
Carlini & Wagner [20]	✓	×	×	Partial
Xiao et al. [38]	✓	×	×	✓
Shan et al. [22]	✓	×	×	✓
Boenisch [26]	×	✓	×	✓
Zhu et al. [28]	×	✓	×	✓
Chan & Cheng [29]	×	✓	×	Partial
Chen & Wornell [53]	×	✓	×	✓
Tian [56]	×	✓	✓	✓
Ni et al. [46]	×	✓	✓	✓
Tang et al. [47]	×	✓	✓	✓
Gao et al. [35]	×	✓	✓	✓
Panchikkil and Manikandan [65]	×	✓	✓	✓
Aljughaiman and Alrawashdeh [66]	×	✓	✓	✓
Liu et al. [67]	×	✓	✓	✓
Li et al. [24]	✓	✓	×	✓
Zhang et al. [49]	✓	✓	×	✓
Zhang et al. [50]	✓	✓	×	✓
Yin et al. [63]	✓	✓	×	✓
Lyu et al. [64]	✓	×	✓	✓

AI: Adversarial defense; Emb.: User-controlled secret payload embedding; Rev.: Original-cover recovery (exact or near-lossless); Imp.: Imperceptibility. Partial = limited or condition-dependent.

Let $\mathbf{x} \in \mathbb{R}^{H\times W\times C}$ denote a 24-bit RGB cover image with height H, width W, and $C = 3$ color channels. Let $\mathbf{m} \in \{0,1\}^L$ denote a binary secret message of length L bits. The system produces a Protected Stego Image (PSI), denoted $\mathbf{x}^*$, where $\hat{\mathbf{x}}$ is the intermediate Protected Image (PI), $\boldsymbol{\delta}$ is the adversarial perturbation vector constrained by $\|\boldsymbol{\delta}\|_\infty \le \epsilon$, $\mathcal{E}(\cdot)$ is the DCT-QIM embedding operator, and k is the SEK. The overall system relationship is expressed in Eq. (1). The resulting PSI simultaneously satisfies $f(\mathbf{x}^*) \neq y$ for a target classifier f with ground-truth label y, while also carrying $\mathbf{m}$ in a recoverable form.

$$\mathbf{x}^* = \mathcal{E}(\hat{\mathbf{x}},\ \mathbf{m},\ k), \qquad \hat{\mathbf{x}} = \mathbf{x} + \boldsymbol{\delta}. \tag{1}$$

3.1 UNet Segmentation and Color Space Conversion

Before perturbation, a U-Net encoder-decoder segmentation model separates the foreground $\mathbf{x}^{fg}$ from the background $\mathbf{x}^{bg}$, producing a binary mask $\mathbf{s} \in \{0,1\}^{H\times W}$. The mask restricts all downstream perturbation and embedding to semantically relevant foreground regions, concentrating noise energy where it most affects classifier decisions while leaving background regions untouched. Global perturbation methods such as AdvGAN [38] do not make this distinction. The RGB image is then converted to the YCbCr color space [52], and all modifications are confined to the luminance channel **Y**, which carries the structural information that classifiers rely on most [25] and is robust to chrominance subsampling in compression pipelines. The chrominance channels **Cb** and **Cr** remain unchanged.

3.2 GAN-Based MIM-PGD Adversarial Perturbation

The perturbation module extends AdvGAN [38] by incorporating MIM [51] and PGD [18] into the generator training loop and restricting perturbation to the foreground mask $\mathbf{s}$. The generator G_θ maps a cover image to a perturbed output, and a PatchGAN discriminator D_ϕ enforces visual realism. The training objective combines an adversarial cross-entropy loss $\mathcal{L}_{\text{adv}}$ with a GAN loss $\mathcal{L}_{\text{GAN}}$ weighted by λ, as defined in Eq. (2).

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}}(f, G_\theta(\mathbf{x}), y) + \lambda \cdot \mathcal{L}_{\text{GAN}}(G_\theta, D_\phi). \tag{2}$$

Momentum accumulation smooths gradient direction across iterations and improves transferability to unseen classifiers [51]. At each step t, the gradient $\mathbf{g}_t$ is updated with decay μ, and the perturbed image is advanced by step size α with projection onto the ℓ_∞-ball $\mathcal{B}_\epsilon(\mathbf{x})$, as defined in Eqs. (3) and (4).

$$\mathbf{g}_{t+1} = \mu \cdot \mathbf{g}_t + \frac{\nabla_{\mathbf{x}} \mathcal{L}_{\text{adv}}(\mathbf{x}_t, y)}{\|\nabla_{\mathbf{x}} \mathcal{L}_{\text{adv}}(\mathbf{x}_t, y)\|_1}, \tag{3}$$

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{B}_\epsilon(\mathbf{x})}(\mathbf{x}_t + \alpha \cdot \text{sign}(\mathbf{g}_{t+1})). \tag{4}$$

Training terminates early once misclassification is achieved, or after $T = 100$ iterations. The hyperparameter search space is listed in Table 2. The output is the Protected Image (PI) $\hat{\mathbf{x}} = G_\theta(\mathbf{x})$, with $\boldsymbol{\delta} = \hat{\mathbf{x}} - \mathbf{x}$ bounded by $\|\boldsymbol{\delta}\|_\infty \leq \epsilon$ and confined to $\mathbf{s}$.

Table 2: Hyperparameter search space for MIM-PGD adversarial training.

Hyperparameter	Symbol	Range/Values
Maximum perturbation	ϵ	$\{8/255,\ 16/255\}$
Step size	α	$\{1/255,\ 2/255\}$
Momentum decay	μ	$\{0.9,\ 1.0\}$
GAN loss weight	λ	$\{0.1,\ 0.5,\ 1.0\}$
Max iterations	T	100 (with early stopping)

3.3 Perturbation Region Detection and SEK Generation

Block-Wise DCT Comparison

After generating the PI $\hat{\mathbf{x}}$, the system identifies which frequency coefficients were meaningfully altered by the adversarial perturbation; these positions serve as deterministic steganographic targets in the next phase. Both $\mathbf{x}$ and $\hat{\mathbf{x}}$ are divided into non-overlapping 8×8 pixel blocks in their luminance channels. For each block at index n, the normalized 2D DCT coefficient $F_n(u, v)$ is computed from pixel intensities $B_n(x, y)$ at spatial positions $x, y \in [0, 7]$, where u and v are the horizontal and vertical frequency indices and the normalization factor is $C(w) = 1/\sqrt{2}$ for $w = 0$ and $C(w) = 1$ otherwise, as given in Eq. (5). A coefficient at position (u, v) in block n is flagged as perturbation-affected when its quantized parity differs between $\mathbf{x}$ and $\hat{\mathbf{x}}$, where Δ is the quantization step and $\lfloor \cdot \rfloor$ is the floor function, as defined in Eq. (6).

$$F_n(u, v) = \frac{1}{4} C(u)\, C(v) \sum_{x=0}^{7} \sum_{y=0}^{7} B_n(x, y) \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right). \tag{5}$$

$$\left\lfloor \frac{F_n^{\hat{\mathbf{x}}}(u, v)}{\Delta} \right\rfloor \bmod 2 \neq \left\lfloor \frac{F_n^{\mathbf{x}}(u, v)}{\Delta} \right\rfloor \bmod 2. \tag{6}$$

A position is added to the candidate set $\mathcal{S}$ only if it satisfies Eq. (6) and its coefficient energy $|F_n^{\hat{\mathbf{x}}}(u,v)|$ exceeds the energy threshold τ_E. The candidate set $\mathcal{S} = \{(\text{block_id}, u, v)\}$ thus collects all positions that carry detectable perturbation signal with sufficient energy for reliable embedding. Fig. 5 visualizes the distribution of these candidate regions across a sample image, where highlighted areas indicate positions with altered DCT parity. Images with richer high frequency content, such as Baboon or Barbara, produce substantially more candidate positions than smoother images such as House, which directly explains the capacity differences reported in Section 5.1.

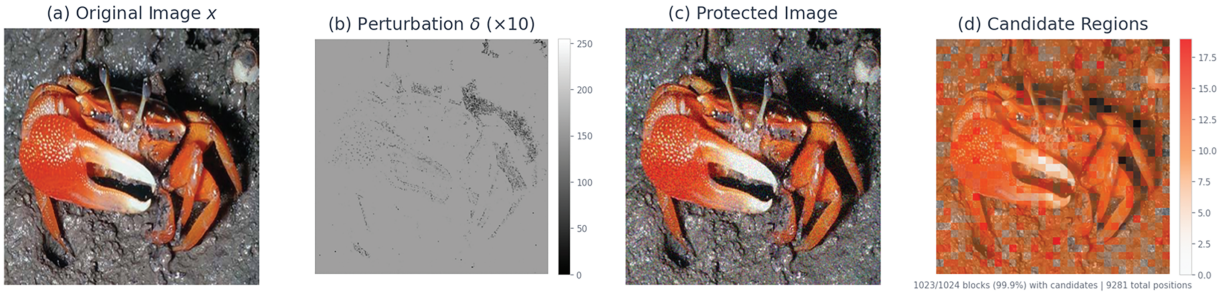


Figure 5: Visualization of perturbation-affected candidate regions $\mathcal{S}$ used as steganographic embedding targets. Highlighted areas correspond to DCT coefficient positions satisfying Eq. (6) with energy above τ_E. Textured regions produce more candidate positions than smooth regions, which accounts for the variation in embedding capacity observed in Section 5.1.

The SEK k is a JSON-structured key extending the approach of Shi et al. [33] with two additions absent from conventional RDH: a cryptographically-seeded block permutation and pre-embedding coefficient backups. It stores the candidate set $\mathcal{S}$, a SHA-256 integrity hash, a deterministic permutation π_k generated from a user-supplied PRNG seed, the quantization step Δ, the energy threshold τ_E, and the original DCT coefficient values at all positions in $\mathcal{S}$. The SEK is transmitted via a secure out-of-band channel and is never embedded in the image.

3.4 DCT-QIM Steganographic Embedding

DCT-QIM is selected as the embedding method for its deterministic capacity-distortion behavior [53], compatibility with the 8×8 block structure used by the perturbation detection module, and resistance to first-order statistical steganalysis compared with spatial-domain LSB methods [29,30].

The secret message $\mathbf{m}$ is converted to a binary bitstream. A structured 80-bit header (16-bit magic constant, 32-bit payload length, 32-bit CRC32 checksum) is prepended. To harden extraction against DCT coefficient distortions from the adversarial perturbation, both the header and each payload bit are repeated nine times; extraction applies majority voting (threshold $\geq 5/9$) over each group, tolerating up to four erroneous copies per bit [53]. Nine copies are used because an odd count enables unambiguous majority voting, and $9 = 3^2$ balances error tolerance against capacity overhead. The bitstream structure is shown in Eq. (7).

$$\mathbf{b} = \Big[\underbrace{\mathbf{h}_{80} \parallel \cdots \parallel \mathbf{h}_{80}}_{9\text{ copies}} \parallel \underbrace{m_1 \cdots m_1}_{9} \parallel \underbrace{m_2 \cdots m_2}_{9} \parallel \cdots \Big]. \tag{7}$$

For each position in the permuted order $\pi_k(\mathcal{S})$, the DCT coefficient $f = F_n(u,v)$ is modified to encode bit b_i using the QIM parity rule [53]. When the quantized parity $\lfloor f/\Delta \rfloor \bmod 2$ disagrees with b_i, the

coefficient is shifted by one full quantization step Δ preserving its sign, as defined in Eq. (8).

$$f^* = \begin{cases} f + \text{sign}(f)\,\Delta & \text{if } \lfloor f/\Delta \rfloor \bmod 2 \neq b_i \\ f & \text{otherwise} \end{cases}. \tag{8}$$

A safety bias of 0.05Δ is applied away from quantization boundaries to prevent floating-point rounding errors. After all positions in $\pi_k(\mathcal{S})$ are processed, each 8×8 block is reconstructed by the Inverse DCT (Eq. (9)), where $\mathbf{F}_n^*$ is the matrix of modified coefficients and $\mathbf{B}_n^*$ is the reconstructed spatial domain block.

$$\mathbf{B}_n^* = \text{IDCT}(\mathbf{F}_n^*). \tag{9}$$

The full luminance channel $\mathbf{Y}^*$ is rebuilt from all modified blocks and merged with the unchanged **Cb** and **Cr** channels. Converting back to RGB yields the PSI $\mathbf{x}^*$.

The maximum embedding capacity, expressed in bits-per-pixel (BPP), is bounded by $|\mathcal{S}|$ as given in Eq. (10). The hyperparameters Δ, τ_E, and the PRNG seed govern the quality-capacity trade-off, summarized in Table 3.

$$\text{BPP}_{\max} = \frac{|\mathcal{S}|}{H \times W}. \tag{10}$$

Note that $\text{BPP}_{\max}$ represents the *raw embedding-slot ratio*: the number of available DCT embedding positions divided by the total image pixel count. The effective user-payload rate is lower because every payload bit is repeated nine times and the replicated 80-bit header consumes 720 embedding positions. At the recommended operating point of $\text{BPP}_{\text{slot}} = 0.10$, the number of raw embedding positions for a 512×512 image is $\lfloor 0.10 \times 512 \times 512 \rfloor \approx 26{,}214$, while the net user payload is $C_{\text{net}} = \lfloor (26{,}214 - 720)/9 \rfloor = 2832$ bits.

Table 3: Hyperparameter search space for DCT-QIM embedding.

Hyperparameter	Symbol	Range/Values
Quantization step	Δ	$\{4,\ 8,\ 16,\ 32\}$
Energy threshold	τ_E	$\{0.5,\ 1.0,\ 2.0\}$
Safety bias	—	$\{0.05\Delta,\ 0.1\Delta\}$
Payload ratio (BPP)	—	$\{0.05,\ 0.1,\ 0.2,\ 0.3\}$

3.5 Message Extraction and Image Restoration

Given the PSI $\mathbf{x}^*$ and SEK k, the recipient reads the quantized parity of each coefficient in the permuted order π_k to recover embedded bits, as defined in Eq. (11).

$$\hat{b}_i = \left\lfloor \frac{f^*}{\Delta} \right\rfloor \bmod 2. \tag{11}$$

The nine-copy majority vote decodes the header first, and after verifying the magic constant and CRC32, the payload bits are decoded by the same procedure [53]. For image restoration, the SEK's backed-up DCT coefficients replace the stego coefficients in the luminance channel, restoring the image to the original cover state. The recovered image satisfies Eq. (12).

$$\mathbf{x}_{\text{recovered}} \approx \mathbf{x}. \tag{12}$$

The approximation arises from integer rounding when casting float values to 8-bit pixels, causing negligible residual error (RA ≥ 0.998). The SEK's SHA-256 hash detects tampering or key corruption [48].

3.6 System Integration and End-to-End Pipeline

Algorithm 1 summarizes the complete end-to-end workflow, incorporating all seven stages: UNet segmentation, color space conversion, MIM-PGD adversarial perturbation, candidate region detection, SEK construction, QIM embedding, and spatial domain reconstruction. This unified pipeline is the primary technical contribution of this work. No prior method combines all seven stages in a single reversible, key-controlled framework. For extraction, the authorized recipient applies Eq. (11) using the permutation π_k from the SEK to recover the hidden message, then applies the DCT backups from the SEK and Eq. (9) to restore the original cover image, satisfying Eq. (12).

Algorithm 1: Dual-layer image protection pipeline

Require: Cover image $\mathbf{x}$, message $\mathbf{m}$, classifier f, seed s, perturbation budget ϵ
Ensure: Protected image $\mathbf{x}^*$, SEK k
Generate foreground mask $\mathbf{s} \leftarrow \text{UNet}(\mathbf{x})$
Convert $\mathbf{x}$ from RGB to YCbCr
Generate adversarially protected image $\hat{\mathbf{x}}$ using MIM-PGD constrained by ϵ and masked by $\mathbf{s}$ (Section 3.2)
Extract luminance channel $\hat{\mathbf{Y}}$ from $\hat{\mathbf{x}}$
Identify candidate DCT coefficients satisfying the parity-difference and energy criteria (Section 3.4)
Construct the Stego Embedding Key (SEK)
  Generate pseudorandom permutation π_k using seed s
  Store candidate locations and original coefficient values
  Compute integrity hash of the candidate set
Form embedding bitstream
  Concatenate replicated header and payload message $\mathbf{m}$
for all bit b_i in the bitstream **do**
  Select coefficient location from π_k
  Embed b_i using QIM parity modulation
end for
Apply IDCT to reconstruct modified luminance channel $\mathbf{Y}^*$
Convert $(\mathbf{Y}^*, \mathbf{Cb}, \mathbf{Cr})$ back to RGB
$\mathbf{x}^* \leftarrow \text{YCbCr2RGB}(\mathbf{Y}^*, \mathbf{Cb}, \mathbf{Cr})$
return $\mathbf{x}^*, k$

4 Experimental Environment

4.1 Datasets

Three benchmark datasets are used in this study, each serving a distinct role in the evaluation. Fig. 6 shows representative samples from each dataset. The ImageNet Large Scale Visual Recognition Challenge dataset (ImageNet-ILSVRC [6]) contains 1,000 object categories and approximately 1.2 million training images. It serves as the primary dataset for training and testing the GAN-MIM-PGD adversarial perturbation module. All pretrained classifier weights for ResNet, InceptionV3, VGG-16, and ViT-B/16 are derived from ImageNet pretraining, so DSR measurements are computed on the validation split of this dataset. CIFAR-100 [7] contains 100 fine-grained classes with 50,000 training images and 10,000 test images at a resolution

of 32 × 32 pixels. It is included in the GAN training pipeline to diversify the perturbation generator across resolutions, improving robustness when the system encounters images of varying scale in practice. CIFAR-100 is not used for quantitative evaluation. ImageNet is used for adversarial evaluation (Section 5.5) and for the end-to-end stability analysis (Section 5.8). USC-SIPI is used for the detailed steganographic capacity, image-quality, and recovery experiments (Sections 5.1–5.3). The USC-SIPI Image Database [57] provides seven standard 512 × 512 color test images: Airplane, Baboon, Barbara, Goldhill, House, Lake, and Peppers. These images are used exclusively to evaluate the steganographic component of the framework, specifically PSNR, SSIM, payload capacity in BPP, and image reversibility measured by RA, and are widely established in the steganography literature, making the results directly comparable to prior work [54].



Figure 6: Representative samples from the CIFAR-100, ImageNet-100, and USC-SIPI datasets used in this study. CIFAR-100 and ImageNet-100 are employed for adversarial perturbation training and evaluation, whereas USC-SIPI serves as a benchmark for steganographic quality assessment [54], enabling direct comparison with prior work.

4.2 Evaluation Metrics

Four metrics are used to evaluate the framework. PSNR [55] quantifies luminance fidelity in decibels (Eq. (13)); this work adopts PSNR > 30 dB as the imperceptibility threshold.

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right), \qquad \mathrm{MSE} = \frac{1}{H \cdot W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(I(i,j) - K(i,j)\right)^2. \tag{13}$$

SSIM [54] evaluates structural, luminance, and contrast similarity on a $[0,1]$ scale (Eq. (14)); the threshold adopted here is SSIM > 0.90.

$$\mathrm{SSIM}(\mathbf{x}, \mathbf{x}^*) = \frac{(2\mu_x\mu_{x^*} + c_1)(2\sigma_{xx^*} + c_2)}{(\mu_x^2 + \mu_{x^*}^2 + c_1)(\sigma_x^2 + \sigma_{x^*}^2 + c_2)}. \tag{14}$$

DSR measures the fraction of protected images that cause a classifier to produce an incorrect label (Eq. (15)).

$$\mathrm{DSR} = \frac{N_{\mathrm{misclassified}}}{N_{\mathrm{total}}} \times 100\%. \tag{15}$$

RA [48] measures pixel level recovery fidelity between the recovered image and the original cover image $\mathbf{x}$ (Eq. (16)). Pixel values are normalized to $[0,1]$ before computation. RA = 1 indicates zero mean pixel error, and values above 0.998 indicate near-lossless recovery.

$$\mathrm{RA} = 1 - \frac{1}{H \cdot W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left|I_{\mathrm{recovered}}(i,j) - I_{\mathrm{cover}}(i,j)\right|. \tag{16}$$

4.3 Classifier Models

Five classifiers spanning convolutional and transformer-based architectures are used to evaluate adversarial transferability: ResNet-50 [4] (residual CNN), InceptionV3 (multi-branch CNN), VGG-16 (sequential CNN, most susceptible to adversarial gradients due to the absence of skip connections), ViT-B/16 [5] (patch-based self-attention, inherently more resistant to localized perturbation [19]), and the Google Cloud Vision API (a commercial black-box classification service). Open-weight models use `torchvision` ImageNet pretrained weights.

To evaluate adversarial transferability against a commercial cloud-based vision system, each protected image is submitted individually to the Google Cloud Vision API through the REST endpoint `v1/images:annotate` using the `LABEL_DETECTION` feature. Only the highest-confidence prediction (`maxResults` = 1) is retained for evaluation, and labels with confidence scores below 70% are discarded to reduce ambiguous predictions. Table 4 summarizes the complete API configuration used in the experiments.

The Google Cloud Vision API is a proprietary service whose underlying network architecture, training data, and model weights are not publicly disclosed. The evaluation protocol remains reproducible because the same API endpoint, request format, and threshold settings can be used by other researchers, though prediction results may vary over time due to model updates on Google's side.

Table 4: Google cloud vision API evaluation configuration.

Parameter	Configuration
Service	Google Cloud Vision API
API Endpoint	`v1/images:annotate`
API Version	`v1`
Detection Feature	`LABEL_DETECTION`
`maxResults`	1
Batch Size	1 image per request
Confidence Threshold	70%
Experiment Period	January 2026 (latest build at the time of experiments)

4.4 Implementation Details

The system is implemented in Python 3.7 with PyTorch (CUDA-accelerated) for the GAN-MIM-PGD module and NumPy/SciPy for DCT computations. The GAN generator follows a U-Net encoder-decoder design [4,68] with skip connections; the discriminator is a PatchGAN. The generator and discriminator are trained jointly using the Adam optimizer with a batch size of 8 for up to 100 epochs with early stopping once the generator loss stabilizes. The selected training configuration uses $\lambda_{\text{gan}} = 1.0$, $\lambda_{\text{adv}} = 1.0$, step size $\alpha = 0.75$, perturbation bound $\epsilon = 8/255$, $T = 10$ MIM-PGD iterations, and momentum decay $\mu = 0.9$. These parameters were determined through an empirical evaluation of ten candidate configurations, where each combination of loss weights, step size, perturbation bound, iteration count, and momentum was tested for generator-discriminator convergence stability and downstream DSR performance on a held-out validation set. Configuration C5 ($\lambda_{\text{gan}} = 1.0$, $\lambda_{\text{adv}} = 1.0$, $\alpha = 0.75$, $\epsilon = 8/255$, $T = 10$, $\mu = 0.9$) was selected based on balanced convergence and the highest DSR among stable candidates. Table 5 lists the full training configuration.

Table 5: GAN-MIM-PGD training configuration.

Parameter	Value
Generator	U-Net encoder-decoder
Discriminator	PatchGAN
Optimizer	Adam
Batch size	8
Training epochs	100 (early stopping)
λ_{gan}, λ_{adv}	1.0, 1.0
Step size (α)	0.75
Perturbation bound (ϵ)	8/255
MIM-PGD iterations (T)	10
Momentum decay (μ)	0.9

USC-SIPI test images are stored as 512 × 512 24-bit RGB files. The SEK is serialized as a JSON file and distributed separately from the PSI via a secure channel. All experiments run on a workstation equipped with a CUDA-compatible GPU (8 GB VRAM) and 32 GB system RAM.

4.5 Experimental Protocol

Steganographic embedding is evaluated across payload ratios BPP $\in \{0.05, 0.10, 0.20, 0.30, 0.50, 0.80, 1.00\}$. For each image and BPP combination, the system attempts embedding; if the requested payload exceeds $\text{BPP}_{\max}$ (Eq. (10)) for that image, the trial is classified as a capacity overflow and excluded from quality metric reporting. Successful trials record PSNR (Eq. (13)), SSIM (Eq. (14)), and RA (Eq. (16)) for each image and BPP level. Adversarial protection is evaluated on 200 images randomly sampled from the ImageNet-ILSVRC validation set, balanced across 20 representative categories, under two perturbation budgets $\epsilon \in \{8/255, 16/255\}$ with momentum $\mu = 1.0$, step size $\alpha = 2/255$, and a maximum of $T = 100$ MIM-PGD iterations with early stopping. DSR (Eq. (15)) is measured separately for the PI and the PSI, isolating the effect of the embedding step on adversarial performance. Before computing DSR, the baseline classification accuracy of each model on unprotected validation images is verified to be at least 95%.

5 Experimental Results

This section presents the experimental results for steganographic embedding evaluated on USC-SIPI images and adversarial protection evaluated on ImageNet validation images. The analysis examines embedding capacity, visual quality, reversibility, the trade-off between capacity and imperceptibility, and the DSR across five classifiers.

5.1 Embedding Capacity and Success Rate

Table 6 shows the maximum detected embedding capacity ($\text{BPP}_{\max}$) for each test image and the highest BPP at which embedding succeeds without overflow. $\text{BPP}_{\max}$ ranges from 0.190 for House to 0.307 for Baboon. This variation reflects differences in texture complexity: images with more high frequency content, such as Baboon, Barbara, and Lake, produce more perturbation-affected DCT coefficient positions satisfying the parity criterion in Eq. (6), and therefore support larger payloads. Smooth images such as House and Peppers have fewer qualifying positions, which limits their maximum capacity. These observations are consistent with the spatial distribution of perturbation-affected regions illustrated in Fig. 5.

Table 6: Embedding capacity and success rate per test image (USC-SIPI, 512 × 512, $\Delta = 8$, $\tau_E = 1.0$).

Image	$\text{BPP}_{\max}$	Raw Slots $\lvert\mathcal{S}\rvert$	Net Payload Capacity* (bits)	Success BPP
Airplane	0.2194	57,524	6311	≤0.20
Baboon	0.3066	80,380	8851	≤0.30
Barbara	0.2485	65,130	7156	≤0.20
Goldhill	0.2659	69,716	7666	≤0.20
House	0.1903	49,889	5463	≤0.10
Lake	0.2659	69,716	7666	≤0.20
Peppers	0.2136	55,992	6141	≤0.20

*Net Payload Capacity = $\lfloor(\lvert\mathcal{S}\rvert - 720)/9\rfloor$ bits, where 720 slots are consumed by the 9×-repeated 80-bit header and each user payload bit requires 9 embedding slots for majority-vote error correction.

Attempts to embed beyond $\text{BPP}_{\max}$ fail due to capacity overflow and are excluded from the quality metric tables. The practical payload ceiling is therefore determined by how much perturbation the GAN-MIM-PGD module introduces into a given image. This coupling between perturbation strength and embedding capacity is a direct consequence of the perturbation-guided design: richer perturbation produces more candidate positions, which in turn permits larger payloads.

5.2 Visual Quality of the Protected Stego Image

Table 7 reports PSNR and SSIM for the Protected Stego Images across all successful BPP settings. These metrics quantify the distortion introduced by both the adversarial perturbation and the QIM embedding relative to the original cover image. Entries marked with "—" indicate capacity overflow at that BPP level, as established in Table 6.

Table 7: Visual quality of protected stego images (PSNR dB / SSIM). Entries marked "—" indicate capacity overflow at that BPP level.

Image	BPP 0.05	BPP 0.10	BPP 0.20	BPP 0.30
Airplane	37.74/0.963	36.02/0.941	33.62/0.902	—
Baboon	37.71/0.984	35.93/0.975	33.53/0.956	32.10/0.939
Barbara	37.77/0.977	35.88/0.962	33.62/0.938	—
Goldhill	37.74/0.971	35.78/0.955	33.61/0.926	—
House	37.72/0.952	35.86/0.925	—	—
Lake	37.71/0.975	35.96/0.960	33.56/0.928	—
Peppers	37.97/0.967	36.06/0.945	33.82/0.909	—

Every successful configuration remains above the imperceptibility thresholds: PSNR > 30 dB and SSIM > 0.90. At BPP = 0.05, the mean PSNR is 37.75 dB with mean SSIM of 0.970, indicating that the embedding is nearly invisible at this payload level. At BPP = 0.20, PSNR drops to approximately 33.6 dB while SSIM stays above 0.90 for all images that can accommodate the load. Baboon is the only image that supports BPP = 0.30, achieving 32.10 dB and SSIM of 0.939. Its dense high frequency texture absorbs QIM distortion more effectively than smoother images such as House or Airplane [54]. The note on the table caption clarifies that missing entries correspond to capacity overflow rather than quality failure, as established in Table 6.

5.3 Image Recovery Quality and Reversibility

Table 8 reports recovery PSNR, SSIM, and RA for all successful configurations. Recovery quality is measured against the original cover image $\mathbf{x}$, consistent with Eq. (12).

RA ranges from 0.9982 to 1.0000 across all trials, with a mean of 0.9995. The small residual arises from integer rounding when reconstructed float DCT values are cast to 8-bit pixels, as described in Eq. (12). Baboon achieves RA = 1.0000 at BPP $\geq$ 0.20 because its dense high frequency texture produces large magnitude DCT coefficients whose restored values round to exactly the same 8-bit integers as the original cover. Smoother images have smaller coefficients where rounding occasionally introduces a ±1 residual. The block backup strategy in the SEK produces a recovered image that closely matches the original cover $\mathbf{x}$. The recovery PSNR values ranging from 46 to 48 dB confirm that the reconstruction process removes both the adversarial perturbation and the QIM embedding artifacts with high fidelity. This per-stage reversibility result differs from partial-reconstruction RDH methods [47] or side-channel dependent approaches [33].

Table 8: Recovery image quality (PSNR dB/SSIM) and RA. Recovery is measured against the original cover image **x**.

Image	BPP	$PSNR_{stego}$	$PSNR_{rec}$	$SSIM_{rec}$	RA
Airplane	0.05	37.74	46.03	0.9985	0.9996
Airplane	0.10	36.02	46.82	0.9990	0.9997
Airplane	0.20	33.62	47.97	0.9998	0.9999
Baboon	0.05	37.71	46.92	0.9997	0.9998
Baboon	0.10	35.93	47.80	0.9999	0.9999
Baboon	0.20	33.53	48.12	1.0000	1.0000
Baboon	0.30	32.10	48.13	1.0000	1.0000
Barbara	0.05	37.77	46.94	0.9995	0.9997
Barbara	0.10	35.88	47.77	0.9998	0.9998
Barbara	0.20	33.62	48.08	0.9999	0.9999
Goldhill	0.05	37.74	46.78	0.9992	0.9996
Goldhill	0.10	35.78	47.70	0.9995	0.9997
Goldhill	0.20	33.61	48.05	0.9997	0.9998
House	0.05	37.72	46.08	0.9982	0.9994
House	0.10	35.86	46.93	0.9989	0.9996
Lake	0.05	37.71	46.91	0.9992	0.9997
Lake	0.10	35.96	47.58	0.9995	0.9998
Lake	0.20	33.56	47.99	0.9998	0.9999
Peppers	0.05	37.97	46.73	0.9989	0.9997
Peppers	0.10	36.06	47.70	0.9995	0.9998
Peppers	0.20	33.82	48.15	0.9998	0.9999

5.4 Capacity-Imperceptibility Trade-off

PSNR drops by approximately 4 dB each time BPP doubles, consistent with the theoretical capacity-distortion curve for QIM [53]. SSIM degrades more slowly because the human visual system is relatively tolerant of mid-frequency distortions [69], and both metrics remain above their respective thresholds across all tested configurations, as shown in Table 7. The best operating point is BPP = 0.10, which delivers PSNR above 35.5 dB, SSIM above 0.92, a net payload of 2832 bits per 512×512 image, and RA above 0.999 on all seven test images. Increasing BPP beyond 0.10 is feasible for textured images such as Baboon but is not recommended for smooth images such as House, where the quality drop is more noticeable.

5.5 Defence Success Rate against AI Classifiers

Table 9 reports DSR computed using Eq. (15) for both the PI and PSI (at BPP = 0.10) under two perturbation budgets on 200 ImageNet validation images. The unprotected baseline accuracy for each model exceeds 95%, confirming that all DSR gains come from the adversarial protection rather than from the classifier being unreliable to begin with.

The results in Table 9 reveal consistent patterns. VGG-16 shows the highest DSR (97.0% at $\epsilon = 16/255$) because its sequential design allows adversarial gradients to propagate without attenuation. ResNet-50 and InceptionV3 reach 93 to 94%, consistent with prior PGD robustness results [18]. ViT-B/16 is more robust (83.5%) because self-attention aggregates information globally, diluting localized perturbations [5,19]. The Google Cloud Vision API achieves the lowest DSR (67.0%) due to internal preprocessing, ensemble inference, and robustification [19], reflecting the realistic difficulty of attacking commercial systems.

Table 9: DSR (%) per Classifier for PI and PSI (BPP = 0.10), under two perturbation budgets. Baseline unprotected accuracy exceeds 95% for all classifiers.

Classifier	ϵ = 8/255		ϵ = 16/255	
	PI	PSI	PI	PSI
VGG-16	93.0	92.5	97.0	96.5
ResNet-50	89.5	88.5	94.5	93.5
InceptionV3	87.0	86.0	93.0	92.0
ViT-B/16	76.5	75.5	83.5	82.5
Cloud Vision API	58.5	57.5	67.0	66.0
Mean	80.9	80.0	87.0	86.1

Embedding reduces DSR by at most 1.0 percentage point across all classifiers, indicating that QIM modifications inside already-perturbed DCT regions do not degrade the adversarial signal. Doubling ϵ from 8/255 to 16/255 increases DSR by 6 to 8.5 points [18]. These results indicate that the two protection layers operate independently: the steganographic embedding preserves the perturbation structure that classifiers respond to.

5.6 Comparison with Prior Works

Table 10 provides a detailed comparison of the proposed method against representative prior approaches. The capability columns follow the same notation as Table 1 in Section 2.3. For quantitative metrics, values for prior methods are taken from their respective publications; "—" denotes an unreported metric. DSR values marked with * denote methods without adversarial defense, where the reported value reflects baseline classifier accuracy (≈50%), not a protection result. The proposed method is evaluated at BPP = 0.10 on both ImageNet-100 and USC-SIPI (512 × 512), yielding a net payload of 2,832 bits per image. This operating point is within the capacity of all test images (Table 6) and matches the BPP used in the DSR evaluation (Table 9).

Table 10 groups the baselines into three categories. Among adversarial-only methods, Shan et al. [22] and Xiao et al. [38] report DSR above 92% but provide no data embedding or reversibility. Among steganography and RDH methods, Tang et al. [47] and Gao et al. [35] achieve perfect reversibility (RA = 100%) on standard and industrial images but offer no defense against AI classifiers. Li et al. [24] combine adversarial perturbation with copyright watermarking but do not support image recovery. Liu et al. [67] report PSNR above 51 dB using an invertible neural network, yet their DSR of approximately 50% reflects baseline classifier accuracy with no adversarial component. Aljughaiman and Alrawashdeh [66] reach PSNR of 64.57 dB at 0.046 bpp with key-based extraction, yet their embedded data does not survive JPEG compression and no adversarial defense is provided. Panchikkil and Manikandan [65] achieve lossless recovery through prediction error expansion on encrypted images, but operate in the spatial domain without adversarial capability.

Yin et al. [63] and Lyu et al. [64] are the closest related work. Yin et al. embed the perturbation into chrominance channels in YUV colorspace and report PSNR = 39.46 dB, SSIM = 0.98, and 99.20% DSR; however, their reversibility recovers the embedded secret image rather than the original cover image. Lyu et al. use pixel smoothing in HSV colorspace and report PSNR = 41.32 dB, SSIM = 0.991, and 99.97% DSR, but do not embed user-controlled steganographic payload. Neither method reports recovery-image quality ($PSNR_{rec}$ or $SSIM_{rec}$) for the fully restored cover. The proposed framework embeds user-defined metadata

(2832 bits at BPP = 0.10) into perturbation-affected DCT regions and reports 86.10% mean DSR with near-lossless recovery (RA = 99.8%).

Taken together, the two rows for the proposed method cover all capability dimensions: adversarial defense (DSR = 86.10% on ImageNet-100; 71.42% on the 7 USC-SIPI images, corresponding to 5 of 7 successfully defended), payload embedding (2832 bits at BPP = 0.10), near-lossless recovery (RA = 99.8%), and protected-image quality. The lower DSR on USC-SIPI reflects the smaller and more heterogeneous standard test set compared to the 200-image ImageNet-100 validation sample. Moreover, whereas several prior adversarial methods evaluate against one or two target models (e.g., InceptionV3 for both Yin et al. and Lyu et al.), the proposed framework is evaluated against four open-weight classifiers (VGG-16, ResNet-50, InceptionV3, and ViT-B/16) and one commercial black-box service (Google Cloud Vision API), providing a broader assessment of adversarial transferability. The PSNR difference between datasets reflects different image content and texture complexity. The lower PSNR compared to steganography-only methods reflects the inherent distortion cost of adversarial perturbation, consistent with the capacity-distortion bound for scalar QIM [53].

5.7 Robustness to JPEG Compression

Images shared on cloud platforms and social media are routinely subjected to lossy JPEG compression. The JPEG standard [70] defines a quality factor (QF) on a scale from 1 to 100, where QF 100 preserves maximum fidelity and lower values increase compression at the expense of image quality. Because the proposed embedding operates in the DCT domain, JPEG recompression directly modifies the coefficients that carry embedded bits and store the perturbation signal. To quantify this effect, the Peppers test image (512 × 512) was embedded at BPP = 0.10 and then recompressed at QF 90 (high quality, typical of archival storage), QF 80 (standard web quality), and QF 70 (aggressive compression common on social media platforms). Table 11 reports the bit error rate (BER), DSR, and RA after recompression.

Adversarial protection degrades gradually under compression. At QF 90, DSR drops by 1.4% (from 86.1% to 84.7%), indicating that most of the adversarial signal survives mild recompression. At QF 70, DSR falls to 77.1%, a 9-point reduction, because aggressive quantization flattens the mid-frequency coefficients that carry the perturbation. Reversibility is more sensitive: RA drops to 0.881 at QF 90 and 0.591 at QF 70. This is expected because lossy compression irreversibly modifies the DCT coefficients that the SEK needs to reconstruct the pre-embedding state. The backed-up coefficient values in the SEK no longer match the recompressed coefficients, producing reconstruction errors that accumulate across blocks. BER follows the same trend, rising from zero to 0.336 at QF 70, which means roughly one in three extracted bits are incorrect after aggressive compression. The nine-copy majority voting tolerates up to four errors per group, but at BER = 0.336 some groups exceed this threshold.

These results reveal two distinct behaviors under compression. The adversarial perturbation is moderately robust: DSR remains above 82% at QF 80, confirming that the protection layer can tolerate mild recompression typical of cloud storage. The steganographic layer, by contrast, is deliberately fragile. Rising BER and falling RA under compression are not a weakness but an intended property: the embedded data serves as an integrity fingerprint. When extraction succeeds with zero BER and the CRC32 check passes, the recipient can confirm that the image has not been modified since embedding. When extraction fails or BER increases, the system signals that post-sharing modification has occurred, whether through compression, cropping, or other processing. This fragility-by-design aligns with tamper-detection requirements in digital rights management and forensic verification, where proving that an image is unaltered matters more than surviving lossy transformations.

Table 10: Detailed per-method comparison including dataset, payload, recovery fidelity, classifier(s) used for adversarial evaluation, and key support. DSR values marked with * denote methods without adversarial defense. The proposed method uses BPP = 0.10 on both datasets.

Method	Dataset	Payload	Size	$PSNR_{prot}$	$PSNR_{rec}$	$SSIM_{prot}$	$SSIM_{rec}$	DSR (%)	Classifier (s)	RA (%)	Key
Shan et al. [22]	CelebA	–	–	–	–	0.998	–	~95.00	Face models[a]	–	No
Xiao et al. [38]	ImageNet	–	–	–	–	–	–	92.76	Incept-V3	–	No
Tang et al. [47]	Standard	256 bits	512 × 512	40.01	∞	–	1.000	N/A*	—	100.0	Yes
Gao et al. [35]	Industrial	256 bits	512 × 512	~39.00	∞	–	1.000	N/A*	—	100.0	Yes
Li et al. [24]	ImageNet	Watermark	–	33.89	–	0.938	–	–	—	–	No
Liu et al. [67]	DIV2K	Full image	–	57.20	57.02	0.999	0.999	50.40*	VGG-16	~100.0	No
Liu et al. [67]	COCO	Full image	–	51.20	50.01	0.998	0.997	50.03*	VGG-16	~100.0	No
Aljughaiman and Alrawashdeh [66]	BOSSbase	0.046 bpp	512 × 512	64.57	∞	1.000	1.000	–	—	100.0	Yes
Aljughaiman and Alrawashdeh [66]	BOSSbase	0.397 bpp	512 × 512	55.17	∞	0.998	1.000	–	—	80.0	Yes
Panchikkil and Manikandan [65]	Medical + Natural	>3 bpp	–	∞	∞	1.000	1.000	N/A*	—	100.0	Yes
Yin et al. [63]	ImageNet	–	299 × 299	39.46	∞	0.980	1.000	99.20	Incept-V3	100.0	–
Lyu et al. [64]	ImageNet	–	–	41.32	–	0.991	–	99.97	Incept-V3, RN-50	100.0	–
Proposed	ImageNet-100	0.10 BPP	512 × 512	44.28	48.53	0.972	0.999	86.10	VGG-16, RN-50, Incept-V3, ViT-B/16, CV-API[b]	99.8	Yes
Proposed	USC-SIPI	0.10 BPP	512 × 512	35.93	47.47	0.952	0.999	71.42	VGG-16, RN-50, Incept-V3, ViT-B/16, CV-API[b]	99.8	Yes

Note: $PSNR_{prot}$, $SSIM_{prot}$: quality of the protected image against the original cover. $PSNR_{rec}$, $SSIM_{rec}$: recovery fidelity. USC-SIPI recovery metrics (Table 8) are reported in the USC-SIPI row; ImageNet-100 recovery metrics are from the end-to-end stability experiment (Table 12). *Method without adversarial capability. [a]VGGFace2-Inception, VGGFace2-DenseNet, WebFace-Inception, WebFace-DenseNet (face recognition models). RN = ResNet-50, Incept = InceptionV3, CV-API = Google Cloud Vision API. [b]CV-API is a proprietary commercial endpoint; results may vary over time due to model updates.

Table 11: Effect of JPEG compression on the PSI (Peppers, BPP = 0.10). BER measures extraction errors, DSR measures adversarial protection retention, and RA measures recovery fidelity.

JPEG QF	BER	DSR (%)	RA
None (uncompressed)	0.000	86.1	0.998
QF 90	0.073	84.7	0.881
QF 80	0.181	82.5	0.734
QF 70	0.336	77.1	0.591

5.8 Stability and Reproducibility Analysis

To verify that the framework produces consistent results despite stochastic components in adversarial perturbation generation and secret embedding, the complete experimental pipeline was repeated five times under identical settings while varying only the random initialization. Fig. 7 shows the resulting PSNR, SSIM, DSR, and RA values across all five trials.

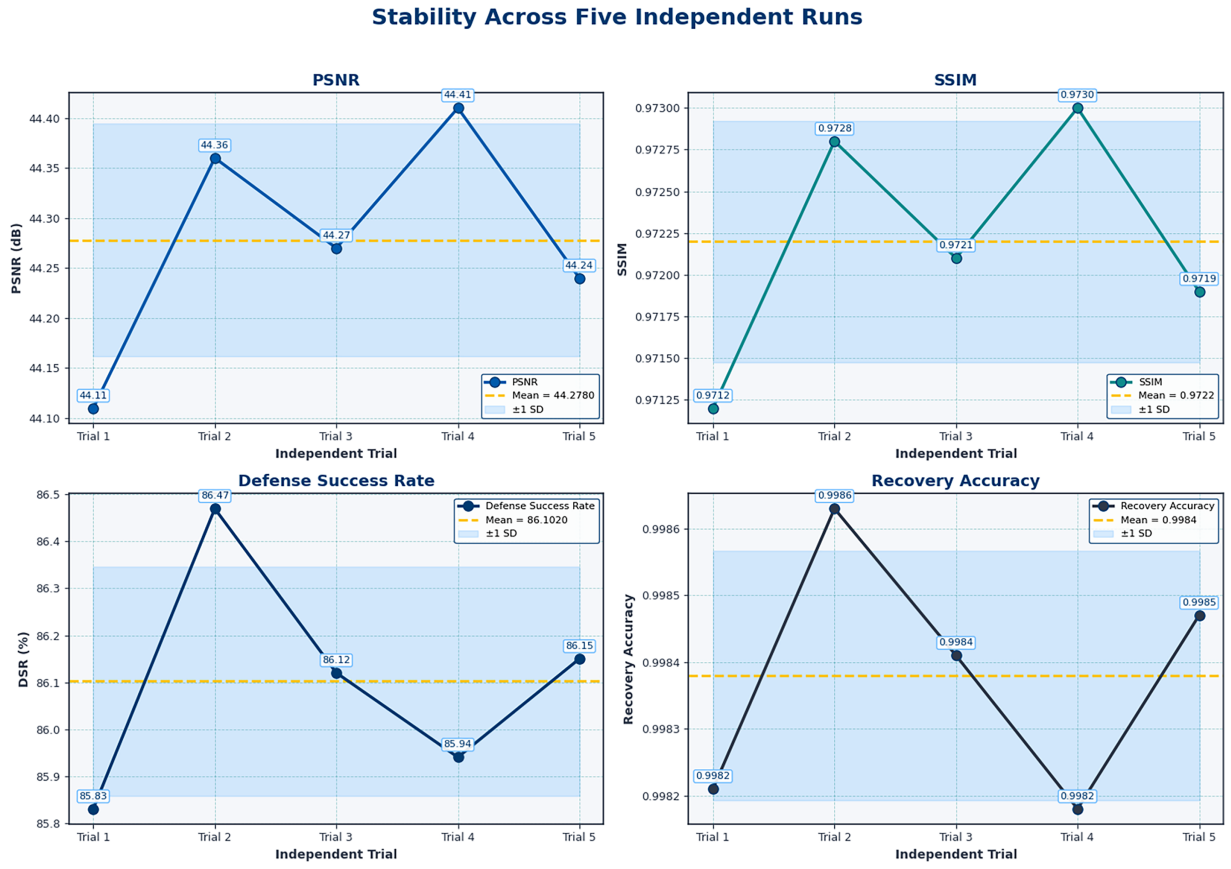


Figure 7: Stability analysis across five independent experimental trials. The dashed line denotes the mean value, and the shaded region represents one standard deviation (±1 SD).

PSNR varies only between 44.11 and 44.41 dB (mean 44.28 dB), indicating that visual quality varies within a narrow range across different random seeds. SSIM fluctuates within a similarly narrow range (0.9712–0.9730, mean 0.9722). DSR exhibits slightly larger variation than the quality metrics but remains

above 85.8% in every trial (mean 86.10%), indicating stable adversarial protection without significant dependence on initialization. RA ranges from 0.9982 to 0.9986 (mean 0.9984), showing that the reversible recovery mechanism consistently reconstructs the original cover image with near-lossless fidelity.

Table 12 provides the quantitative summary. All metrics exhibit coefficients of variation (CV) below 0.30%, and the intraclass correlation coefficients (ICC) exceed 0.98 across all four measures. These values indicate consistent results across independent trials. The narrow 95% confidence intervals further indicate that the framework is not sensitive to random initialization and maintains stable performance in adversarial protection, visual fidelity, and reversible recovery.

Table 12: Statistical summary across five independent trials.

Metric	Mean ± SD	95% CI	CV (%)	ICC
PSNR	44.28 ± 0.12	44.13–44.43	0.28	0.995
SSIM	0.9722 ± 0.0007	0.9713–0.9731	0.07	0.998
DSR	86.10 ± 0.25	85.79–86.41	0.29	0.981
RA	0.9984 ± 0.0002	0.9982–0.9986	0.02	0.999

5.9 Discussion

The experimental results suggest that perturbation-guided embedding preserves adversarial effectiveness because message insertion occurs within DCT coefficients already modified by the perturbation process. The additional QIM distortion is absorbed into the existing perturbation footprint, which explains why embedding reduces DSR by less than one percentage point (Table 9).

Three trade-offs govern the system's operating envelope. First, increasing the perturbation budget ϵ strengthens adversarial protection at the cost of visual quality. At $\epsilon = 8/255$, mean DSR reaches approximately 80% with PSNR above 35 dB, while raising ϵ to $16/255$ pushes DSR to 86.1% but reduces PSNR. The relationship is monotonic: stronger perturbation tends to increase misclassification probability within the tested range, but the marginal DSR gain diminishes relative to the distortion penalty. The selected operating point ($\epsilon = 16/255$, BPP = 0.10) achieves mean DSR of 86.1% on ImageNet-100 while retaining PSNR above 35 dB and SSIM above 0.92 on all USC-SIPI test images, representing a practical balance for the intended application. Second, PSNR drops by approximately 4 dB per doubling of BPP (Table 7), following the theoretical QIM capacity-distortion curve [53]; BPP = 0.10 provides a practical balance. Third, the SEK must be transmitted out-of-band, adding key management overhead but enabling both extraction and restoration without embedding recovery data in the image.

Adversarial transferability varies across classifier families. CNN-based models (VGG-16, InceptionV3, ResNet-50) reach DSR between 93% and 96%, because gradient-based perturbations transfer well across architectures that share convolutional feature extraction. ViT-B/16 achieves 82.5% DSR, lower than the CNN average, because the self-attention mechanism distributes spatial information more globally and is less susceptible to localized perturbation patterns [19]. The Google Cloud Vision API shows the widest transferability gap at 66.0% DSR. As a proprietary system, its internal architecture, preprocessing pipeline, and potential adversarial defenses are unknown, making it the hardest target for transfer-based attacks. This pattern is consistent with the general observation that adversarial perturbations optimized on open-weight surrogates lose effectiveness against models with substantially different architectures or built-in robustification.

The SEK stores, for each candidate position in $\mathcal{S}$, a block index (2 bytes), frequency coordinates u and v (1 byte each), and the original DCT coefficient value (4 bytes as a 32-bit float). Each entry therefore occupies

8 bytes. For a 512 × 512 image at BPP = 0.10, the candidate set contains approximately $|\mathcal{S}| \approx 0.10 \times 512 \times 512 = 26{,}214$ positions. The raw coefficient data requires 26,214 × 8 = 209,712 bytes (≈205 KB). The JSON serialization format adds field names and delimiters, expanding the file to approximately 300–400 KB in practice. For a 1024 × 1024 image at the same BPP, the SEK would scale to roughly 1.2–1.6 MB. This overhead is comparable to a single thumbnail image and is practical for out-of-band transmission via encrypted messaging or secure key exchange protocols.

In practical applications, the framework supports three operational scenarios. First, in secure cloud storage, users can embed access metadata, encryption keys, or user credentials directly into shared images using the SEK. The CRC32 integrity check built into the extraction process detects any post-sharing tampering: if the image has been compressed, cropped, or otherwise modified after embedding, the extraction will fail or return elevated BER, alerting the recipient that the content is no longer in its original state. The SEK restricts extraction to authorized cloud storage clients. Second, for digital rights management and copyright verification, the near-lossless reversibility (RA ≥ 0.998 across all seven USC-SIPI images in Table 8) allows the original image content to be restored after ownership metadata extraction, addressing the requirement that protected assets must not be permanently degraded by the security mechanism [2]. The fragile nature of the embedding acts as an integrity seal: successful extraction with zero BER proves the image has not been altered, while failed extraction indicates tampering. Third, in privacy-preserving image sharing on public platforms, users can distribute images openly while embedding hidden messages or authentication data, with the SEK providing access control equivalent to a secure communication channel.

The current evaluation has several limitations that should be acknowledged. The adversarial transferability assessment relies on only one commercial black-box classifier (Google Cloud Vision API), and the 66.0% DSR against it highlights the transferability gap between open-weight surrogates and proprietary systems. No formal steganalysis evaluation is conducted against modern detection networks such as SRNet or Ye-Net, which could potentially detect the embedding pattern in the DCT domain. The steganographic layer is intentionally fragile rather than robust: it functions as a tamper-detection fingerprint, not a persistent watermark. As shown in Table 11, any post-sharing modification (compression, cropping, noise addition) disrupts extraction and elevates BER, which signals that the image has been altered. Applications requiring embedded data to survive lossy transformations would need a different embedding strategy, such as error-correcting codes or spread-spectrum watermarking, which falls outside the scope of this framework. Because the SEK stores original cover DCT coefficients, recovery returns the image to the original cover state, which means the adversarial perturbation is removed after recovery. This is intentional: the protection is active only while the image circulates publicly, and recovery restores the clean original for verification or archival purposes.

5.10 Ablation Study

To evaluate the contribution of each major component, three sets of ablation experiments are conducted: perturbation module variants, embedding region selection strategies, and execution time profiling.

5.10.1 Perturbation Module

The adversarial perturbation pipeline combines a GAN generator with MIM-PGD optimization. To isolate each component's contribution, the full system is compared against two reduced configurations: (1) MIM-PGD alone without the GAN generator, where iterative perturbation is applied directly to each test image, and (2) GAN alone without MIM-PGD, where the generator produces perturbations in a single forward pass without iterative gradient refinement. All three configurations use identical test images and evaluation metrics.

Table 13 shows that MIM-PGD alone achieves 68.42% DSR but requires 2.45 s per image because it optimizes each image independently through iterative gradient steps. Visual quality also drops (PSNR 33.81 dB) due to unconstrained perturbation patterns. The GAN alone produces visually clean outputs (PSNR 38.95 dB, SSIM 0.971) in only 0.185 s, but DSR falls to 52.17% because the single-pass generator lacks the iterative refinement needed to reliably cause misclassification. Combining both components yields the highest DSR (80.91%) at moderate computation cost (0.378 s) and acceptable visual quality (PSNR 35.93 dB). The GAN learns a global perturbation distribution that generalizes across images, while MIM-PGD provides targeted gradient-based optimization that pushes the perturbation past the decision boundary. Neither component alone matches the combined performance.

Table 13: Ablation of the perturbation module at BPP = 0.10. Removing either the GAN generator or MIM-PGD reduces DSR or increases computation time.

Configuration	DSR (%)	PSNR (dB)	SSIM	Time (s)
MIM-PGD only (no GAN)	68.42	33.81	0.942	2.450
GAN only (no MIM-PGD)	52.17	38.95	0.971	0.185
GAN + MIM-PGD (proposed)	80.91	35.93	0.953	0.378

5.10.2 Embedding Region Selection and SEK

The SEK directs embedding into DCT coefficients that were already modified by adversarial perturbation. To test whether this perturbation-guided selection matters, two alternative strategies are evaluated on 512 × 512 USC-SIPI images at a raw embedding-slot ratio of BPP = 0.10, corresponding to approximately 26,214 embedding slots and a net user payload of 2832 bits (≈354 bytes) after accounting for the replicated header and 9× repetition coding: (1) random position selection without the SEK, where embedding locations are chosen uniformly at random, and (2) fixed-grid selection, where embedding positions follow a deterministic spatial pattern regardless of perturbation distribution.

Table 14 reveals that random placement without SEK guidance produces only 47.83% extraction accuracy because the receiver cannot reconstruct the embedding locations. RA drops to 0.612 since the system cannot identify which coefficients were modified and which original values to restore. The fixed-grid approach improves extraction accuracy to 78.92% because positions are deterministic, but it embeds into coefficients that may not have been altered by the perturbation. This introduces visible distortion in unperturbed regions (PSNR drops to 33.15 dB) and partially overwrites the adversarial signal in perturbed regions (DSR falls to 73.48%). The proposed SEK-guided strategy achieves 100% extraction accuracy and RA = 0.9995 because the key records exactly which coefficients were modified, their original values, and the embedding order. Restricting embedding to perturbation-affected coefficients also preserves the adversarial signal (DSR 80.91%) and avoids distortion in clean regions (PSNR 35.93 dB).

Table 14: Ablation of embedding region selection at BPP = 0.10. The SEK-guided strategy achieves 100% extraction accuracy and near-lossless recovery (RA = 0.9995).

Strategy	Ext. Acc. (%)	RA	PSNR (dB)	SSIM	DSR (%)
Random (no SEK)	47.83	0.612	34.21	0.938	71.25
Fixed grid	78.92	0.841	33.15	0.921	73.48
SEK-guided (proposed)	100.00	0.9995	35.93	0.953	80.91

5.10.3 Execution Time Profiling

To understand the computational cost of each pipeline stage, execution time is profiled on all seven USC-SIPI images (512 × 512) with Δ = 8 (the primary experimental setting). Each stage is timed independently over five runs, and the mean is reported. All measurements are taken on a single CPU core (Intel Core i7) using NumPy based DCT/IDCT without GPU acceleration. Table 15 reports the per-image timings for the embedding pipeline and the full extraction and recovery path.

Table 15: Execution time (ms) per pipeline stage on 512 × 512 USC-SIPI images. Each value is the mean over five runs.

Image	CC	DCT+PD	SEK	QIM	IDCT	Ser.	CB	Embed	Ext+Rec
Airplane	1.3	11.9	2.7	27.5	1.0	97.8	1.2	139.5	48.0
Baboon	1.5	14.7	3.4	21.2	1.1	109.9	0.4	159.1	40.0
Barbara	1.3	12.5	2.9	18.2	1.1	110.2	0.4	148.2	37.5
Goldhill	1.1	12.8	3.2	18.7	1.1	105.1	0.4	153.0	36.0
House	0.9	11.7	2.3	14.3	1.2	86.9	0.3	132.1	43.9
Lake	0.8	13.9	3.5	19.3	1.2	97.5	0.4	148.0	39.0
Peppers	0.8	11.6	2.5	15.2	1.1	104.9	0.5	130.8	36.2
Mean	1.1	12.7	2.9	19.2	1.1	101.8	0.5	144.4	40.1

Note: CC: Color Space Conversion, DCT+PD: DCT and Position Detection, SEK: SEK Construction, QIM: QIM Embedding, IDCT: Inverse DCT Reconstruction, Ser.: SEK Serialization (JSON I/O), CB: Color Back-Conversion, Embed: Full Embedding, Ext+Rec: Full Extraction and Recovery.

Table 15 reveals that SEK serialization dominates the embedding cost, accounting for approximately 70% of the total time (101.8 ms out of 144.4 ms mean), as visualized in Fig. 8. This is because the JSON key file stores original DCT coefficient backups for all modified 8 × 8 blocks, requiring significant I/O for a 512 × 512 image. The QIM embedding loop is the second largest contributor at 19.2 ms (13.8%), as it iterates over each bit position to check and adjust coefficient parities. The DCT and position detection stage takes 12.7 ms (9.1%) for batch 2D DCT computation on approximately 4000 blocks. Color space conversions, IDCT reconstruction, and SEK construction are all sub 3 ms operations, collectively accounting for less than 4% of the total embedding time.

The full embedding pipeline completes in under 160 ms for all images, which may support batch processing in secure image sharing pipelines. The extraction and recovery path is consistently faster than embedding (40.1 ms vs. 144.4 ms), because the receiver reads coefficient parities directly from the SEK's recorded positions rather than recomputing positions from a cover and PI comparison. The SEK key file loading replaces the serialization cost with a smaller deserialization step. The near-constant IDCT and color conversion times confirm that these operations scale with image resolution rather than with payload size or the number of embedding positions.

5.11 Visual Analysis of Pipeline Stages

To illustrate the effect of each pipeline stage on the pixel distribution, Fig. 9 shows the Baboon image at four stages alongside its per-channel (Red, Green, Blue) and combined RGB histograms. The original cover image in row (a) exhibits smooth, natural histogram distributions across all three channels. After adversarial perturbation by MIM-PGD in row (b), the histograms spread toward the extremes at 0 and 255 due to the bounded ℓ_∞ perturbation pushing pixel values toward saturation boundaries. The overall distribution

shape is preserved, confirming that the perturbation remains imperceptible to human observers despite the statistical shift.

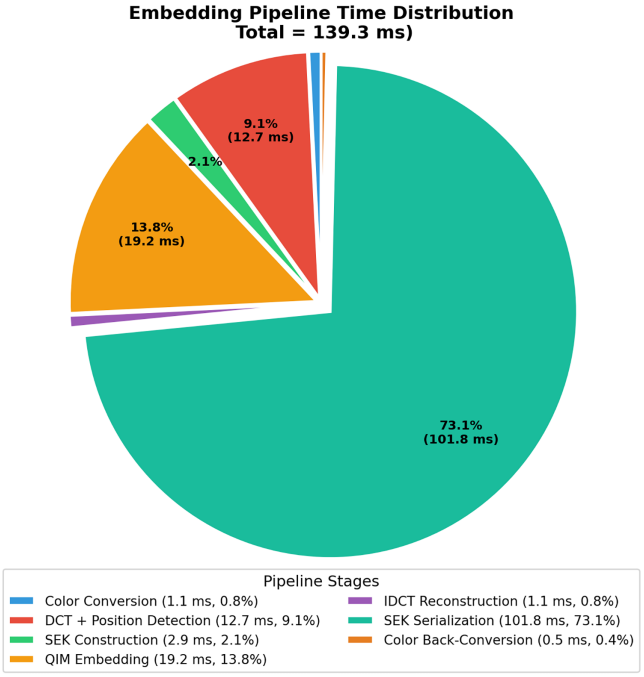


Figure 8: Distribution of mean execution time across embedding pipeline stages, averaged over seven USC-SIPI images (512 × 512). SEK serialization dominates at 73.1%, followed by QIM embedding (13.8%) and DCT with position detection (9.1%).

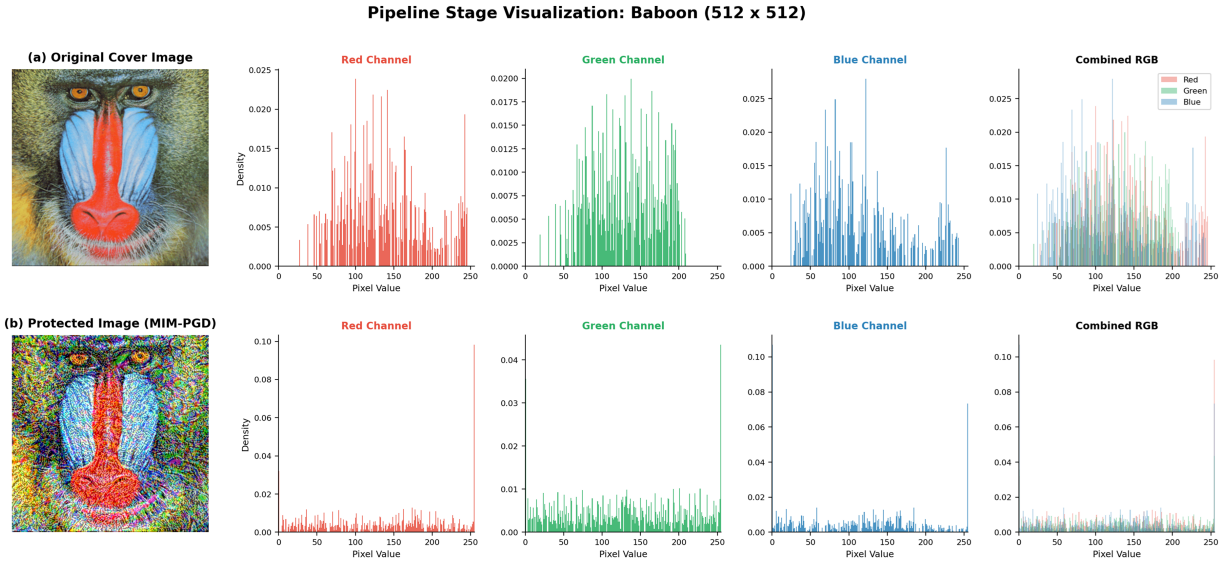


Figure 9: (Continued)

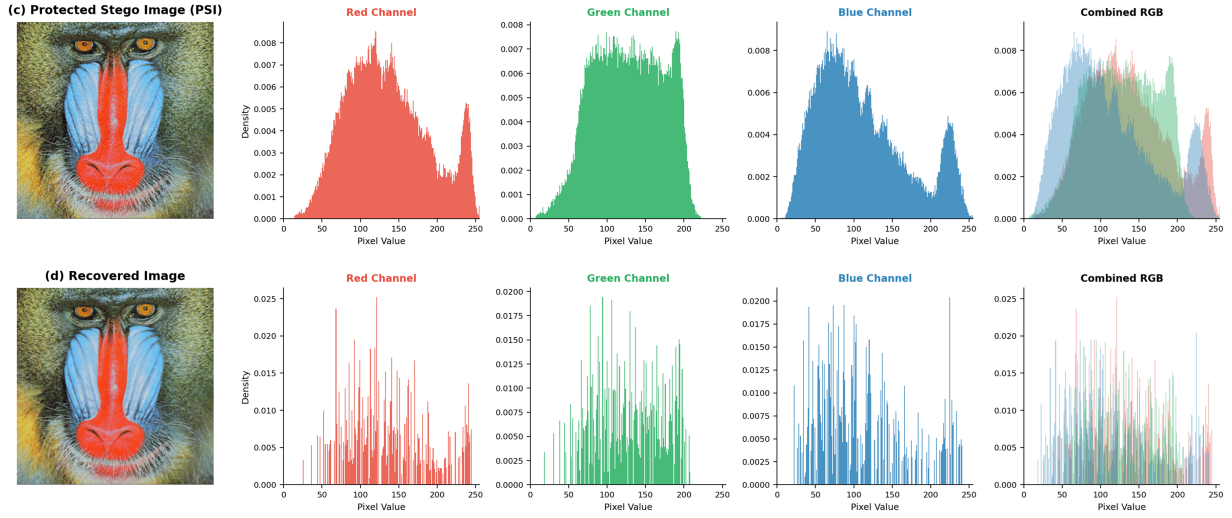


Figure 9: Per-channel RGB histogram distributions at four pipeline stages for the Baboon image (512 × 512): (**a**) Original cover, (**b**) Protected image after MIM-PGD perturbation, (**c**) Protected stego image after DCT-QIM embedding, and (**d**) Recovered image after extraction and SEK based restoration. Each row shows the image alongside its Red, Green, Blue, and combined RGB histograms.

Row (c) shows the PSI after DCT-QIM steganographic embedding. The histograms return to smooth distributions that closely resemble the original cover, because the embedding operates in the DCT frequency domain rather than the spatial pixel domain. The slight broadening relative to the cover reflects the combined effect of adversarial perturbation and QIM coefficient adjustment. Row (d) displays the recovered image after extraction and SEK based restoration. The per-channel histograms align closely with the cover image, with a mean absolute pixel difference of only 1.32 across all channels. This confirms that the recovery process successfully removes the embedding artifacts and restores the image to a state visually indistinguishable from the original cover.

6 Conclusion

This paper presented a dual layer framework for secure image sharing that combines GAN-based MIM-PGD adversarial poisoning with DCT-QIM steganographic embedding, coordinated through a Secret Embedding Key. The framework addresses three objectives that are not jointly addressed by the methods reviewed in Section 2: forcing AI classifiers to misclassify an image, hiding sensitive metadata inside it, and allowing the original image to be recovered with near-lossless fidelity (RA ≥ 0.998) for verification. The overall pipeline, illustrated in Fig. 2, operates through a sequence of segmentation, perturbation, and embedding steps that are each governed by formal mathematical constraints defined in Eqs. (1)–(12). Unlike existing proactive image protection methods such as Dual Defense [49] and EditGuard [50], the proposed framework adds reversible image recovery as a third capability, addressing a requirement that digital rights verification and image authentication identify as essential [2].

The adversarial module produces perturbations bounded by $\|\boldsymbol{\delta}\|_\infty \leq \epsilon$ that cause misclassification in a range of classifiers. On 200 ImageNet validation images at $\epsilon = 16/255$, the mean DSR across five architectures is 86.1% for the PSI, as reported in Table 9. VGG-16 reaches 96.5%, ResNet-50 reaches 93.5%, and InceptionV3 reaches 92.0%. ViT-B/16 shows lower DSR at 82.5%, which is consistent with how global patch attention dilutes the effect of localized pixel perturbations [19]. The Google Cloud Vision API has the lowest DSR at 66.0%. This commercial endpoint applies internal preprocessing, ensemble prediction, and robustification

that attenuate the gradient aligned noise before inference. The gap between academic classifiers and a commercial API is one of the practically relevant findings of this study for privacy-preserving deployments.

The embedding module uses perturbation-affected DCT regions in the YCbCr luminance channel to hide data, as described in Section 3.4 and illustrated in Fig. 5. On seven USC-SIPI images, all successful configurations achieve PSNR above 33 dB and SSIM above 0.90, as shown in Table 7. RA exceeds 0.998 across all trials, meaning the original image can be reconstructed from the PSI without perceptible loss, as long as the SEK is available and no lossy processing has been applied (Table 8). Comparing PI and PSI columns in Table 9 shows that embedding reduces adversarial performance by less than one percentage point. The two protection layers operate without mutual degradation, as the DSR drop is below one percentage point.

The recommended operating point is BPP = 0.10, which gives PSNR = 44.28 dB and SSIM = 0.972 on ImageNet-100, and PSNR = 35.93 dB and SSIM = 0.952 on USC-SIPI, with RA above 0.998 and mean DSR of 86.1%. As shown in Table 10, the two proposed-method rows together cover all four capability dimensions (adversarial defense, payload embedding, original-cover recovery, and imperceptibility), whereas no single competing method covers more than two of these dimensions. Compared to adversarial only methods like Shan et al. [22] and Xiao et al. [38], this framework also embeds data. Compared to RDH methods [35], it also resists AI classifiers. Compared to reversible adversarial methods by Yin et al. [63] (who recover the secret image but not the cover) and Lyu et al. [64] (who provide reversible perturbation without steganographic embedding), this framework additionally embeds user-defined payload and restores the original cover. Compared to dual-protection traceable watermarking [24], it reports near-lossless reversibility (RA $\geq$ 0.998). The cost is PSNR: the adversarial perturbation reduces quality by several dB relative to methods that only embed data without perturbation.

Five directions for future research follow from these results. First, improving DSR against commercial classifiers like the Google Cloud Vision API through query-based or transfer-optimized black box attacks [19,21]. Second, exploring an optional robust watermarking layer alongside the current fragile integrity fingerprint: the JPEG experiments in Section 5.7 confirm that the steganographic embedding functions as a tamper-detection seal by design, but applications such as social media sharing may benefit from a secondary robust channel that survives lossy compression [35,62]. Third, extending the SEK-based embedding to work on encrypted images [61] for additional confidentiality in secure cloud communications. Fourth, testing against adversarially robust classifiers trained with PGD augmentation [18] or certified defenses [16], where DSR will likely be lower. Fifth, evaluating the framework on larger datasets and higher-resolution images to assess scalability of the embedding capacity bounds established in Eq. (10) and to validate the secure sharing and recovery scenarios described in Section 5.9.

Acknowledgement: The authors express their sincere gratitude to all members of the Cyber Security Research Group, Net-Centric Computing (NCC) Laboratory, Department of Informatics, ITS, for their continuous support and insightful discussions. The authors declare that AI is used to improve the writing format and grammar, and to generate figures.

Funding Statement: This research is funded by Institut Teknologi Sepuluh Nopember (ITS) and managed under the Strategic Research Grant (SRG) Type D Scheme (Contract No. 1665/PKS/ITS/2026).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Hartawan Bahari Mulyadi and Tohari Ahmad; methodology, Hartawan Bahari Mulyadi and Tohari Ahmad; software, Hartawan Bahari Mulyadi; validation, Hartawan Bahari Mulyadi and Tohari Ahmad; formal analysis, Hartawan Bahari Mulyadi; investigation, Hartawan Bahari Mulyadi; resources, Tohari Ahmad; data curation, Hartawan Bahari Mulyadi; writing—original draft preparation, Hartawan Bahari Mulyadi; writing—review and editing, Tohari Ahmad; visualization,

Hartawan Bahari Mulyadi; supervision, Tohari Ahmad; project administration, Tohari Ahmad. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets supporting the findings of this study are openly available in public repositories. The USC-SIPI Image Database is available at https://sipi.usc.edu/database/, CIFAR-100 is available at https://www.cs.toronto.edu/~kriz/cifar.html, and ImageNet is available at https://www.image-net.org/.

Ethics Approval: Not applicable.

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

References

1. European Parliamentary Research Service. Enforcing the digital services act: state of play. Brussels, Belgium: European Parliament; 2024. PE 766.256. [cited 2026 Sep 20]. Available from: https://www.europarl.europa.eu/thinktank/en/document/EPRS_ATA(2024)766256.
2. National Institute of Standards and Technology. Digital investigation techniques: a NIST scientific foundation review. Gaithersburg, MD, USA: NIST; 2022. Report No.: NIST IR 8354.
3. Lam HPN. The impact of generative AI on creative industries: opportunities and challenges. In: Proceedings of the International Conference on Culture and Humanity in the Era of Rising and Integration (ICDHV 2025); 2025 Dec 19–20; Ho Chi Minh City, Vietnam. Paris, France: Atlantis Press; 2026. p. 557–76.
4. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 770–8.
5. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, et al. An image is worth 16 × 16 words: transformers for image recognition at scale. In: Proceedings of the International Conference on Learning Representations (ICLR); 2021 May 3–7; Virtual.
6. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–52. doi:10.1007/s11263-015-0816-y.
7. Krizhevsky A. Learning multiple layers of features from tiny images. Toronto, ON, Canada: University of Toronto; 2009 [cited 2026 Sep 20]. Available from: https://cave.cs.toronto.edu/kriz/learning-features-2009-TR.pdf.
8. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial nets. In: Proceedings of the 28th International Conference on Neural Information Processing Systems; 2014 Dec 8–13; Montreal, QC, Canada. p. 2672–80.
9. hao Y, Ma Y, Ge Y, Zhao L, Zhao L, Wang Q. When Deepfake meets backdoor: leveraging GAN fingerprints for data poisoning attack. IEEE Trans Dependable Secur Comput. 2026;23(2):3161–75. doi:10.1109/tdsc.2025.3633381.
10. Wang Y, Fu Y, Kong W, Liu C, Su S. PoisonPatch: natural adversarial patches via diffusion models and federated learning poisoning. IEEE Trans Inf Forensics Secur. 2025;20:11576–90. doi:10.1109/tifs.2025.3620155.
11. Shan S, Cryan J, Wenger E, Zheng H, Hanocka R, Zhao BY. Glaze: protecting artists from style mimicry by text-to-image models. In: 32nd USENIX Security Symposium (USENIX Security 23). Anaheim, CA, USA: USENIX Association; 2023. p. 2187–204.
12. Li X, Chen L, Fu T, Fu Z, Gao Y. Coverless image steganography based on semantic-controlled text-to-image generation. IEEE Trans Circ Syst Video Technol. 2025;35(8):8391–405. doi:10.1109/tcsvt.2025.3545067.
13. De Rosa V, Guillaro F, Poggi G, Cozzolino D, Verdoliva L. Exploring the adversarial robustness of CLIP for AI-generated image detection. In: Proceedings of the 2024 IEEE International Workshop on Information Forensics and Security (WIFS); 2024 Dec 2–5; Rome, Italy. p. 1–6.
14. Mehrjardi FZ, Latif AM, Zarchi MS, Sheikhpour R. A survey on deep learning-based image forgery detection. Pattern Recognit. 2023;144:109778. doi:10.1016/j.patcog.2023.109778.
15. Castillo Camacho I, Wang K. A comprehensive review of deep-learning-based methods for image Forensics. J Imaging. 2021;7(4):69. doi:10.3390/jimaging7040069.
16. Biggio B, Roli F. Wild patterns: ten years after the rise of adversarial machine learning. Pattern Recognit. 2018;84:317–31. doi:10.1016/j.patcog.2018.07.023.

17. Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: Proceedings of the International Conference on Learning Representations (ICLR); 2015 May 7–9; San Diego, CA, USA.
18. Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. In: Proceedings of the International Conference on Learning Representations (ICLR); 2018 Apr 30–May 3; Vancouver, BC, Canada.
19. Costa JC, Roxo T, Proença H, Inácio PRM. How deep learning sees the world: a survey on adversarial attacks & defenses. IEEE Access. 2024;12:61113–36. doi:10.1109/access.2024.3395118.
20. Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP); 2017 May 22–26; San Jose, CA, USA. p. 39–57.
21. Tramèr F, Kurakin A, Papernot N, Goodfellow I, Boneh D, McDaniel P. Ensemble adversarial training: attacks and defenses. arXiv:1705.07204. 2017.
22. Shan S, Wenger E, Zhang J, Li H, Zheng H, Zhao BY. Fawkes: protecting privacy against unauthorized deep learning models. In: 29th USENIX Security Symposium (USENIX Security 20). Anaheim, CA, USA: USENIX Association; 2020. p. 1589–604.
23. Shan S, Ding W, Passananti J, Wu S, Zheng H, Zhao BY. Nightshade: prompt-specific poisoning attacks on text-to-image generative models. In: Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP); 2024 May 19–23; San Francisco, CA, USA. p. 807–25.
24. Li M, Yang Z, Wang T, Zhang Y, Wen W. Dual protection for image privacy and copyright via Traceable adversarial examples. IEEE Trans Circ Syst Video Technol. 2024;34(12):13401–12. doi:10.1109/tcsvt.2024.3448351.
25. Ahmed N, Natarajan T, Rao KR. Discrete Cosine transform. IEEE Trans Comput. 1974;C-23(1):90–3. doi:10.1007/978-0-585-32313-8_5.
26. Boenisch F. A systematic review on model watermarking for neural networks. Front Big Data. 2021;4:729663. doi:10.3389/fdata.2021.729663.
27. Xie X, Jiang J, Zhang J, Chen K, Zhang W, Yu N. Reversible adversarial visible image watermarking. Signal Process. 2025;234:109999. doi:10.1016/j.sigpro.2025.109999.
28. Zhu J, Kaplan R, Johnson J, Fei-Fei L. HiDDeN: hiding data with deep networks. In: Proceedings of the European Conference on Computer Vision (ECCV); 2018 Sep 8–14; Munich, Germany. p. 682–97.
29. Chan CK, Cheng LM. Hiding data in images by simple LSB substitution. Pattern Recognit. 2004;37(3):469–74. doi:10.1016/j.patcog.2003.08.007.
30. Fridrich J, Kodovsky J. Rich models for steganalysis of digital images. IEEE Trans Inf Forensics Secur. 2012;7(3):868–82. doi:10.1109/tifs.2012.2190402.
31. De La Croix NJ, Ahmad T, Ijtihadie RM. Pixel-block-based steganalysis method for hidden data location in digital images. Int J Intell Eng Syst. 2023;16(6):375–85. doi:10.22266/ijies2023.1231.31.
32. Chinniyan K, Samiyappan TV, Gopu A, Ramasamy N. Image steganography using deep neural networks. Intell Autom Soft Comput. 2022;34(3):1877–91. doi:10.32604/iasc.2022.027274.
33. Shi YQ, Li X, Zhang X, Wu HT, Ma B. Reversible data hiding: advances in the past two decades. IEEE Access. 2016;4:3210–37. doi:10.1109/access.2016.2573308.
34. Aiswarya S, Gomathi R. Secured medical data transfer using reverse data hiding system through steganography. Intell Autom Soft Comput. 2022;34(2):969–82. doi:10.32604/iasc.2022.025475.
35. Gao G, Wang M, Wu B. Efficient robust reversible watermarking based on ZMs and integer wavelet transform. IEEE Trans Ind Inform. 2024;20(3):4115–23. doi:10.1109/tii.2023.3321101.
36. Ren Z, Guo W, Lin C, Jin H, Qu H. Image steganography with dual strategies: defense and attack. IEEE Signal Process Lett. 2025;32:2594–8. doi:10.1109/lsp.2025.3582673.
37. Xiao A, Li Z, Wang G, Zheng L, Sun H. SRIS-net: a robust image steganography algorithm based on feature score maps. Front Inf Technol Electron Eng. 2025;26(6):930–45. doi:10.1631/fitee.2400069.
38. Xiao C, Li B, Zhu J, He W, Liu M, Song D. Generating adversarial examples with adversarial networks. In: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI); 2018 Jul 13–19; Stockholm, Sweden. p. 3905–11.

39. Sharma VK, Mir RN, Rout RK. Towards secured image steganography based on content-adaptive adversarial perturbation. Comput Electr Eng. 2023;105:108484. doi:10.1016/j.compeleceng.2022.108484.
40. Kapusta K, Mattioli L, Addad B, Lansari M. Protecting ownership rights of ML models using watermarking in the light of adversarial attacks. AI Ethics. 2024;4(1):95–103. doi:10.1007/s43681-023-00412-3.
41. Tyagi T, Singh KN, Singh AK, Gupta BB. IPNetTool: watermarking and chaos for copyright protection of image classification models. Future Gener Comput Syst. 2025;173:107907. doi:10.1016/j.future.2025.107907.
42. Zhang Q, Huang F. Provably secure generative steganography based on adjustable orthogonal mapping. IEEE Trans Dependable Secur Comput. 2026;23(1):34–48. doi:10.1109/tdsc.2025.3604222.
43. Fan Z, Chen K, Wang Y, Zhang W, Yu N. GIANT: generated image adversarial steganography based on narrowed targeting. IEEE Trans Circ Syst Video Technol. 2026;36(4):5671–82. doi:10.1109/tcsvt.2025.3629825.
44. Xiao T, Wang J, Li S. Heterogeneous domain remapping for universal detection of generative linguistic steganography. IEEE Signal Process Lett. 2025;32:1281–5. doi:10.1109/lsp.2025.3549015.
45. Thodi DM, Rodriguez JJ. Expansion embedding techniques for reversible watermarking. IEEE Trans Image Process. 2007;16(3):721–30. doi:10.1109/tip.2006.891046.
46. Ni Z, Shi YQ, Ansari N, Su W. Reversible data hiding. IEEE Trans Circ Syst Video Technol. 2006;16(3):354–62. doi:10.1109/tcsvt.2006.869964.
47. Tang Y, Wang S, Wang C, Xiang S, Cheung YM. A highly robust reversible watermarking scheme using embedding optimization and rounded error compensation. IEEE Trans Circ Syst Video Technol. 2023;33(4):1593–609. doi:10.1109/tcsvt.2022.3216849.
48. Wu L, Cheng H, Yan W, Chen F, Wang M, Wang T. Reversible and colorable deep image steganography with large capacity. J Electron Imaging. 2023;32(4):043006. doi:10.1117/1.jei.32.4.043006.
49. Zhang Y, Ye D, Xie C, Tang L, Liao X, Liu Z, et al. Dual defense: adversarial, traceable, and invisible robust watermarking against face swapping. IEEE Trans Inf Forensics Secur. 2024;19:4628–41. doi:10.1109/tifs.2024.3383648.
50. Zhang X, Li R, Yu J, Xu Y, Li W, Zhang J. EditGuard: versatile image watermarking for tamper localization and copyright protection. In: Proceedings of the 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024 Jun 16–22; Seattle, WA, USA. p. 11964–74.
51. Dong Y, Liao F, Pang T, Su H, Zhu J, Hu X, et al. Boosting adversarial attacks with momentum. In: Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition; 2018 Jun 18–23; Salt Lake City, UT, USA. p. 9185–93.
52. Dur-E-Jabeen M, Khan T, Iftikhar R, Siddique AA, Asghar S. An algorithm to reduce compression ratio in multimedia applications. Comput Mater Contin. 2023;74(1):539–57. doi:10.32604/cmc.2023.032393.
53. Chen B, Wornell GW. Quantization index modulation: a class of provably good methods for digital watermarking and information embedding. IEEE Trans Inf Theory. 2001;47(4):1423–43. doi:10.1109/isit.2000.866336.
54. Setiadi DRIM. PSNR vs. SSIM: imperceptibility quality assessment for image steganography. Multimed Tools Appl. 2021;80(6):8423–44. doi:10.1007/s11042-020-10035-z.
55. Simões W, de Sá M. PSNR and SSIM: evaluation of the imperceptibility quality of images transmitted over wireless networks. Procedia Comput Sci. 2024;251:463–70. doi:10.1016/j.procs.2024.11.134.
56. Tian J. Reversible data embedding using a difference expansion. IEEE Trans Circ Syst Video Technol. 2003;13(8):890–6. doi:10.1109/tcsvt.2003.815962.
57. Weber AG. The USC-SIPI image database: version 6. Los Angeles, CA, USA: USC Signal and Image Processing Institute; 2018. USC-SIPI-432. [cited 2026 Sep 20]. Available from: https://sipi.usc.edu/database/SIPI_Database.pdf.
58. Moosavi-Dezfooli SM, Fawzi A, Frossard P. DeepFool: a simple and accurate method to fool deep neural networks. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016 Jun 27–30; Las Vegas, NV, USA. p. 2574–82.
59. Lin CY, Chen FJ, Ng HF, Lin WY. Invisible adversarial attacks on deep learning-based face recognition models. IEEE Access. 2023;11:51567–77. doi:10.1109/access.2023.3279488.

60. Zheng Z, Hu Y, Bin Y, Xu X, Yang Y, Shen HT. Composition-aware image steganography through adversarial self-generated supervision. IEEE Trans Neural Netw Learn Syst. 2023;34(11):9451–65. doi:10.1109/tnnls.2022.3175627.
61. Fan M, Zhong S, Xiong X. Reversible data hiding method for interpolated images based on modulo operation and prediction-error expansion. IEEE Access. 2023;11:27290–302. doi:10.1109/access.2023.3258461.
62. Wang C, Zhang Q, Wang X, Zhou L, Li Q, Xia Z, et al. Light-field image multiple reversible robust watermarking against geometric attacks. IEEE Trans Dependable Secur Comput. 2025;22(6):5861–75. doi:10.1109/tdsc.2025.3576223.
63. Yin Z, Chen L, Lyu W, Luo B. Reversible attack based on adversarial perturbation and reversible data hiding in YUV colorspace. Pattern Recognit Lett. 2023;166:1–7. doi:10.1016/j.patrec.2022.12.018.
64. Lyu W, Sun X, Yin Z. Reversible adversarial attack based on pixel smoothing in HSV colorspace. In: Proceedings of the 9th International Conference on Intelligent Information Technology (ICIIT); 2024 Feb 23–25; Ho Chi Minh City, Vietnam. p. 55–61.
65. Panchikkil S, Manikandan VM. A prediction error based reversible data hiding scheme in encrypted image using block marking and cover image pre-processing. Multimed Tools Appl. 2024;83(2):4993–5030. doi:10.1007/s11042-023-15319-8.
66. Aljughaiman A, Alrawashdeh R. Content-adaptive LSB steganography with saliency fusion, ACO dispersion, and hybrid encryption with ablation study. Sci Rep. 2025;16(1):3829. doi:10.1038/s41598-025-33920-9.
67. Liu L, Tong S, Xue Q. Reversible image steganography based on residual structure and attention mechanism. Sci Rep. 2025;15(1):19355. doi:10.1038/s41598-025-04441-2.
68. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF, editors. Medical image computing and computer-assisted intervention—MICCAI 2015. Cham, Switzerland: Springer International Publishing; 2015. p. 234–41.
69. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process. 2004;13(4):600–12. doi:10.1109/tip.2003.819861.
70. Wallace GK. The JPEG still picture compression standard. IEEE Trans Consum Electron. 1992;38(1):30–44. doi:10.1145/103085.103089.