GENSTEGOBENCH: AUDITING PERSISTENT COVERT CHANNELS IN LATENT GENERATIVE IMAGE SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern generative image systems are increasingly built on diffusion or flow-style iterative generation, where images can often be approximately inverted to latent or noise representations. This generation–recovery property creates a security concern: an ordinary-looking generated image may carry a covert message that remains decodable after common online processing. We introduce *GenStegoBench*, a security audit benchmark for measuring when generative-image steganographic channels persist, when they fail, and which transformations can disrupt them. Our audit provides both attacker- and defender-side insights: generator backbones and solvers transport decoder margins and can strengthen hidden-channel reliability, while common image transformations break different recoverable invariants. GenStegoBench audits covert-channel persistence through the generation–channel–recovery lifecycle of generative images.

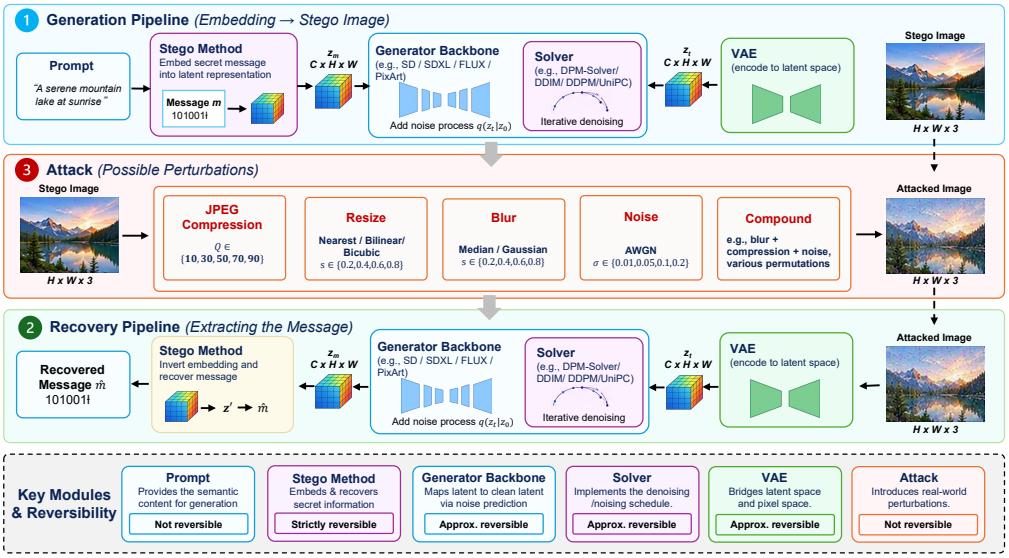


Figure 1: Image-based generative steganography.

Keywords: Generative Image Models, Steganography, Robustness, Benchmark

1 INTRODUCTION

Modern generative image systems are largely built on diffusion, diffusion-transformer, or flow-style iterative generation (Ho et al., 2020; Song et al., 2020; Rombach et al., 2022; Ramesh et al., 2022; Podell et al., 2023; Peebles & Xie, 2023; Chen et al., 2023; Esser et al., 2024). They transform noisy latents into realistic images through structured trajectories, and often admit approximate recovery paths that map a generated image back to latent representation. This generation–recovery property supports editing and reconstruction, but also creates a substrate for covert communication.

Generated images can therefore serve as persistent covert-channel carriers. A sender may encode a secret in the latent or trajectory state, synthesize a visually ordinary image, transmit it through public multimedia channels, and let a receiver recover the message through inversion and decoding. Unlike traditional steganography, which modifies an existing cover image, generative image steganography embeds information before or during synthesis (Hu et al., 2024; Yang et al., 2024b; Zhou et al., 2023; 2025; Ding et al., 2023); the carrier is a generated image whose latent structure may preserve hidden information. This is especially relevant for social media, where AI-generated images can be reposted and redistributed while remaining visually plausible and difficult to distinguish from ordinary generated content.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

The central security question is: *when can a covert message persist through the lifecycle of a generated image?* Images are rarely shared as pristine model outputs: they may be serialized, resized, cropped, enhanced, and re-uploaded by platforms or users. Such natural transformations can disrupt the structures needed for message recovery; conversely, survival through them would indicate an unintended covert channel in everyday multimedia processing.

We audit this problem from both attacker and defender perspectives. On the attacker side, the generator backbone, solver, and inversion path determine which message-bearing structures remain recoverable in latent or image space. On the defender side, the same analysis identifies common platform operations that can corrupt unauthorized covert channels while preserving visible semantics, without requiring white-box access to every possible decoder.

Existing evaluations provide only partial insight. Prior work usually reports extraction accuracy for individual methods under limited settings (Hu et al., 2024; Yang et al., 2024b; Zhou et al., 2023; 2025); it often treats the generator as an implementation detail, overlooks method-specific invariants and payload-dependent failure modes, and relies on single-step perturbations that do not capture ordered multimedia lifecycles.

In this paper, we introduce *GenStegoBench*, a benchmark for auditing persistent covert channels in generative image systems. GenStegoBench evaluates representative methods under unified settings spanning generator backbones, solvers, inversion regimes, single-step transformations, ordered compound channels, payload levels, datasets, and prompts. Beyond comparing methods, it diagnoses why hidden channels survive, when they fail, and which transformations can disrupt them.

GenStegoBench is organized around three main questions:

(1) How do model-side factors determine persistence boundaries? We evaluate how generator backbones, solvers, and recovery paths transport decoder margins, induce empirically stable recovery directions, and affect message persistence.

(2) How to design a generator-compatible embedding geometry? We model method-specific decoder margins across spatial, frequency-domain, projection-based, distribution-bin, and order-based carriers. The analysis yields simple generator-compatible modifications that substantially improve reliability and clarifies how payload changes the embedding geometry.

(3) What kind of perturbations can corrupt covert communication channels? We evaluate common content-preserving transformations, including rescaling, JPEG compression, blurring, noise, cropping, and feature enhancement, and further study ordered compound pipelines that model social-media-like image lifecycles.

Main contributions are summarized as follows:

1. We formulate generative image steganography robustness as a security problem of *covert-channel persistence* through the generation–channel–recovery lifecycle, where success depends on preserving message-bearing invariants after realistic multimedia transformations.
2. We introduce *GenStegoBench*, a unified audit benchmark that evaluates persistent covert channels across steganographic methods, generator backbones, solver and inversion regimes, payload levels, datasets, and benign perturbations introduced by common multimedia channels. It also reports visual-quality and detectability guardrails to verify whether covert-channel persistence occurs without obvious perceptual degradation or classical steganalysis failures.
3. We provide a mechanism-level analysis showing that generator-dependent recovery geometry and payload-induced decoder-margin compression govern attacker-side reliability, while common image processing operations disrupt different latent, spatial, positional, and frequency-domain invariants from the defender side.

2 Background

This section introduces the concepts used throughout GenStegoBench. We first describe the unified generation–recovery loop, then summarize the latent-vector steganographic methods we evaluate, and finally define the recoverable invariants that guide our analysis.

2.1 Image Generation as a Recovery Loop

Modern image generators, including GANs (Radford et al., 2016; Karras et al., 2019; Kang et al., 2023), flow-based models (Dinh et al., 2015; 2017; Kingma & Dhariwal, 2018), and diffusion or diffusion-transformer models (Ho et al., 2020; Song et al., 2020; Rombach et al., 2022; Ramesh et al., 2022; Podell et al., 2023; Peebles & Xie, 2023; Chen et al., 2023; Esser et al., 2024), share

a common interface: image synthesis begins from a latent variable z, often sampled from a simple prior such as a standard Gaussian. The generator maps z to an image through model-specific transformations, such as a transformer backbone for noise prediction, diffusion or flow solvers for denoising, and a VAE that maps latent vectors to the image space.

For hidden communication, the important object is not only the forward generator G, but the full recovery regime used by the receiver. We write this regime as P, which includes the generator family, solver and inversion procedure, and other method-specific trajectory interfaces. The corresponding forward and recovery maps are denoted by G_P and R_P. A hidden channel through an image transformation T shown in Fig. 1 can be written as

$$m \xrightarrow{E} z_m \xrightarrow{G_P} x \xrightarrow{T} \tilde{x} \xrightarrow{R_P} \hat{z} \xrightarrow{D} \hat{m}.$$

The channel persists when $\hat{m}$ remains equal or close to m under the intended decoding metric.

2.2 Steganography with Latent Vectors

The methods considered in GenStegoBench share the generation–recovery schema above, but instantiate different message-bearing structures. These differences matter because each method defines a recoverable invariant–signs, ranks, bins, projections, frequencies, or stable positions–that must survive image generation, channel transformation, and recovery.

GenStegoBench covers seven latent-vector methods. BaseGen encodes one bit per latent coordinate through its sign, serving as a minimal sign-preservation baseline. S2IRT (Zhou et al., 2023) conveys messages through secret-guided grouping and positional rearrangement. PARIS (Yang et al., 2024b) maps ciphertext into Gaussian latent coordinates by an invertible CDF transform and decodes Gaussian-CDF bins. GRDH (Hu et al., 2024) uses keyed orthogonal projection and shuffling, with recovery depending on projection and rounding cells. GSD (Zhou et al., 2025) embeds in latent-frequency coefficients rather than spatial coordinates. MDDM (Xu et al., 2025) uses sign-separated truncated Gaussian sampling on selected diffusion-noise coordinates. LDStega (Peng et al., 2024) selects latent positions stable under a latent-to-image-to-latent round trip and modulates them relative to the denoising mean. Together, these methods cover sign, order, bin, projection, frequency, diffusion-coordinate, and stable-position carriers.

2.3 Recoverable Invariants and Failure Boundaries

We use *recoverable invariant* to denote the structure that carries the message and must remain decodable after the image channel. Examples include spatial coordinate signs, frequency coefficient signs, and Gaussian-CDF bins. Let $I_D(z)$ denote the decoder-relevant invariant induced by a method D. Persistence requires

$$I_D(R_P(T(G_P(z_m)))) = I_D(z_m),$$

or an approximate version of this equality under the method's extraction metric.

3 Perturbations and Threat Model

Generative image steganography can be viewed as covert communication through synthetic visual media. Public multimedia channels are rarely distortion-free: storage, resizing, compression, filtering, and user edits alter the signal used for decoding. GenStegoBench models these changes as post-generation transformations, including *single-step perturbations* that isolate one transformation family and *compound perturbations* that capture multi-stage image-processing pipelines.

3.1 Threat Model

We consider covert communication enabled by a public generative image system. A sender and receiver agree on a generative steganographic method and use a public or reproducible generator. The sender encodes a secret message m into a generator-side state, synthesizes a plausible image x, and transmits it through a public multimedia channel. The receiver then recovers the hidden message from the transformed image $\tilde{x}$.

Formally, we write the complete hidden communication lifecycle as

$$m \xrightarrow{E} z_m \xrightarrow{G_P} x \xrightarrow{\mathcal{T}} \tilde{x} \xrightarrow{R_P} \hat{z} \xrightarrow{D} \hat{m},$$

where E is the embedding procedure, G_P is the generator under backbone/solver regime P, $\mathcal{T}$ is the public image channel, R_P is the recovery procedure, and D is the decoder. The attacker-side objective is $\hat{m} = m$, or high extraction accuracy under the method's decoding metric, while the carrier remains visually plausible.

The channel $\mathcal{T}$ represents non-adaptive, content-preserving transformations that commonly occur during image storage, transmission, redistribution, and platform-side processing. For a single channel operation, the received image is

$$\tilde{x} = \mathcal{T}_s(x),$$

where $\mathcal{T}$ denotes a transformation family and s its severity. Deployment-like lifecycles use ordered compound channels $\tilde{x} = \mathcal{T}_{s_k}^{(k)} \circ \cdots \circ \mathcal{T}_{s_1}^{(1)}(x)$.

This threat model induces two complementary audit goals. From the attacker-side perspective, we ask which generator and embedding factors make covert channels more persistent. From the defender-side perspective, we ask which image-processing operations can reduce covert-channel reliability without destroying the visible image.

We make two assumptions. First, the receiver does not have access to the original clean image x during decoding. Second, the channel transformations are agnostic to the underlying embedding method and are not optimized against a particular decoder. White-box adaptive attacks specifically designed to defeat individual hidden communication mechanisms represent a complementary security problem and are beyond the scope of this study.

4 Benchmark Protocol and Settings

Scope. Unless otherwise stated, each run fixes the generator–recovery pair, method, prompt source, perturbation family, severity, and random seed, then reports extraction accuracy/BER plus quality and detectability guardrails.

Image generator. The primary backbones are Stable Diffusion v1.5/v2.1 (Rombach et al., 2022), SDXL (Podell et al., 2023), PixArt-α (Chen et al., 2023), Stable Diffusion 3.5 Medium (Esser et al., 2024), and FLUX.1-schnell (Black Forest Labs, 2024). All use 512×512 resolution except SDXL, which uses native 1024×1024 generation with a $4 \times 128 \times 128$ latent grid. Forced-512 SDXL, native-1024 PixArt-Sigma, and Glow are diagnostic only unless stated.

Recovery pairs. We report only implemented pairs and exclude mixed solver–inversion settings: DDIM/DDIM inversion; DPM-Solver++/matching reverse or Diffusers inverse; Euler/reverse-Euler; UniPC/aligned SD1.5 reverse multistep; and SD3.5/FLUX with flow-Euler recovery.

Steganographic methods. We evaluate seven representative methods illustrated in Section 2.2. For each reported comparison, we control the evaluated dimension by fixing the generator–recovery pair, prompt source, perturbation family and severity, payload setting when applicable, and random seed protocol. Method-specific internal parameters that define the original embedding geometry are kept in their canonical configurations: S2IRT uses auto-payload planning; GSD uses default binary DCT-sign coding; MDDM uses dense tail-separated sampling with its default adaptive threshold; LDStega uses $\gamma = 0.3$; other parameters follow repository defaults. These choices are not used to tune one method against another; they preserve each method's intended carrier while the benchmark varies the audited factor. Payload sweeps use the comparable multi-rate subset.

Perturbations. Single-step perturbations use a PNG image-I/O round trip as storage control; nearest-neighbor, bilinear, and bicubic rescaling at factors $\{0.2, 0.4, 0.6, 0.8, 1.0\}$; JPEG qualities $\{10, 30, 50, 70, 90\}$; blur kernels $\{3, 5, 7, 9\}$; Gaussian noise $\sigma \in \{0.01, 0.05, 0.1, 0.2\}$; centered-crop keep ratios $\{0.95, 0.9, 0.85, 0.75\}$ with zero-filled missing regions; contrast factors $\{1.2, 1.5, 1.8\}$; and sharpening strengths $\{1.0, 1.5, 2.0\}$; implementation details are provided in Appendix A. Compound tests pair JPEG quality 70 with, in both orders, nearest-neighbor rescaling at 0.8, bilinear rescaling at 0.6, Gaussian blur kernel 3, median blur kernel 5, crop ratio 0.85, or Gaussian noise $\sigma = 0.05$. Lifecycle extensions cover JPEG-only, resize-only, platform-block, human–platform-block, randomized platform/human/channel regimes, and JPEG–crop–resize order probes at depths $\{0, 1, 2, 4, 8\}$ or $\{0, 1, 2, 4, 8, 16, 32\}$.

Datasets. We use three prompt sources throughout the benchmark: *Flickr8k* (Hodosh et al., 2015), *MSCOCO* (Lin et al., 2014), and *Conceptual Captions (CC)* (Sharma et al., 2018). Unless otherwise stated, robustness and cross-dataset comparisons are evaluated on the corresponding prompt pools from all three datasets under the same protocol. Single-prompt visualizations and ten-prompt cached probes are used only as explicitly labeled mechanism diagnostics, not as the dataset scope of the benchmark.

5 Results

Following the threat model, we organize GenStegoBench as a two-sided AI security audit of the same covert-channel lifecycle. The attacker-side audit asks which generator and encoding choices make hidden messages persist through generation and recovery; the defender-side audit asks which legal, content-preserving image operations can disrupt unauthorized covert communication in ordinary multimedia pipelines.

Table 1: Boundary-crossing rate $C = \Pr[\gamma' < 0]$ for the single-prompt backbone probe, comparing spatial-sign and DCT-sign carriers across generator backbones.

Carrier	SD2.1	SDXL	SD3.5	FLUX	PixArt
BaseGen spatial sign	0.142	0.228	0.292	0.357	0.266
GSD DCT sign	0.031	0.157	0.074	0.221	0.174

Table 2: Within-backbone solver sensitivity $\Delta_S C$ over DDIM20, DPM-Solver++20, Euler20+inverse, and UniPC20+inverse.

Backbone	BaseGen	GSD	GRDH	S2IRT	Mean
SD1.5	0.012	0.006	0.007	0.017	0.011
SD2.1	0.038	0.032	0.036	0.055	0.040
SDXL-1024	0.044	0.061	0.057	0.061	0.056
PixArt-α-512	0.166	0.218	0.185	0.154	0.181

5.1 Attacker Side: Generator and Steganographic Factors in Covert Channel Persistence

The attacker-side audit asks how far a generative steganographic channel can be strengthened by choosing the generator regime and message geometry. We study the clean generator–solver lifecycle of three representative backbone families: UNet latent diffusion (SD1.5/SD2.1/SDXL), where a convolutional denoiser transports VAE latents through a diffusion trajectory; DiT-style token denoising (PixArt-α), where the latent grid is tokenized and mixed by transformer blocks; and multimodal flow-transformer denoising (SD3.5/FLUX), where image and text representations are jointly transported along a flow-style path (Rombach et al., 2022; Podell et al., 2023; Peebles & Xie, 2023; Chen et al., 2023; Esser et al., 2024; Black Forest Labs, 2024). These regimes expose different stable subspaces to the steganographic decoder, so we treat the backbone and solver as part of the covert channel rather than as interchangeable image-generation details.

Core Finding I: Frequency-domain carriers are generally more stable than spatial carriers, while spatial carriers become persistent when their message space aligns with empirically stable recovery directions.

A modern text-to-image generator is not an exactly invertible bijection from a latent vector to an image; it preserves the latent information useful for its learned generation trajectory and contracts, rotates, or mixes other directions. For a complete generator regime P and encoded latent z_m, the clean lifecycle is

$$F_P = R_P \circ G_P, \qquad z' = F_P(z_m), \qquad \delta = z' - z_m.$$

Here, G_P denotes generation, R_P denotes recovery, and F_P is their composition. We use **latent stable space** as an operational diagnostic for the directions whose decoder margins are empirically preserved by this composed map, rather than as a complete local-Jacobian characterization. A steganographic method defines its own **message space**; it tends to remain reliable when that message space overlaps the empirically preserved directions.

We use a **common margin representation** to compare carrier geometries. BaseGen defines spatial-domain constraints,

$$\gamma_i^{\mathrm{sp}}(z) = s_i z_i, \qquad s_i = 2m_i - 1,$$

while GSD defines frequency-domain constraints,

$$\gamma_i^{\mathrm{freq}}(z) = s_i[\mathrm{DCT}(z)]_i,$$

Frequency-domain embedding can make the carrier less exposed to raw coordinate displacement by aggregating latent variation in a spectral basis. Figure 2 shows that, in this diagnostic, DCT boundary crossings concentrate near low-frequency regions associated with visible semantic structure and then spread into backbone-specific low-to-mid-frequency bands; Fig. 3 shows the complementary spatial behavior, where BaseGen sign crossings follow prompt-level semantic contours under SD3.5. These observations are consistent with the lower GSD boundary-crossing rates in Table 1: crossings drop from 0.142 to 0.031 on SD2.1, 0.292 to 0.074 on SD3.5, 0.357 to 0.221 on FLUX, and 0.266 to 0.174 on PixArt-α. We therefore treat the stable-space view as a mechanism-oriented interpretation of the measured margin transport, not as a standalone proof of the underlying generator geometry.

Design implication and minimal modification for BaseGen. For spatial-domain embedding, the attacker should not allocate bits uniformly over raw latent coordinates. Guided by the stable-space analysis, we make two minimal BaseGen changes without altering the generator backbone. First, embedding only on the empirically stable 80% of semantically aligned coordinates improves clean accuracy from 70.93% to 76.37% on the original bonesaw prompt and from 72.40% to 76.95% over a 10-prompt diagnostic set. Second, a lightweight residual stego module trained on the clean transport raises cached-surrogate original-secret accuracy from 79.84% to 91.52% and random-secret accuracy from 84.03% to 93.48%. Under an end-to-end visual guardrail, reducing the residual amplitude to 0.05 keeps SSIM at .901 and still improves accuracy from 81.44% to 82.80%; larger amplitudes increase recovery but also increase visual drift. This exposes a direct reliability–quality trade-off.

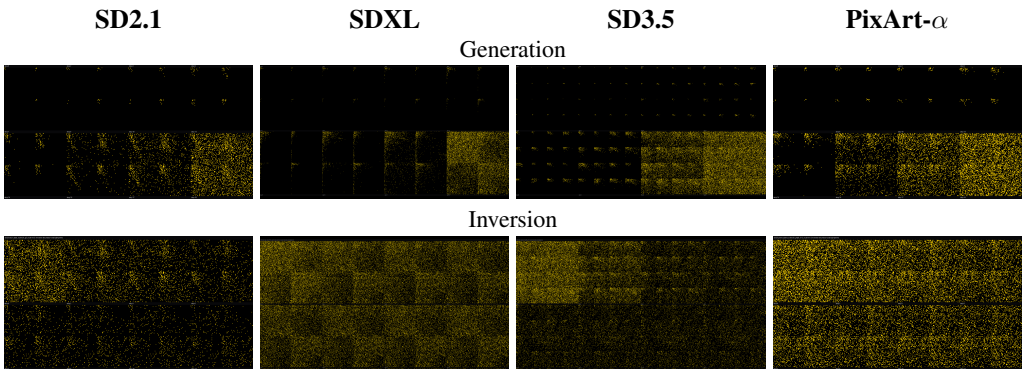
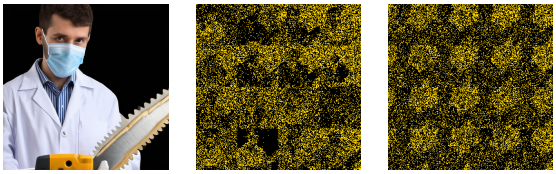


Figure 2: Stepwise GSD boundary-crossing lightpoints in the DCT domain. Columns compare backbones, and rows compare generation versus inversion. The upper-left region in each panel is DC/low frequency, and moving right or downward increases horizontal or vertical frequency. Across backbones, crossing events concentrate around the low-frequency semantic core, while the spread pattern is backbone-specific.

Table 3: Minimal BaseGen interventions guided by the stable-space analysis. Accuracy values are clean extraction accuracy (%) without any image transformation. Stable-support rows are cached clean-transport probes; other rows regenerate images and then invert them under an end-to-end visual guardrail.

Modification	Reference → Modified	Gain	Guardrail
Stable 80% (bonesaw)	70.93 → 76.37	+5.44	80% payload
Stable 80%	72.40 → 76.95	+4.55	80% payload
Module .35	79.84 → 91.52	+11.67	fixed 91.90
Module .35	84.03 → 93.48	+9.45	fixed 93.80
Module .05	81.44 → 82.80	+1.36	SSIM .901
Module .10	81.44 → 84.19	+2.75	SSIM .782
Module .15	81.44 → 84.07	+2.63	SSIM .681



Clean image Generation 70% Inversion 70%

Figure 3: Focused semantic alignment in BaseGen under SD3.5. The generated white-coat image is shown next to the 70% generation and inversion lightpoint snapshots with the latent space flattened into 16 channels. The boundary-crossing structure visually follows the prompt-level semantic silhouette.

Solver effects: discretized transport is secondary but measurable.

The backbone represents the learned transformation that maps latent states toward image-relevant representations, while the solver and inversion path determine the numerical trajectory through which the latent state is denoised and recovered.

We next isolate the solver axis. For a fixed backbone B, the denoiser gives a learned local field or prediction $f_B(z_t, t, c)$, where c is the text conditioning. A solver does not learn a new image prior; it chooses how this same prediction is integrated over the noise schedule. We can write one clean generation or recovery step abstractly as

$$z_{k-1} = \Phi_k^S(z_k;\ f_B, \alpha_k, \sigma_k, h_k, \mathcal{H}_k)\,,$$

where (α_k, σ_k) describe the schedule point, h_k is the step size, and $\mathcal{H}_k$ denotes any solver history. Thus the solver changes the discretized transport map,

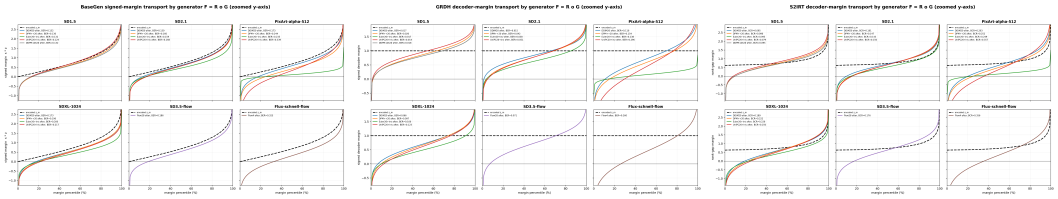
$$F_{B,S} = R_{B,S} \circ G_{B,S}, \qquad \Delta\gamma_j^S = a_j^\top (F_{B,S}(z_m) - z_m)\,,$$

but it still acts through the same backbone field. Our benchmark proposes a hierarchy: solver choice should perturb the margin transport around a backbone-specific stable space, while the backbone should determine the dominant direction and scale of the displacement.

Figure 4 and Table 2 support the predicted hierarchy. For the UNet-based SD family, the solver span is small: the mean $\Delta_S C$ is only 0.011 for SD1.5, 0.040 for SD2.1, and 0.056 for SDXL. This indicates that, once the backbone has fixed a characteristic recovered-margin pattern, changing the numerical path mainly produces a second-order displacement of the same margin distribution. PixArt-α is a useful exception: its tokenized DiT latent interface yields a larger solver span (0.181 on average), suggesting that solver discretization can amplify lower-tail margin transport when the backbone mixes message-bearing directions more aggressively. Thus, the solver should be reported as part of the generation–recovery pair and treated as a tunable component of covert-channel reliability.

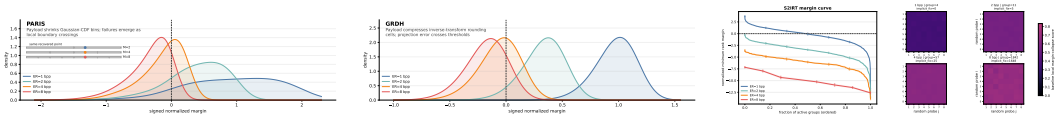
Payload stress reveals decoder geometry.

Under the same margin view used in Sec. 5.1, payload increase can be considered a decoder-side move on the recovered-margin distribution. For each payload-dependent constraint j, the signed



(a) BaseGen spatial sign (b) GRDH projection sign (c) S2IRT rank gap

Figure 4: Zoomed solver-controlled recovered-margin curves grouped by carrier geometry. The horizontal zero line is the crossing boundary. The SD-family panels show nearly overlapping solver curves near zero. Flow-style backbones are shown as native single-solver references rather than solver ablations.



(a) PARIS (b) GRDH (c) S2IRT

Figure 5: Decision-margin behavior under payload stress. PARIS: payload shrinks Gaussian-CDF bins, shifting the signed boundary-margin distribution toward zero. GRDH: payload compresses projection-rounding cells, causing more threshold-crossing events. S2IRT: payload reduces active rank gaps, pushing the ordered minimum-rank-margin curves downward while globally brightening local-collapse heatmaps.

margin $\gamma_j(z; ER_l)$ measures the distance from the recovered variable to its correct decoding region. Increasing ER_l narrows bins, compresses projection cells, or reduces rank gaps. Thus payload shifts the initial margins toward the boundary, making the same perturbation $\Delta\gamma_j$ more likely to satisfy the boundary-crossing condition $\gamma_j(z_m; ER_l) + \Delta\gamma_j < 0$.

Fig. 5 illustrates this common mechanism for three representative decoder geometries: PARIS as distribution-bin decoding, GRDH as projection-based decoding, and S2IRT as order-based decoding. Their concrete failure patterns differ, but all can be read as payload-induced margin push and displacement toward decoder boundaries. We therefore use this figure as the main-text evidence and defer detailed payload trade-off numbers to Appendix C.1.

5.2 Defender Side: Benign Processing Against Covert Channels

GenStegoBench uses single-step perturbations to probe corruption mechanisms and compound perturbations to simulate lifecycle risk when generated images become carriers for covert communication. Combining both views, we report our second core finding:

Core Finding II: Sample-selection transformations are strong single-step disruptors, but lifecycle-level anti-stego gains saturate as depth increases.

The defender-side audit asks whether ordinary, legal, content-preserving operations can suppress covert messages as generated images move through public platforms. Unlike white-box adversarial attacks, these transformations correspond to image I/O, resizing, compression, filtering, enhancement, and user edits in the threat model of Sec. 3.

Single-attack conclusions. For defense composition, no benign operation uniformly erases every hidden channel, but mild resampling, recompression, and filtering break complementary invariants while preserving visible semantics. Table 4 shows that operations which change sample correspondence are the strongest single-step disruptors: on MSCOCO, nearest-neighbor rescaling reduces GSD from 98.50% to 77.44% on average, while bilinear and bicubic keep it near 94%. JPEG and nonlinear filtering expose decoder-specific weaknesses: JPEG causes 6–9 point average drops for the strongest high-identity methods, and median filtering is much harsher than Gaussian blur for GSD (89.47% vs. 95.87%) and LDStega (78.67% vs. 80.62%). This follows from the operator geometry: JPEG projects coefficients onto a quantized DCT lattice, median filtering applies local order-statistic selection, and Gaussian blur mostly attenuates high frequencies.

Compound lifecycles. We next ask whether benign processing becomes progressively more effective when repeated along an image lifecycle. Let $x_{t+1} = P_t(x_t)$ be the image after the t-th platform or user-side operation, and let $A(t)$ be the recovered bit accuracy from x_t. A natural defender-side expectation is that deeper lifecycles should keep accumulating anti-stego effects. Instead, Fig. 7 and Fig. 6 show cumulative but strongly sublinear degradation. In the deterministic lifecycle view, repeated JPEG and resize reduce mean accuracy from 92.45% at $T = 0$ to 77.62% and 77.36% at $T = 16$, while stronger structured pipelines reach 68.78% for the platform block and 62.73% for the human–platform block. Common processing can damage covert channels, but increasing lifecycle depth does not yield proportional defensive gain.

Table 4: Defense-oriented robustness audit on MSCOCO at latent embedding rate $ER_l = 1$, using SD1.5 with DPM-Solver++ at 20 steps and guidance 5. The identity column is the PNG/uint8 storage control. Each attack cell reports **min / max / avg (difference)** extraction accuracy (%) over three severities; difference is average minus identity, in percentage points. Each setting contains 10 valid images.

Method	Identity	Rescaling			JPEG	Blur		Noise	Crop	Contrast	Sharpen	Brightness	Saturation
	(No atk)	NN	Bilinear	Bicubic	Q	Median	Gaussian	Gaussian					
BaseGen	86.25	64.44 / 74.62 69.91 (-16.34)	78.67 / 84.90 82.47 **(-3.78)**	77.89 / 85.52 82.51 **(-3.74)**	74.09 / 80.45 77.33 (-8.92)	73.97 / 84.03 79.17 **(-7.09)**	81.29 / 84.94 83.30 (-2.96)	75.38 / 85.36 80.11 (-6.14)	69.60 / 82.30 75.81 **(-10.44)**	81.89 / 85.06 83.47 (-2.78)	84.06 / 85.21 84.61 (-1.64)	85.87 / 86.11 85.99 (-0.26)	85.49 / 85.93 85.75 (-0.50)
S2IRT	90.47	72.81 / 77.00 74.88 **(-15.59)**	80.17 / 88.41 84.98 (-5.49)	79.43 / 89.26 85.09 (-5.38)	78.72 / 84.18 81.44 (-9.03)	76.66 / 86.32 81.17 (-9.29)	83.18 / 88.44 85.97 (-4.50)	79.17 / 89.77 84.26 (-6.21)	74.77 / 84.35 79.14 (-11.33)	83.52 / 89.16 86.29 (-4.18)	87.24 / 88.91 88.05 (-2.42)	89.99 / 90.40 90.17 (-0.29)	89.64 / 90.35 90.04 (-0.42)
PARIS	86.60	64.56 / 74.88 70.12 (-16.48)	78.75 / 85.42 82.77 (-3.83)	77.92 / 85.91 82.76 (-3.84)	75.02 / 81.38 78.26 (-8.34)	73.75 / 84.08 79.01 (-7.60)	81.27 / 85.45 83.52 (-3.09)	75.64 / 85.72 80.61 (-6.00)	69.83 / 82.40 75.98 (-10.63)	82.29 / 85.45 83.81 (-2.79)	84.46 / 85.61 85.01 (-1.59)	86.36 / 86.49 86.43 (-0.17)	85.97 / 86.38 86.19 (-0.42)
GRDH	97.93	68.13 / 82.46 75.80 (-22.13)	86.59 / 96.59 92.66 (-5.27)	85.02 / 97.14 92.45 (-5.47)	85.26 / 93.07 89.46 (-8.47)	79.73 / 94.34 86.98 (-10.95)	91.43 / 96.61 94.29 (-3.63)	86.03 / 97.42 92.07 (-5.85)	77.85 / 92.62 84.85 (-13.08)	92.89 / 97.21 95.12 (-2.80)	96.00 / 97.10 96.54 (-1.38)	97.66 / 97.83 97.77 **(-0.15)**	97.58 / 97.79 97.71 **(-0.22)**
GSD	98.50	69.44 / 84.30 **77.44** (-21.06)	88.95 / 97.64 **94.37** (-4.13)	87.35 / 98.07 **94.01** (-4.49)	87.15 / 94.28 90.93 (-7.57)	82.28 / 96.45 **89.47** (-9.03)	93.53 / 97.71 **95.87 (-2.63)**	88.21 / 98.26 **93.48 (-5.02)**	78.59 / 93.51 **85.68** (-12.82)	94.35 / 97.73 **96.08** (-2.42)	96.84 / 97.73 **97.27 (-1.23)**	98.12 / 98.39 **98.29** (-0.21)	98.09 / 98.39 **98.27** (-0.23)
MDDM	86.63	64.41 / 74.88 70.00 (-16.63)	78.94 / 85.23 82.65 (-3.98)	78.54 / 85.87 82.85 (-3.79)	76.62 / 81.70 79.29 (-7.34)	74.18 / 83.89 79.04 (-7.59)	81.28 / 85.18 83.37 (-3.26)	77.03 / 85.57 81.32 (-5.32)	69.49 / 82.34 75.76 (-10.87)	83.34 / 85.77 84.59 **(-2.04)**	84.64 / 85.67 85.08 (-1.55)	86.16 / 86.42 86.33 (-0.30)	86.18 / 86.46 86.34 (-0.29)
LDStega	98.16	61.82 / 76.13 69.13 (-29.04)	79.84 / 93.23 87.75 (-10.41)	85.70 / 97.33 92.93 (-5.24)	87.31 / 95.05 **91.51 (-6.66)**	67.89 / 91.46 78.67 (-19.49)	71.07 / 90.17 80.62 (-17.55)	81.29 / 97.84 89.88 (-8.28)	75.01 / 91.25 82.99 (-15.17)	85.60 / 95.53 90.43 (-7.73)	89.04 / 95.02 92.04 (-6.12)	97.06 / 98.15 97.50 (-0.67)	97.02 / 98.06 97.66 (-0.50)

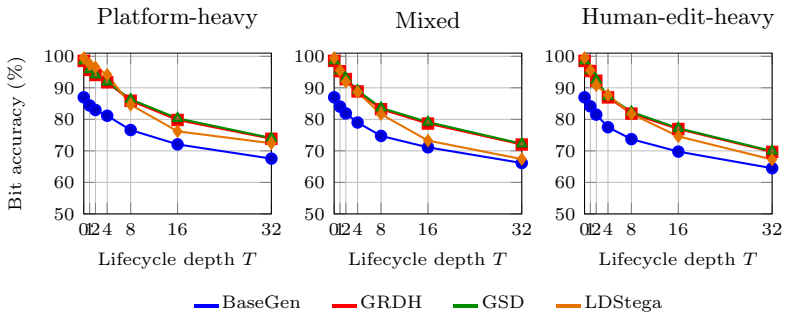


Figure 6: Randomized-lifecycle trajectories on MSCOCO for the completed SD2.1-DDIM subset. Curves report mean bit accuracy across ten lifecycle seeds for each controlled regime and depth prefix.

Define cumulative defense gain as $L(T) = A(0) - A(T)$ and interval gain as

$$\Delta(T_1, T_2) = L(T_2) - L(T_1) = A(T_1) - A(T_2). \tag{1}$$

The randomized curves indicate that $\Delta(8, 16)$ is already comparable to $\Delta(16, 32)$, although the latter interval contains twice as many steps. Equivalently, $(A(8) - A(16))/8 \approx 2(A(16) - A(32))/16$, so the per-step anti-stego effect after depth 16 is roughly halved in this setting. We use this as evidence of saturating, sublinear degradation rather than as a fitted functional law. Mechanistically, early processing removes fragile low-margin carriers first; later surviving components are aligned with more stable spatial, frequency, or generative invariants, so another mild transformation has weaker marginal effect.

This lifecycle result refines the defender-side implication. Randomized platform-heavy, mixed, and human-edit-heavy regimes still lower recovery to 71.96%, 69.46%, and 67.84% at $T = 32$, respectively, confirming that benign processing is useful as a disruption mechanism. Yet the observed saturation means that ordinary propagation cannot be treated as a reliable erasure channel: after the early lifecycle, each additional single processing event contributes less incremental protection. In practical social-media lifecycles, where images often experience only a few rounds of upload, recompression, resizing, and reposting, this leaves a residual covert-channel risk rather than eliminating generative steganography by default.

5.3 Supporting Dimensions: Practical Guardrails

The two security audits above focus on covert-channel robustness; we therefore use visual quality and detectability only as practical guardrails. For the completed five-method guardrail subset at $ER_l = 1$, embedding causes small quality shifts (ΔFID $= 7.74$–8.26 and near-zero BRISQUE change relative to clean generations), while XuNet (Xu et al., 2016), SRNet (Boroumand et al., 2019), and YeNet (Yedroudj et al., 2018) remain close to random guessing (45.0–56.8% detection accuracy). These results indicate no obvious perceptual or classical steganalysis artifact in that auxiliary setting; full quality tables, qualitative examples, and the detection protocol are provided in Appendix D.

6 Related Work

Generative Image Steganography. Generative image steganography has been instantiated on GANs (Radford et al., 2016; Karras et al., 2019; Kang et al., 2023), flow-based models (Dinh et al., 2015; 2017; Kingma & Dhariwal, 2018), and diffusion or diffusion-transformer models (Ho et al.,

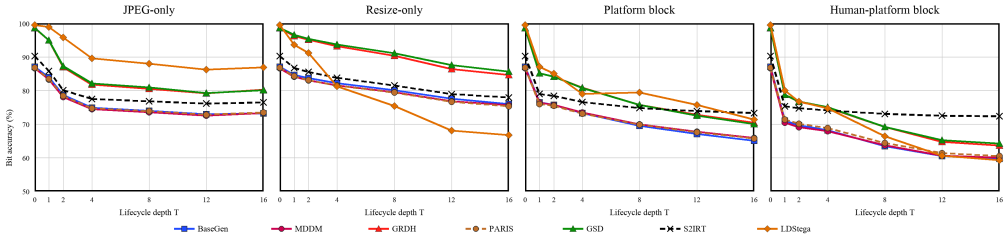


Figure 7: Lifecycle-depth trajectories on MSCOCO for the completed SD2.1-DDIM fixed-generator slice. Each curve reports mean bit accuracy for one steganographic method under one repeated lifecycle policy, using depth points $T \in \{0, 1, 2, 4, 8, 12, 16\}$.

2020; Song et al., 2020; Rombach et al., 2022; Ramesh et al., 2022; Podell et al., 2023; Peebles & Xie, 2023; Chen et al., 2023; Esser et al., 2024; Black Forest Labs, 2024). Recent latent-vector methods such as S2IRT (Zhou et al., 2023), PARIS (Yang et al., 2024b), GRDH (Hu et al., 2024), and GSD (Zhou et al., 2025) encode information through positional rearrangement, distribution matching, dual-key projection, or frequency-domain modulation. Other generative steganographic systems, including CRoSS (Yu et al., 2023), DiffStega (Yang et al., 2024a), StegaStyleGAN (Su et al., 2024), and SCRIPT (Zeng et al., 2024), target related coverless, container, distribution-preserving, or black-box settings and fall outside the latent-vector schema audited here.

Robustness Evaluation in Steganography. Robustness evaluation has long been central to steganography. Conventional robust methods are usually tested under specific perturbation families, such as JPEG compression (Zeng et al., 2023b;a), rescaling (Punia & Malik, 2022; Zhang et al., 2018a;b), or interpolation-based scaling attacks (Zhu et al., 2021). Recent work studies lossless recovery after downsampling with pre-detected transmission scenarios (Li et al., 2024) or recovery without pre-detection under restricted scaling operations (Song et al., 2025). GenStegoBench differs by auditing generative-image covert channels across generator regimes, decoder geometries, and sequential multimedia lifecycles.

7 Conclusion

We introduced *GenStegoBench*, a security audit benchmark for persistent covert channels in generative image systems. Our results show that covert-channel reliability is governed by the interaction between generator recovery geometry, decoder margins, payload pressure, and channel transformations. From the attacker side, frequency-domain carriers and generator-aligned spatial carriers are more reliable when their message-bearing directions remain stable under the measured generation–recovery loop. From the defender side, sample-selection transformations are effective single-step disruptors, but lifecycle-level gains saturate: repeated benign processing degrades recovery sublinearly and therefore cannot be treated as automatic channel erasure. GenStegoBench thus reframes generative steganography evaluation as an AI security problem spanning attack-side reliability, defender-side disruption, and threat-model-aware multimedia lifecycles.

AI use statement

In this work, we used generative AI tools to polish the language of the manuscript and to assist with programming and code editing. We did not use generative AI tools to generate datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims or proofs, or support qualitative or thematic data analysis. Although some figures in the paper contain images produced by generative image systems, these images are part of the research subject, rather than images generated by AI tools for manuscript preparation or idea presentation.

All AI-assisted text and code were reviewed by the authors for correctness, accuracy, consistency with the paper's claims, and faithfulness to the reported experiments. The authors take responsibility for the final content of this work, including any text, code, claims, figures, or artifacts produced with the aid of generative AI.

REFERENCES

Black Forest Labs. FLUX.1-schnell. Hugging Face model card, 2024. URL https://huggingface.co/black-forest-labs/FLUX.1-schnell. Accessed: 2026-09-07.

Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, 2019. doi: 10.1109/TIFS.2018.2871749.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. URL https://arxiv.org/abs/2310.00426.

Jinyang Ding, Kejiang Chen, Yaofei Wang, Na Zhao, Weiming Zhang, and Nenghai Yu. Discop: Provably secure steganography in practice based on "distribution copies". In *2023 IEEE Symposium on Security and Privacy (SP)*, pp. 2238–2255. IEEE, May 2023. ISBN 978-1-66549-336-9. doi: 10.1109/SP46215.2023.10179287. URL https://ieeexplore.ieee.org/document/10179287.

Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation, 2015. URL https://arxiv.org/abs/1410.8516.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp, 2017. URL https://arxiv.org/abs/1605.08803.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. URL https://arxiv.org/abs/2403.03206.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: data, models and evaluation metrics. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pp. 4188–4192. AAAI Press, 2015. ISBN 9781577357384.

Xiaoxiao Hu, Sheng Li, Qichao Ying, Wanli Peng, Xinpeng Zhang, and Zhenxing Qian. Establishing robust generative image steganography via popular stable diffusion. *IEEE Transactions on Information Forensics and Security*, 19:8094–8108, 2024. doi: 10.1109/TIFS.2024.3444311.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2019. URL https://arxiv.org/abs/1812.04948.

Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions, 2018. URL https://arxiv.org/abs/1807.03039.

Sheng Li, Zichi Wang, Xiudong Zhang, and Xinpeng Zhang. Robust image steganography against general downsampling operations with lossless secret recovery. *IEEE Transactions on Dependable and Secure Computing*, 21(1):340–352, 2024. doi: 10.1109/TDSC.2023.3253691.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. URL https://api.semanticscholar.org/CorpusID:14113767.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023. URL https://arxiv.org/abs/2212.09748.

Yinyin Peng, Yaofei Wang, Donghui Hu, Kejiang Chen, Xianjin Rong, and Weiming Zhang. Ldstega: Practical and robust generative image steganography based on latent diffusion models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, pp. 3001–3009, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400706868. doi: 10.1145/3664647.3681635. URL https://doi.org/10.1145/3664647.3681635.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis, 2023. URL https://arxiv.org/abs/2307.01952.

Riya Punia and Aruna Malik. Image interpolation-based steganographic techniques under spatial domain: A survey. In *International Conference on Computing, Communications, and Cyber-Security*, pp. 671–685. Springer, 2022.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016. URL https://arxiv.org/abs/1511.06434.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv*, abs/2204.06125, 2022. URL https://api.semanticscholar.org/CorpusID:248097655.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, June 2022.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, July 2018. doi: 10.18653/v1/P18-1238. URL https://aclanthology.org/P18-1238/.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, October 2020. URL https://arxiv.org/abs/2010.02502.

Ruizhe Song, Wei Tong, and Sheng Zhong. Robust steganographic system against multi-scale scaling attacks. In *GLOBECOM 2025 - 2025 IEEE Global Communications Conference*, pp. 5175–5180, 2025. doi: 10.1109/GLOBECOM59602.2025.11432596.

Wenkang Su, Jiangqun Ni, and Yiyan Sun. Stegastylegan: towards generic and practical generative image steganography. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i1.27776. URL https://doi.org/10.1609/aaai.v38i1.27776.

Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708–712, 2016. doi: 10.1109/LSP.2016.2548421.

Zihao Xu, Dawei Xu, Zihan Li, and Chuan Zhang. MDDM: Practical Message-Driven Generative Image Steganography Based on Diffusion Models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 69832–69848, 2025. URL https://mlanthology.org/icml/2025/xu2025icml-mddm/.

Yiwei Yang, Zheyuan Liu, Jun Jia, Zhongpai Gao, Yunhao Li, Wei Sun, Xiaohong Liu, and Guangtao Zhai. Diffstega: Towards universal training-free coverless image steganography with diffusion models. In *International Joint Conference on Artificial Intelligence*, 2024a. URL https://api.semanticscholar.org/CorpusID:271213242.

Zijin Yang, Kejiang Chen, Kai Zeng, Weiming Zhang, and Nenghai Yu. Provably secure robust image steganography. *IEEE Transactions on Multimedia*, 26:5040–5053, 2024b. doi: 10.1109/TMM.2023.3330098.

Mehdi Yedroudj, Frederic Comby, and Marc Chaumont. Yedrouj-net: An efficient cnn for spatial steganalysis, 2018. URL https://arxiv.org/abs/1803.00407.

Jiwen Yu, Xuanyu Zhang, Youmin Xu, and Jian Zhang. Cross: Diffusion model makes controllable, robust and secure image steganography, 2023. URL https://arxiv.org/abs/2305.16936.

Kai Zeng, Kejiang Chen, Weiming Zhang, and Yaofei Wang. Upward robust steganography based on overflow alleviation. *IEEE Transactions on Multimedia*, 26:299–312, 2023a.

Kai Zeng, Kejiang Chen, Weiming Zhang, Yaofei Wang, and Nenghai Yu. Robust steganography for high quality images. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9): 4893–4906, 2023b.

Kai Zeng, Kejiang Chen, Jiansong Zhang, Weiming Zhang, and Nenghai Yu. Toward secure and robust steganography for black-box generated images. *IEEE Transactions on Information Forensics and Security*, 19:3237–3250, 2024. doi: 10.1109/TIFS.2024.3361220.

Yue Zhang, Xiangyang Luo, Jinwei Wang, Chunfang Yang, and Fenlin Liu. A robust image steganography method resistant to scaling and detection. *Journal of Internet Technology*, 19 (2):607–618, 2018a.

Yue Zhang, Dengpan Ye, Junjun Gan, Zhenyu Li, and Qingfeng Cheng. An image steganography algorithm based on quantization index modulation resisting scaling attacks and statistical detection. *Computers, Materials & Continua*, 56(1), 2018b.

Qing Zhou, Ping Wei, Zhenxing Qian, Xinpeng Zhang, and Sheng Li. Improved generative steganography based on diffusion model. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(7):6494–6507, 2025. doi: 10.1109/TCSVT.2025.3539832.

Zhili Zhou, Yuecheng Su, Jin Li, Keping Yu, Q. M. Jonathan Wu, Zhangjie Fu, and Yunqing Shi. Secret-to-image reversible transformation for generative steganography. *IEEE Transactions on Dependable and Secure Computing*, 20(5):4118–4134, 2023. doi: 10.1109/TDSC.2022.3217661.

Liyan Zhu, Xiangyang Luo, Yi Zhang, Chunfang Yang, and Fenlin Liu. Inverse interpolation and its application in robust image steganography. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6):4052–4064, 2021.