CHARACTERIZING DETECTABILITY IN 3DGS POISONING: A STAGE-WISE BENCHMARK

Anonymous authors
Paper under double-blind review

ABSTRACT

3D Gaussian Splatting (3DGS) has rapidly emerged as a leading representation for real-time novel view synthesis, but recent work has shown that it is vulnerable to diverse poisoning attacks, including illusory object injection, computation cost amplification, and post hoc model watermarking. Despite this expanding threat surface, existing studies primarily focus on attack success, while defense and detection remain underexplored. From a detection perspective, a key challenge and opportunity arise from the multi-stage nature of the 3DGS reconstruction pipeline, which produces heterogeneous intermediate representations. Importantly, forensic signals for detecting poisoning are inherently stage dependent: an attack introduced at one stage may produce signals that emerge only at later stages or become more detectable as the pipeline progresses. This motivates a *stage-wise view of detectability* that goes beyond single-stage evaluation. **In this work,** we introduce **Poison-3DGS**, a benchmark designed for *stage-wise characterization of poisoning detection in 3DGS*, comprising 100 diverse base scenes and 1,253 poisoned variants. Moving beyond attack-specific evaluation, Poison-3DGS systematically exposes the stage-specific representations naturally produced throughout the standard 3DGS reconstruction pipeline, including multi-view images, SfM-derived geometry, training dynamics, and the final Gaussian representation. This representation-centric design enables an attack-agnostic, stage-wise evaluation protocol that naturally accommodates existing and future poisoning attacks. Using this benchmark, we conduct the first systematic study of poisoning detectability across the 3DGS reconstruction pipeline. Our study reveals that the stage where an attack is introduced is not necessarily the stage where it is most detectable. Instead, downstream reconstruction transforms attack effects into stage-specific forensic signals whose detectability varies substantially across stages and attack types. Overall, Poison-3DGS provides a principled benchmark and the first systematic characterization of stage-dependent detectability in 3DGS, establishing a foundation for future research on robust and reliable 3DGS systems. **We include the code and benchmark in the submission.**

1 INTRODUCTION

3D Gaussian Splatting (3DGS) Kerbl et al. (2023); Chen & Wang (2024) has recently emerged as a leading representation for real-time photorealistic scene reconstruction, with growing deployment in robotics, augmented and virtual reality, mapping, and content creation Zhai et al. (2025); Chen et al. (2024); Matsuki et al. (2024). As 3DGS moves into production pipelines, the integrity of its training inputs and intermediate artifacts becomes a concrete security concern.

A growing number of poisoning attacks now target 3DGS under fundamentally different threat models. StealthAttack Ke et al. (2025) composites an illusory object into a target view and injects Gaussian seed points into low-density voids of the Structure-from-Motion (SfM) point cloud Schönberger & Frahm (2016). Poison-Splat Lu et al. (2024) adds bounded adversarial noise to training images and forces the optimizer to over-densify, inflating Gaussian count, GPU memory, and training time by a factor of two to seven. GaussTrap Hong et al. (2025) performs post-hoc fine-tuning to implant malicious scene-confusion content at designated viewpoints while preserving rendering fidelity at benign viewpoints. Meanwhile, 3D-GSW Jang et al. (2025) finetunes a trained model to embed a decodable covert watermark in Gaussian colour, opacity, rotation, and scale. GuardSplat Chen

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.

et al. (2025b) similarly modifies a trained model by embedding messages into spherical harmonic features. Although originally proposed for watermarking or ownership protection, an adversary with write access to a trained 3DGS asset can repurpose these mechanisms to embed arbitrary, potentially malicious hidden content without the owner's authorization, thereby covertly modifying the learned representation while preserving normal rendering quality. We therefore consider such adversarial post-training modifications as model-level poisoning under a model-integrity threat. Together, these attacks span data-level, training-level, and model-level poisoning across the 3DGS pipeline, from input data to the final Gaussian representation.

Research Gap. Existing 3DGS attack studies Ke et al. (2025); Chen et al. (2025b); Jang et al. (2025); Lu et al. (2024) focus on attacker-centric metrics, such as visual stealth, Gaussian inflation ratio, or watermark recovery accuracy, but lack a systematic, defender-centric evaluation of detectability. From a detection perspective, a key challenge and opportunity arise from the multi-stage nature of the 3DGS reconstruction pipeline. The process involves stage-specific artifacts, including multi-view images, geometry, training dynamics, and Gaussian parameters. As a result, forensic signals for detecting poisoning are inherently *stage dependent*. Importantly, an attack introduced at one stage may produce an attack footprint whose corresponding forensic signal emerges only at later stages or becomes more detectable as the pipeline progresses. This multi-stage structure creates both challenges and opportunities for detection, as different stages expose distinct and complementary evidence, a property that is particularly pronounced in 3DGS compared to conventional data poisoning settings.

This motivates a deeper view of 3DGS poisoning detection: beyond asking whether detection is possible, it is important to study *where* forensic signals arise and become detectable. While detection has been extensively studied in 2D image forensics Dong et al. (2022); Liu et al. (2024); Su et al. (2025) and 3D anomaly detection Zhou et al. (2024), these settings typically assume a single representation or uniform access to data. In contrast, 3DGS is a multi-stage pipeline where detection performance could depend on which stage is analyzed. Despite this, there is no unified framework or benchmark that enables systematic, stage wise characterization of poisoning detectability.

In this paper, we address this gap by introducing **Poison-3DGS**, a benchmark for studying 3DGS poisoning detection from a stage wise perspective. Building on the stage-dependent view illustrated in Fig. 1, we move beyond treating detection as a single-stage problem and instead characterize how forensic signals arise and become detectable across the 3DGS pipeline. To this end, Poison-3DGS is designed to expose stage-specific artifacts and enable systematic analysis of detectability across different stages. Using this benchmark, we conduct a comprehensive study across representative attacks and derive key insights into stage-dependent detection behavior. Our contributions are summarized as follows.

- **Stage-wise benchmark design.** We propose **Poison-3DGS**, a benchmark comprising 100 diverse base scenes and 1,253 poisoned variants that exposes the heterogeneous representations naturally produced throughout the standard 3DGS reconstruction pipeline: multi-view images, 3D information, training dynamics, and final Gaussian representations. Poison-3DGS's central design principle is that the stage structure is defined independently of the attack space, allowing new attacks to be incorporated under the same benchmark structure.
- **Systematic study of stage-wise poisoning detectability.** Enabled by Poison-3DGS, we systematically study poisoning detectability under an *attack-agnostic* evaluation protocol. Detectors are assigned according to the representation exposed at each stage rather than to specific attack families. The same fixed, stage-appropriate detectors are evaluated across poisoning attacks without attack labels, poisoned-sample training, or attack-specific adaptation, reflecting realistic deployment where the incoming attack is unknown and enabling consistent comparison of stage-wise detectability.
- **Empirical insights on stage-dependent detectability.** Our study characterizes where attack-induced forensic signals become detectable across the 3DGS reconstruction pipeline. Under the evaluated detector suite, no single stage consistently dominates across attacks. Importantly, *an attack's injection stage and its stage of strongest detectability can be decoupled*: attack signals introduced early in the pipeline may become most detectable only after subsequent reconstruction or optimization. These results reveal where existing detectors capture stage-specific forensic signals and where poisoning remains difficult to detect.

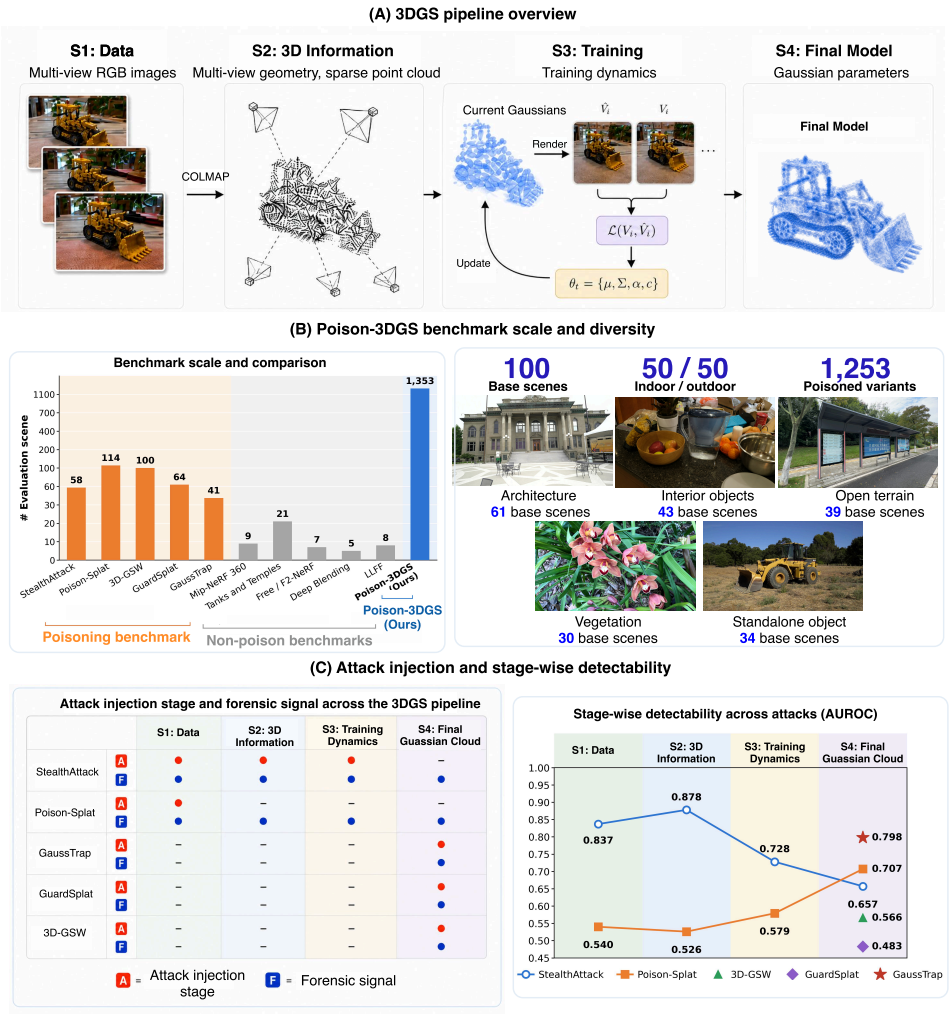

Figure 1: **Our stage-wise view of 3DGS poisoning detection.** (a) The multi-stage 3DGS reconstruction pipeline. We define a *stage* as a distinct point in the pipeline where a particular representation becomes naturally available for forensic analysis; these stage-specific representations include multi-view images, geometry, training dynamics, and the final Gaussian representation. (b) In addition to large scale and coverage, our benchmark supports stage-wise characterization of detectability and analysis of forensic signals at different stages. The stage-wise formulation enables understanding of the contribution of individual sources of evidence. Base-scene diversity is characterized across five main scene characteristics. Each base scene may exhibit multiple characteristics; hence, the counts are non-exclusive and may exceed 100 in total. (c) *Attack injection stages* and their corresponding *forensic signals* across the pipeline. The plot reports the best AUROC among evaluated detectors at each applicable stage. **An attack introduced at one stage may be most detectable at another**, demonstrating that attack injection and strongest detectability can be *decoupled*.

2 Related Work

3DGS poisoning threats. Recent work shows that 3D Gaussian Splatting is vulnerable to attacks that directly affect users of reconstructed scenes. StealthAttack Ke et al. (2025) targets scene integrity by injecting viewpoint-dependent illusory content through coordinated image and SfM manipulation, while Poison-Splat Lu et al. (2024) targets availability by perturbing training images to trigger excessive densification, increasing Gaussian count, memory use, and training cost. Although watermarking methods such as 3D-GSW Jang et al. (2025) and GuardSplat Chen et al. (2025b) are designed for ownership protection, they also demonstrate that a converged 3DGS model can be

modified directly at the final Gaussian representation to encode hidden information while preserving visual quality. GaussTrap Hong et al. (2025) further shows that poisoned views can induce targeted scene confusion. These studies establish a growing threat surface for 3DGS, but their evaluations remain mostly attack-centric: each work studies its own scenes, artifacts, and success criteria. Poison-3DGS instead studies these manipulations under a unified detector-centric protocol, with the goal of characterizing where forensic signals become detectable across the 3DGS pipeline.

Defenses for 3DGS poisoning. Recent 3DGS defenses study parts of 3DGS pipeline, particularly for computation-cost attacks. RemedyGS trains a supervised detector and purifier using Poison-Splat-specific data, while DefenseSplat suppresses attack-induced Gaussian growth through frequency-based filtering and regularization Li et al. (2025a); Qiao et al. (2026). *These methods study mitigation mechanisms for their targeted attacks, but are largely developed and evaluated with knowledge of a specific attack or attack mechanism.* In practical deployment, however, a defender may not know a priori which attack, if any, has compromised an incoming scene. *Poison-3DGS therefore takes a complementary attack-agnostic perspective*: the benchmark brings heterogeneous attacks into a unified evaluation setting and evaluates fixed, stage-appropriate detectors without attack labels or attack-specific adaptation. This enables a systematic study of stage-wise poisoning detectability under a common attack-agnostic protocol.

3 Preliminaries: 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) Kerbl et al. (2023) represents a scene as a set of anisotropic Gaussian primitives $\mathcal{G} = \{g_i\}_{i=1}^{N}$. Each Gaussian $g_i = (\mu_i, \Sigma_i, \alpha_i, c_i)$ contains a 3D center μ_i, covariance Σ_i, opacity α_i, and view-dependent color coefficients c_i. Given posed multi-view images $\mathcal{D} = \{(V_k, \Pi_k)\}_{k=1}^{K}$, where V_k is the ground-truth image and Π_k denotes the camera parameters, 3DGS renders a predicted view $\hat{V}_k = \mathcal{R}(\mathcal{G}; \Pi_k)$ through differentiable splatting and alpha compositing.

Training optimizes the Gaussian parameters by minimizing a photometric reconstruction loss:

$$\min_{\mathcal{G}} \mathcal{L}_{\text{photo}}(\mathcal{D}) = \min_{\mathcal{G}} \sum_{k=1}^{K} \left[(1-\lambda)\mathcal{L}_1(\hat{V}_k, V_k) + \lambda \mathcal{L}_{\text{D-SSIM}}(\hat{V}_k, V_k) \right], \quad \hat{V}_k = \mathcal{R}(\mathcal{G}; \Pi_k), \quad (1)$$

where λ balances pixel-wise and structural reconstruction terms. During optimization, 3DGS also performs adaptive density control, which dynamically adds and removes Gaussian primitives. Gaussians with large view-space positional gradients are densified through cloning or splitting, while Gaussians with low opacity are pruned. This process allows 3DGS to allocate more capacity to regions that remain difficult to reconstruct, but it also makes the number of Gaussians and their parameter statistics part of the training dynamics.

4 Poison-3DGS: A Stage-wise Benchmark for Poisoning Detectability

4.1 Stage-wise Benchmark Design and Evaluation Protocol

Existing 3DGS security works are primarily attack-centric: they evaluate whether an injected object appears, whether perturbations remain imperceptible, whether computation cost increases, or whether a watermark can be decoded. These metrics validate attack success, but they do not answer the defender-side question studied here: at which stage of the 3DGS pipeline can poisoning be detected? Prior poisoning and watermarking papers evaluate their threat models in isolation, often using different scenes, artifacts, and success criteria. Poison-3DGS instead turns heterogeneous 3DGS attacks into a unified stage-wise detection benchmark.

Threat Model. We consider poisoning threats that compromise the integrity or availability of a 3DGS model through adversarial manipulation at different points of the reconstruction pipeline, spanning data-level, training-level, and model-level poisoning. Attack-specific mechanisms and adversarial capabilities follow the threat models of the corresponding attacks Hong et al. (2025); Chen et al. (2025b); Ke et al. (2025); Lu et al. (2024); Jang et al. (2025). *From the defender's perspective, however, the evaluation is attack-agnostic*: the defender does not know the identity of the incoming attack and seeks to determine whether a scene has been poisoned using only the representation

available at a given pipeline stage. Accordingly, detectors receive no attack labels, poisoned samples for training, or attack-specific adaptation. For model-level attacks such as GaussTrap, 3D-GSW and GuardSplat, we additionally consider an integrity threat in which an adversary with write access to a trained 3DGS model repurposes these mechanisms to embed arbitrary, potentially malicious hidden content without the owner's authorization while preserving normal rendering quality. The defender receives the released model without the embedding key or decoder and must determine whether it has been covertly modified.

Stage-wise characterization of poisoning detection. Poison-3DGS enables poisoning detection studies under a unified detector-centric protocol over four stage-specific artifact groups: **S1: Data**, raw multi-view RGB images; **S2: 3D Information**, SfM cameras, tracks, and sparse points; **S3: Training Dynamics**, render–loss–update behavior including losses, gradients, and densification; and **S4: Final Model**, trained Gaussian parameters. A detector can operate on the artifact group of a given stage and produces a scene-level anomaly score. This stage-wise view is necessary because an attack introduced at one stage may be more detectable at other stages. For example, Poison-Splat perturbs images at **S1**, but its signal may be more detectable later due to abnormal densification at **S3** and abnormal final Gaussian statistics at **S4**. Thus, Poison-3DGS enables evaluation of not only whether a scene is poisoned, but also *where* its forensic signal becomes detectable.

Benchmark Design Principles. Poison-3DGS follows three design principles. First, its stage structure is *attack-independent*: the four stages are derived from artifacts naturally exposed by the standard 3DGS reconstruction pipeline rather than from the attacks included in the benchmark. Second, detector assignment is *representation-driven*: detector applicability is determined by compatibility with the artifact exposed at each stage, rather than by attack identity. Third, evaluation is *attack-agnostic*: once assigned to a stage, each detector uses the same weights, configuration, aggregation rule, and anomaly-scoring procedure for every applicable attack, without poisoned samples for detector training or attack-specific adaptation. Consequently, new attacks can be evaluated within the same stage-wise structure without redefining the benchmark or its detector-access rules.

Dataset. Poison-3DGS contains 100 unique clean scenes drawn from eight datasets: Mip-NeRF 360 Barron et al. (2022), Tanks-and-Temples Knapitsch et al. (2017), Free Wang et al. (2023), Deep Blending Hedman et al. (2018), LLFF Mildenhall et al. (2019), ARKitScenes Dehghan et al. (2021), ScanNet++ Yeshwanth et al. (2023), and DL3DV Ling et al. (2024). The pool is balanced between 50 indoor and 50 outdoor scenes and covers object-centric captures, unbounded environments, interior spaces, and large architectural scenes. For every clean scene, we provide the multi-view images, SfM reconstruction, training checkpoints, and final 3DGS model. Each poisoned variant is linked to its clean base scene, making scene-level binary detection well-defined under a shared scene distribution.

Attack Space. Poison-3DGS, illustrated in Fig. 2, includes five attack families spanning data-level, geometry-level, training-time, and final-model interventions. It contains 1,253 realized attack variants: 220 StealthAttack Ke et al. (2025), 264 Poison-Splat Lu et al. (2024), 254 3D-GSW Jang et al. (2025), 263 GuardSplat Chen et al. (2025b) and 252 GaussTrap Hong et al. (2025) variants. For StealthAttack, we vary the injection setup, starting noise scale, and decay schedule. For Poison-Splat, we vary the perturbation budget and poisoned-view ratio, including partial-poisoning and unconstrained settings. For 3D-GSW, we vary message length and embedding-loss weight. GuardSplat variants embed messages through spherical-harmonic offsets while preserving Gaussian geometry. For GaussTrap, we vary the use of Viewpoint Ensemble Stabilization, the stabilization angle, and the number of post-hoc fine-tuning epochs This controlled attack grid supports analysis of how attack mechanisms and configurations shape stage-specific forensic signals. Additional information are provided in Supp.

Metrics and Protocol. We evaluate poisoning detection with *binary scene-level detection*, with clean scenes as negatives and poisoned variants as positives. Each detector assigns a real-valued anomaly score to a scene at a given stage; no multiclass attack classifier is trained. All learned detectors use released pretrained weights, and no detector is trained on Poison-3DGS or given an attack identity. When a detector produces image-, view-, point-, checkpoint-, or Gaussian-level scores, we apply a fixed detector-specific aggregation rule to obtain one scene-level score. Following score-based anomaly-detection protocols Hendrycks & Gimpel (2016); Lee et al. (2018); Yang et al. (2022), we report AUROC and tie-correct FPR@95, computed as the first false-positive rate whose true-positive rate reaches 95%.

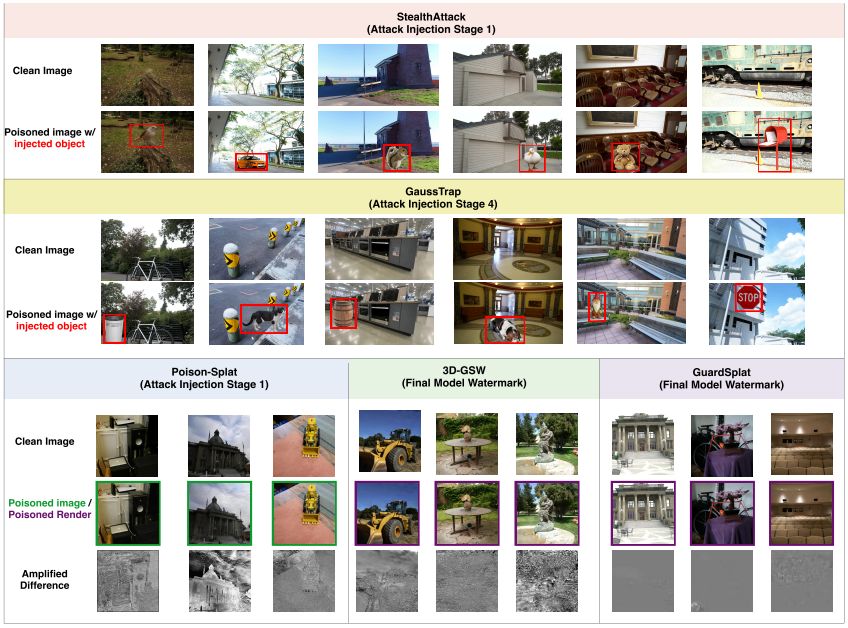

Figure 2: Qualitative examples from **Poison-3DGS**. StealthAttack shows *image injection* by inserting an illusory object into a target training image, GaussTrap implants malicious scene-confusion content, while Poison-Splat shows *image perturbation* through subtle adversarial noise and its amplified residual. For **3D-GSW** and **GuardSplat**, we render the same sample view from the clean model and the watermarked model, and visualize amplified differences.

4.2 Systematic Study of Stage-wise Poisoning Detectability

Enabled by Poison-3DGS, we conduct a systematic study of poisoning detectability across the 3DGS reconstruction pipeline. To achieve this, we identify a suite of existing detectors corresponding to the stage-specific representations naturally exposed by the standard 3DGS pipeline. *Rather than assigning detectors to attack families, detectors are assigned according to the representation available at each stage, enabling an attack-agnostic evaluation protocol.* All detectors use official pretrained weights and released checkpoints, with no poisoned-sample fine-tuning, attack-specific adaptation, or access to attack labels. Each detector receives only the representation available at its corresponding stage and produces a scene-level anomaly score using its standard inference procedure.

S1: Data. At the data stage, we identify detectors that operate on the raw multi-view training images $\mathcal{V} = \{V_k\}_{k=1}^{K}$. A data-stage attack modifies a subset of views $\mathcal{P}$ and produces manipulated images $V'_k = \mathcal{A}(V_k)$ for $k \in \mathcal{P}$, such as the target-view insertion in StealthAttack or the adversarial perturbations in Poison-Splat. We identify MVSS-Net Dong et al. (2022), IML-ViT Ma et al. (2024), and SparseViT Su et al. (2025), and apply each detector f_ψ independently to every training view, yielding a view-level score $s_k = f_\psi(V_k)$, or $s'_k = f_\psi(V'_k)$ for a manipulated view. Since poisoning can affect a small subset of views, we aggregate view-level scores by max pooling, $S_{\text{scene}} = \max_k s_k$. We also include SoftPatch+ Wang et al. (2025a), which estimates patch-level outlier scores and calibrates them using images from the same scene before producing a scene-level score.

S2: 3D Information. At the 3D information stage, we ask whether manipulated views become observable after SfM reconstruction. Given the input views, COLMAP recovers camera intrinsics/extrinsics, feature tracks, and a sparse point cloud $\mathcal{X} = \{x_j\}_{j=1}^{N}$ formed by triangulated 3D points. This stage exposes two complementary forensic signals. First, manipulated views may violate multi-view consistency: an inserted object may be supported only by the target view, while adversarial perturbations may disturb feature matching, correspondence, or view-consistent geometry. We therefore use MET3R Asim et al. (2025), MV-DUSt3R+ Tang et al. (2025), NoPoSplat Ye et al. (2024), and VGGT Wang et al. (2025b) as multi-view consistency probes. Their reconstruction or consistency error is used as a geometric anomaly score s_{geo} and aggregated into a scene-level score using the same max-pooling strategy as S1. Second, injection-based attacks can leave sparse-geometry anomalies;

Table 1: Stage-wise detector performance across poisoning attacks. Rows are grouped by the 3DGS stage at which each detector operates. Entries report scene-level AUROC and FPR@95. Overall reports the macro-average across attacks applicable at that stage. "N/A" indicates that the attack does not modify the detector's input artifact at that stage (i.e., clean and poisoned artifacts are identical). Bold denotes the best result in each metric column among applicable methods.

Method	Overall		StealthAttack		Poison-Splat		GaussTrap		3D-GSW		GuardSplat	
	AUROC↑	FPR@95↓	AUROC↑	FPR@95↓	AUROC↑	FPR@95↓	AUROC↑	FPR@95↓	AUROC↑	FPR@95↓	AUROC↑	FPR@95↓
S1: Data *multi-view RGB images*												
SoftPatch+	0.648	0.770	0.837	0.570	0.459	0.970	N/A	N/A	N/A	N/A	N/A	N/A
SparseViT	0.657	0.840	0.773	0.780	0.540	0.900	N/A	N/A	N/A	N/A	N/A	N/A
MVSS-Net	0.513	0.945	0.707	0.940	0.318	0.950	N/A	N/A	N/A	N/A	N/A	N/A
IML-ViT	0.564	0.955	0.701	0.940	0.427	0.970	N/A	N/A	N/A	N/A	N/A	N/A
S2: 3D Information *multi-view geometry*												
MET3R	0.633	0.735	0.744	0.540	0.521	0.930	N/A	N/A	N/A	N/A	N/A	N/A
VGGT	0.550	0.915	0.635	0.830	0.464	1.000	N/A	N/A	N/A	N/A	N/A	N/A
MV-DUSt3R+	0.506	0.940	0.486	0.960	0.526	0.920	N/A	N/A	N/A	N/A	N/A	N/A
NoPoSplat	0.411	0.965	0.508	0.950	0.314	0.980	N/A	N/A	N/A	N/A	N/A	N/A
S2: 3D Information *sparse point cloud*												
Sonata	0.878	0.550	**0.878**	0.550	N/A‡	N/A‡	N/A	N/A	N/A	N/A	N/A	N/A
Concerto	0.833	0.530	0.833	**0.530**	N/A‡	N/A‡	N/A	N/A	N/A	N/A	N/A	N/A
S3: Training *training dynamics*												
GradNorm-style†	0.654	0.840	0.728	0.800	0.579	0.880	N/A	N/A	N/A	N/A	N/A	N/A
Spectral-style†	0.559	0.875	0.568	0.820	0.550	0.930	N/A	N/A	N/A	N/A	N/A	N/A
S4: Final Model *Gaussian parameters*												
SplatFormer	0.638	0.840	0.657	0.910	**0.707**	**0.740**	**0.798**	**0.730**	**0.566**	**0.860**	0.460	**0.960**
Gaussian-MAE	0.573	0.898	0.514	0.940	0.657	0.780	0.712	0.790	0.506	0.990	0.478	0.990
SceneSplat	0.514	0.920	0.582	0.840	0.520	0.920	0.489	0.900	0.497	0.970	**0.483**	0.970

† These training-stage baselines are adapted to 3DGS training dynamics rather than used as direct classifier-level detectors. We report S3 results at the same fixed training checkpoint for all evaluated attacks.
‡ Under the officially released Poison-Splat protocol, 3DGS is initialized from the sparse point cloud reconstructed from *clean* images, so it carries no attack signal and detectors are not applicable.

for example, StealthAttack may introduce artificial points along the object ray, creating abnormal local density or geometry in $\mathcal{X}$. To capture this signal, we extract per-point features from coordinates, RGB colors, and estimated normals using Sonata Wu et al. (2025) or Concerto Zhang et al. (2025), then fit an Isolation Forest Liu et al. (2008) within each scene to score local geometric irregularities.

S3: Training Dynamics. At the training stage, detectors operate on optimization diagnostics from saved 3DGS states $\theta_t = \{g_i^t\}_{i=1}^{M_t}$, where the number of Gaussians M_t changes over training due to densification and pruning. Unlike fixed-network gradient monitoring, this setting has a dynamic parameter set, no shared layer-wise embedding space across scenes, and view-dependent supervision: each training view affects only the Gaussians visible in its camera frustum. We therefore construct gradient-based baselines adapted to 3DGS dynamics. At each saved optimization state θ_t, we summarize per-Gaussian update magnitudes, such as the positional-gradient norm $|\nabla_{x_i}\mathcal{L}(\theta_t)|_2$. We then aggregate these magnitudes across the visible Gaussians within each training view, yielding a view-indexed gradient matrix that captures how strongly each view drives the optimization. From this matrix, we derive scene-level anomaly scores using two unsupervised baselines: a GradNorm-style score over heavy-tailed gradient magnitudes Huang et al. (2021), and a Spectral-style score over the leading singular component of this matrix Tran et al. (2018).

S4: Final Model. At the final-model stage, detectors operate on the optimized Gaussian representation $\mathcal{G}^\star = \{g_i\}_{i=1}^{M}$, where each Gaussian stores geometry and appearance attributes. This stage is security-critical because $\mathcal{G}^\star$ is the artifact ultimately rendered, shared, or deployed. We use SceneSplat Li et al. (2025b), SplatFormer Chen et al. (2025a), and Gaussian-MAE Ma et al. (2025) as frozen Gaussian encoders. Each final Gaussian cloud $\mathcal{G}_i^\star$ is encoded and pooled into a normalized scene embedding $\hat{z}_i$. For every base scene i, we construct a normalized clean centroid $\hat{\mu}_{(-i)}$ from all other clean scenes, excluding the clean model of scene i. The same leave-one-base-scene-out centroid is used to score both the clean model and every poisoned variant derived from scene i using cosine distance, $s_i = -\hat{z}_i^\top \hat{\mu}_{(-i)}$. This fixed protocol produces a scene-level anomaly score in Gaussian feature space, where larger values indicate greater deviation from the clean final-model distribution.

5 STAGE-SPECIFIC DETECTABILITY AND DETECTOR ALIGNMENT

We analyze Poison-3DGS from a defender-centric perspective: where poisoning becomes detectable in the 3DGS pipeline and which detectors capture the corresponding forensic signals. Every method follows the stage-specific access protocol in Section 4 and outputs one scene-level anomaly score.

5.1 STAGE-SPECIFIC DETECTABILITY ACROSS ATTACKS

Where do forensic signals become most detectable? Table 1 shows that detectability varies substantially across stages, and no single stage consistently dominates across attack families under the evaluated detectors.

StealthAttack leaves forensic signals across multiple stages. Its target-view insertion creates localized image evidence at S1, where SoftPatch+ reaches 0.837 AUROC. Its coordinated geometry injection also produces sparse-structure anomalies at S2, where Sonata reaches the strongest separation of 0.878. The attack remains detectable from training dynamics at S3, where GradNorm-style scoring reaches 0.728 AUROC. Thus, although evidence persists across the pipeline, S2 provides the most informative representation for StealthAttack under the evaluated detectors.

Poison-Splat exhibits a different detectability profile. Although it is injected at S1 through subtle image perturbations, the best image-stage detector reaches only 0.540 AUROC. Its effect becomes more visible after optimization: GradNorm-style scoring reaches 0.579 at S3, while SplatFormer reaches 0.707 at S4. By disrupting adaptive density control, Poison-Splat induces excessive Gaussian growth, making the final Gaussian representation its most informative detection stage.

Post-training attacks form a third case. Because GaussTrap, 3D-GSW and GuardSplat modify only the converged Gaussian model, their forensic signals are observable only at S4 under our protocol. However, their attack objectives produce different modification strengths. GaussTrap implants a viewpoint-targeted illusory object, inducing broad changes in the Gaussian representation. In contrast, 3D-GSW and GuardSplat embed hidden messages while preserving normal rendering quality, resulting in more constrained attribute-level modifications. GaussTrap is therefore the most detectable of the three: SplatFormer reaches 0.798 AUROC, compared with 0.712 for Gaussian-MAE and 0.489 for SceneSplat. Its stronger forensic signal is consistent with the broader modifications required for illusory-object injection.

> ***Finding 1:*** Detectability varies significantly across stages, and no single stage consistently dominates across attack families.

5.2 ATTACK INJECTION AND STRONGEST DETECTABILITY CAN OCCUR AT DIFFERENT STAGES

Does the attack injection stage determine where an attack is most detectable? In a multi-stage 3DGS pipeline, downstream processing can transform an early manipulation into later-stage forensic signals through SfM reconstruction, optimization, or final Gaussian parameterization. Consequently, restricting detection to the attack injection stage may miss the stage at which poisoning is most detectable.

Poison-Splat provides the clearest example. It is injected at S1 through image perturbations, where the best detector reaches only 0.540 AUROC. Its detectability increases downstream, reaching 0.579 at S3 and peaking at 0.707 at S4 with SplatFormer on the final Gaussian representation.

> ***Finding 2:*** Attack injection and strongest detectability are not always aligned.

5.3 DETECTORS' PERFORMANCE ACROSS STAGE-SPECIFIC 3DGS ARTIFACTS

Which detectors capture the relevant forensic signals? Table 1 shows that existing detectors transfer unevenly across stage-specific 3DGS artifacts. This unevenness reflects whether each detector is aligned with the forensic signal exposed at its stage. For StealthAttack, the strongest alignment occurs at S2, where Sonata reaches 0.878 AUROC. The attack injects target-object points

through density-guided point cloud attack, creating an abnormal geometric distribution that is captured effectively by Sonata's pretrained point-cloud representation. StealthAttack also remains detectable at S1, where SoftPatch+ reaches 0.837 AUROC by preserving evidence from localized image edits, and at S3, where the GradNorm-style detector reaches 0.728 AUROC by capturing abnormal gradient created by attack's noise scheduler.

For Poison-Splat, SplatFormer performs best among the S4 detectors, reaching 0.707 AUROC, compared with 0.657 for Gaussian-MAE and 0.520 for SceneSplat. This result suggests that SplatFormer's pretrained representation better preserves the abnormal primitive distribution produced by the attack. Its pretraining objective refines the final Gaussian representation, making it more sensitive to irregularities in the poisoned Gaussian primitives.

The same alignment principle helps explain the different outcomes for 3D-GSW and GuardSplat. SplatFormer reaches 0.566 AUROC on 3D-GSW, whereas GuardSplat remains near chance across the S4 encoders, with a best result of only 0.483 using SceneSplat. This contrast is consistent with the scope of their modifications: 3D-GSW perturbs a broader set of Gaussian parameters, potentially inducing a larger shift in holistic Gaussian representations. GuardSplat instead introduces subtle perturbations primarily to the spherical-harmonic (SH) channels while largely preserving scene geometry and other attributes. SceneSplat retains only the 3 DC SH coefficients as color features and discards the remaining 45 coefficients, although GuardSplat distributes its offsets across all 48. This limited exposure results in weak alignment with the evaluated representation.

In constrast, GaussTrap differs from the subtle Gaussian-based modification attacks in both objective and optimization scope. It performs prolonged fine-tuning using the poisoned target view together with the clean views. The target view forces the model to render a new illusory object, while the clean views constrain benign-view fidelity. Reconciling these objectives induces coordinated changes across the Gaussian representation, producing a broader forensic signal than constrained message embedding. SplatFormer captures this signal most effectively, reaching 0.798 pooled AUROC.

Finding 3: Existing detectors transfer unevenly across stage-specific 3DGS artifacts.

6 Conclusion and Discussion

This paper studies poisoning detection in 3D Gaussian Splatting from a stage-wise, defender-centric perspective. While recent attacks manipulate input views, SfM geometry, training dynamics, and final Gaussian parameters, existing evaluations mainly focus on attack success rather than where evidence becomes detectable. We introduce **Poison-3DGS**, a benchmark for characterizing poisoning detection across 3DGS pipeline stages, exposing artifacts from multi-view images, geometry, training traces, and Gaussian parameters. Based on Poison-3DGS, we conduct a study and show that detectability varies substantially across stages, no single stage consistently dominates across attack types, and attack injection and strongest detectability are often decoupled. We hope Poison-3DGS provides a foundation for future work on robust and reliable 3DGS systems.

Limitations and future directions. Poison-3DGS is built on the current public landscape of 3DGS attacks, using official or reproducible implementations whenever available. As new attacks emerge, the benchmark can be extended while preserving its core stage-wise design.

1. **Extending the attack suite.** Future extensions can incorporate new poisoning, watermarking, or model-manipulation attacks by applying them to the same clean scene pool, recording their pipeline artifacts, and evaluating them under the same unified detector protocol. This would allow new threats to be compared by both attack success and where their forensic signals emerge across the 3DGS pipeline.

2. **From detection to mitigation.** Poison-3DGS characterizes where forensic signals become detectable, but does not propose a complete mitigation method. This motivates stage-aware defenses, such as filtering suspicious views, repairing SfM geometry, regularizing abnormal densification, or auditing final Gaussian attributes before deployment.

AI USE STATEMENT

The authors used AI to review the manuscript for clarity, consistency, formatting, and the presentation of the experimental methodology. All AI-assisted suggestions were independently reviewed and revised by the authors. The authors take full responsibility for the final content, claims, and results presented in this work.

REPRODUCIBILITY AND ETHICS STATEMENT

Reproducibility. We release the evaluation code and a review subset of Poison-3DGS (25 base scenes with their clean references and one poisoned variant per attack) through the anonymized links in Appendix A; the full benchmark of 100 base scenes and 1,253 poisoned variants will be released upon acceptance. The main paper and supplementary material describe the evaluation protocols, detector configurations, and attack settings. **For source datasets whose terms restrict redistribution (e.g., ScanNet++), we do not redistribute the images or artifacts derived from them; instead we release each variant's configuration file, random seeds, attack-specific inputs that contain no scene content** (illusory-object assets and masks, injected point sets, and watermark messages), and, for bounded Poison-Splat budgets, the perturbation residuals, together with scripts that regenerate the variant from the original data obtained under the source dataset's own terms. Variants from datasets that permit redistribution are released in full, so the entire benchmark can be reproduced end to end.

Ethics. This work aims to support the development of defenses for 3DGS by characterizing where poisoning becomes detectable. Poison-3DGS is built from existing public datasets and publicly available attack implementations; we introduce no new attack and collect no data involving human subjects. Releasing poisoning artifacts carries some dual-use risk, so the benchmark is intended for defensive research in controlled, authorized environments. Watermarking methods are treated as adversarial only under our threat model of unauthorized modification. Use and redistribution of the underlying scene data remain subject to the original datasets' licenses and access conditions.

REFERENCES

Mohammad Asim, Christopher Wewer, Thomas Wimmer, Bernt Schiele, and Jan Eric Lenssen. Met3r: Measuring multi-view consistency in generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6034–6044, June 2025.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5470–5479, 2022.

Guikun Chen and Wenguan Wang. A survey on 3d gaussian splatting. *ACM Computing Surveys*, 2024.

Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21476–21485, 2024.

Yutong Chen, Marko Mihajlovic, Xiyi Chen, Yiming Wang, Sergey Prokudin, and Siyu Tang. Splatformer: Point transformer for robust 3d gaussian splatting, 2025a.

Zixuan Chen, Guangcong Wang, Jiahao Zhu, Jianhuang Lai, and Xiaohua Xie. Guardsplat: efficient and robust watermarking for 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16325–16335, 2025b.

Afshin Dehghan, Gilad Baruch, Zhuoyuan Chen, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *NeurIPS Datasets and Benchmarks*, 2 (6):16, 2021.

Chengbo Dong, Xinru Chen, Ruohan Hu, Juan Cao, and Xirong Li. Mvss-net: Multi-view multi-scale supervised networks for image manipulation detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3539–3553, 2022.

Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (ToG)*, 37(6):1–15, 2018.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Jiaxin Hong, Sixu Chen, Shuoyang Sun, Hongyao Yu, Hao Fang, Yuqi Tan, Bin Chen, Shuhan Qi, and Jiawei Li. Gausstrap: Stealthy poisoning attacks on 3d gaussian splatting for targeted scene confusion, 2025. URL https://arxiv.org/abs/2504.20829.

Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Youngdong Jang, Hyunje Park, Feng Yang, Heeju Ko, Euijin Choo, and Sangpil Kim. 3d-gsw: 3d gaussian splatting for robust watermarking. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5938–5948, 2025.

Bo-Hsu Ke, You-Zhe Xie, Yu-Lun Liu, and Wei-Chen Chiu. Stealthattack: Robust 3d gaussian splatting poisoning via density-guided illusions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27400–27411, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Yanping Li, Zhening Liu, Zijian Li, Zehong Lin, and Jun Zhang. Remedygs: Defend 3d gaussian splatting against computation cost attacks. *arXiv preprint arXiv:2511.22147*, 2025a.

Yue Li, Qi Ma, Runyi Yang, Huapeng Li, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, Martin R. Oswald, and Danda Pani Paudel. Scenesplat: Gaussian splatting-based scene understanding with vision-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4961–4972, October 2025b.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22160–22169. IEEE, 2024.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008.

Xuntao Liu, Yuzhou Yang, Haoyue Wang, Qichao Ying, Zhenxing Qian, Xinpeng Zhang, and Sheng Li. Multi-view feature extraction via tunable prompts is enough for image manipulation localization. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 9999–10007, 2024.

Jiahao Lu, Yifan Zhang, Qiuhong Shen, Xinchao Wang, and Shuicheng Yan. Poison-splat: Computation cost attack on 3d gaussian splatting. *arXiv preprint arXiv:2410.08190*, 2024.

Qi Ma, Yue Li, Bin Ren, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, and Danda Pani Paudel. A large-scale dataset of gaussian splats and their self-supervised pretraining. In *2025 International Conference on 3D Vision (3DV)*, pp. 145–155. IEEE, 2025.

Xiaochen Ma, Bo Du, Zhuohang Jiang, Xia Du, Ahmed Y. Al Hammadi, and Jizhe Zhou. Iml-vit: Benchmarking image manipulation localization by vision transformer, 2024. URL https://arxiv.org/abs/2307.14863.

Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18039–18048, 2024.

Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (ToG)*, 38(4):1–14, 2019.

Yiran Qiao, Yiren Lu, Yunlai Zhou, Rui Yang, Linlin Hou, Yu Yin, and Jing Ma. Defensesplat: Enhancing the robustness of 3d gaussian splatting via frequency-aware filtering. *arXiv preprint arXiv:2602.19323*, 2026.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Lei Su, Xiaochen Ma, Xuekang Zhu, Chaoqun Niu, Zeyu Lei, and Ji-Zhe Zhou. Can we get rid of handcrafted feature extractors? sparsevit: Nonsemantics-centered, parameter-efficient image manipulation localization through spare-coding transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 7024–7032, 2025.

Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5283–5293, June 2025.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. *Advances in neural information processing systems*, 31, 2018.

Chengjie Wang, Xi Jiang, Bin-Bin Gao, Zhenye Gan, Yong Liu, Feng Zheng, and Lizhuang Ma. Softpatch+: Fully unsupervised anomaly classification and segmentation. *Pattern Recognition*, 161: 111295, 2025a. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2024.111295. URL https://www.sciencedirect.com/science/article/pii/S003132032401046X.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306, June 2025b.

Peng Wang, Yuan Liu, Zhaoxi Chen, Lingjie Liu, Ziwei Liu, Taku Komura, Christian Theobalt, and Wenping Wang. F2-nerf: Fast neural radiance field training with free camera trajectories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4150–4159, 2023.

Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata: Self-supervised learning of reliable point representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22193–22204, June 2025.

Jingkang Yang, Pengyun Wang, Dejian Zou, Zitang Zhou, Kunyuan Ding, Wenxuan Peng, Haoqi Wang, Guangyao Chen, Bo Li, Yiyou Sun, et al. Openood: Benchmarking generalized out-of-distribution detection. *Advances in Neural Information Processing Systems*, 35:32598–32611, 2022.

Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images, 2024. URL https://arxiv.org/abs/2410.24207.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12–22. IEEE, 2023.

Hongjia Zhai, Xiyu Zhang, Boming Zhao, Hai Li, Yijia He, Zhaopeng Cui, Hujun Bao, and Guofeng Zhang. Splatloc: 3d gaussian splatting-based visual localization for augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

Yujia Zhang, Xiaoyang Wu, Yixing Lao, Chengyao Wang, Zhuotao Tian, Naiyan Wang, and Hengshuang Zhao. Concerto: Joint 2d-3d self-supervised learning emerges spatial representations. In *NeurIPS*, 2025.

Zheyuan Zhou, Le Wang, Naiyu Fang, Zili Wang, Lemiao Qiu, and Shuyou Zhang. R3d-ad: Reconstruction via diffusion for 3d anomaly detection. In *European conference on computer vision*, pp. 91–107. Springer, 2024.

SUPPLEMENTARY MATERIAL

This supplementary material provides release information, benchmark details, additional quantitative analyses, qualitative examples, and reproducibility information that complement the main paper.

Contents

A. Benchmark Release 14
A.1 Code and Data Availability 14
A.2 Benchmark Organization 14

B. Benchmark Details 15
B.1 Dataset and Scene Composition 15
B.2 Attack Families and Configuration Space 16

C. Additional Quantitative Results 17
C.1 Performance by Attack Configuration 18
C.2 Indoor and Outdoor Performance 19
C.3 Cross-stage Ensemble Baseline 19
C.4 S4 Clean-Reference Size Ablation 20

D. Additional Qualitative Results 21

A BENCHMARK RELEASE

A.1 CODE AND DATA AVAILABILITY

To support reproducibility and facilitate future research, we include both the implementation code and the Poison-3DGS benchmark with the submission. The anonymized code repository is available:

`https://anonymous.4open.science/r/Poison-3DGS-2CBC/`

The benchmark artifacts are hosted in an anonymized Hugging Face dataset repository:

`https://huggingface.co/datasets/anonymous-user-submission/Poison-3DGS`

Both repositories are prepared for double-blind review and do not contain author-identifying information. The code repository includes scripts for reproducing the stage-wise detector evaluation, including detector inference, score aggregation, and metric computation. The benchmark repository contains the clean scene references, poisoned variants, stage-specific artifacts, training checkpoints, and metadata needed to reproduce the main results reported in the paper. After the review process, we will replace the anonymized links with the final public release.

A.2 BENCHMARK ORGANIZATION

The benchmark follows a hierarchical organization that supports deterministic pairing between clean scenes and their poisoned variants. Clean artifacts are stored by source dataset and scene, whereas poisoned artifacts are grouped by attack, dataset, scene, and attack configuration:

```
Poison-3DGS/
|-- clean/
|   `-- <dataset>/<scene>/
|       |-- images/
|       |-- sparse/0/
|       `-- ply/iteration_<k>/point_cloud.ply
|
|-- stealthattack/
|-- poisonsplat/
|-- gausstrap/
|-- gsw/
`-- guardsplat/
    `-- poisoned/<dataset>/<scene>/<configuration>/
        |-- config.yaml
        |-- <attack-specific artifacts>
        `-- ply/iteration_<k>/point_cloud.ply
```

The directory names uniquely identify the source dataset, base scene, attack family, and attack configuration. Each poisoned variant contains a `config.yaml` file recording its generation parameters and, where applicable, auxiliary attack artifacts such as poisoned images, masks, triggers, or sparse point clouds. The `images/`, `sparse/`, and `ply/` directories correspond to the image, 3D-information, and Gaussian-representation artifacts exposed by the benchmark. Intermediate iterations support training-stage analysis, while the final iteration provides the converged Gaussian representation. Artifacts unchanged by an attack are referenced from the matched clean scene rather than duplicated. This structure allows evaluation scripts to retrieve stage-specific artifacts using a consistent interface across attacks.

B Benchmark Details

B.1 Dataset and Scene Composition

Poison-3DGS contains 100 clean base scenes drawn from eight established multi-view reconstruction and novel-view-synthesis datasets. The pool is balanced between 50 indoor and 50 outdoor scenes and spans object-centric, room-scale, architectural, and unbounded environments. Table 2 summarizes the dataset composition.

To characterize scene diversity beyond the indoor–outdoor split, we annotate each scene with five non-exclusive content characteristics:

- **Architecture:** buildings, rooms, walls, or other constructed structures dominate the scene.
- **Interior objects:** furniture, appliances, or densely arranged indoor objects occupy a substantial part of the scene.
- **Open terrain:** unstructured ground or water surfaces, such as roads, fields, dirt, or gravel, form a defining scene element.

Table 2: Composition of the 100 clean base scenes. **Left:** distribution across source datasets. **Right:** multi-label main scene content characteristics.

Dataset composition

Dataset	In.	Out.	All
DL3DV-10K	9	23	32
Tanks and Temples	8	13	21
Mip-NeRF 360	4	5	9
ARKitScenes	9	0	9
ScanNet++	9	0	9
LLFF	6	2	8
Free/F2-NeRF	1	6	7
Deep Blending	4	1	5
Total	**50**	**50**	**100**

Scene content

Characteristics	All	%
Architecture	61	61%
Interior objects	43	43%
Open terrain	39	39%
Standalone object	34	34%
Vegetation	30	30%

- **Standalone object:** a vehicle, machine, statue, or another isolated object is the primary subject.
- **Vegetation:** trees, grass, leaves, or other foliage occupy a substantial part of the scene.

The labels are assigned by inspecting multiple input views and are non-exclusive, so their counts do not sum to 100. They describe the content coverage of the base-scene pool and are not used for detector training or evaluation.

Benchmark design and unit of variation. Poison-3DGS separates two complementary dimensions of evaluation: *scene diversity* and *controlled attack variation*. The 100 unique base scenes are the independent content units, spanning eight datasets and balanced across 50 indoor and 50 outdoor environments. Across this diverse scene pool, 1,253 poisoned variants systematically vary attack type, strength, coverage, payload, and training configuration. These variants are controlled interventions rather than additional independent scenes. By varying attack factors while holding scene content fixed, *the benchmark enables fine-grained characterization of how different poisoning attacks affect a scene and how their forensic evidence evolves across the 3DGS pipeline.*

B.2 Attack Families and Configuration Space

Poison-3DGS follows the official implementations, configurations, and evaluation criteria of the corresponding attack papers. We verify attack effectiveness using each method's original success metric and assess its preservation of benign scene quality. For each attack, we vary its native control variables governing attack strength, coverage, or payload, as summarized below.

B.2.1 StealthAttack

StealthAttack Ke et al. (2025) inserts an illusory object into selected training views and injects corresponding points into low-density regions of the SfM point cloud. The benchmark includes 220 variants spanning 20 illusory target objects, providing variation in object appearance and scene context. We also vary the training-time noise schedule. Its initial scale, $\sigma_0 \in \{30, 100\}$, controls the starting noise magnitude, while the linear, cosine, and square-root decay functions determine how that noise evolves during optimization.

B.2.2 Poison-Splat

Poison-Splat Lu et al. (2024) perturbs training images to induce excessive Gaussian densification and increase reconstruction cost. We vary the perturbation budget over $\epsilon \in \{8, 16, 24\}/255$, where ϵ bounds the maximum pixel perturbation, and also include the official unconstrained setting. We further vary the poisoned-view ratio among 60%, 80%, and 100%. For example, a 60% ratio means that 60% of the images in the training set are perturbed, while the remaining images are kept clean. These settings cover different perturbation strengths and levels of attack coverage. The benchmark includes 264 Poison-Splat variants.

B.2.3 GaussTrap

GaussTrap Hong et al. (2025) is a post-training attack that fine-tunes the entire converged Gaussian model using both clean views and a poisoned target view. The target view supervises a viewpoint-specific illusory object, while the clean views preserve rendering quality at benign viewpoints. We evaluate Viewpoint Ensemble Stabilization (VES) with stabilization angles of $a \in \{3°, 5°, 15°\}$, together with an ablation that removes VES. We also vary the post-hoc fine-tuning duration over $e \in \{100, 500, 800, 2500\}$ epochs. The anchor configuration uses VES with $a = 5°$ and $e = 2500$; the densification cutoff is fixed at 100k iterations in all configurations. These settings control the scope and duration of the post-hoc model modification. The benchmark includes 252 GaussTrap variants across the evaluated configurations.

B.2.4 3D-GSW

3D-GSW Jang et al. (2025) fine-tunes a pretrained 3DGS model to embed a binary message in its color, opacity, rotation, and scale attributes. We vary the message length among 32, 48, and 64 bits, controlling the amount of embedded information. We also vary the embedding-loss weight over $\lambda \in \{0.2, 0.4, 0.8\}$. This weight controls the balance between message recovery and preservation of

Table 3: StealthAttack AUROC across training-time noise-schedule configurations at S3 and S4. The noise schedule is applied only during optimization; therefore, the schedule-independent S1 and S2 results are omitted from this comparison and reported separately in the main stage-wise evaluation. Bold denotes the best detector within each stage and configuration.

Method	$\sigma_0 = 100$			$\sigma_0 = 30$		
	Linear	**Cosine**	**Sqrt**	**Linear**	**Cosine**	**Sqrt**
***S3: Training**: training dynamics*						
GradNorm-style	**0.685**	**0.736**	**0.725**	**0.775**	**0.791**	**0.765**
Spectral-style	0.590	0.581	0.619	0.515	0.526	0.511
***S4: Final Model** Gaussian parameters*						
SplatFormer	**0.696**	**0.742**	**0.758**	**0.508**	**0.570**	**0.538**
Gaussian-MAE	0.523	0.528	0.521	0.485	0.508	0.494
SceneSplat	0.616	0.651	0.652	0.483	0.488	0.498

Table 4: Poison-Splat AUROC across perturbation budgets and poisoned-view ratios. Bold denotes the best result within each stage and configuration.

Method	$8/255$ **100%**	$16/255$ **60%**	$16/255$ **80%**	$16/255$ **100%**	$24/255$ **100%**	**Unconstrained 100%**
***S1: Data** multi-view RGB images*						
SoftPatch+	0.474	0.479	0.443	0.459	0.424	**0.477**
SparseViT	**0.533**	**0.632**	**0.621**	**0.554**	**0.652**	0.174
MVSS-Net	0.338	0.471	0.417	0.297	0.278	0.113
IML-ViT	0.435	0.455	0.417	0.424	0.410	0.431
***S2: 3D Information** multi-view geometry*						
MET3R	**0.528**	**0.537**	**0.536**	0.489	0.498	0.603
VGGT	0.521	0.520	0.496	0.493	0.509	0.155
MV-DUSt3R+	0.503	0.521	0.523	**0.507**	**0.518**	**0.633**
NoPoSplat	0.346	0.407	0.351	0.304	0.253	0.227
***S3: Training** training dynamics*						
GradNorm-style	**0.588**	**0.599**	**0.605**	0.538	0.566	0.667
Spectral-style	0.446	0.504	0.527	**0.543**	**0.605**	**0.718**
***S4: Final Model** Gaussian parameters*						
SplatFormer	**0.623**	**0.677**	**0.691**	**0.693**	**0.725**	**0.890**
Gaussian-MAE	0.582	0.581	0.653	0.681	0.690	0.722
SceneSplat	0.505	0.506	0.506	0.499	0.470	0.699

the original scene representation; a larger value places greater emphasis on recovering the embedded message. The benchmark includes 254 variants across the evaluated message-length and loss-weight combinations.

B.2.5 GUARDSPLAT

GuardSplat Chen et al. (2025b) embeds a binary message through learned offsets to the spherical-harmonic coefficients while preserving Gaussian geometry. We vary the message length among 16, 32, and 48 bits to evaluate different payload capacities. We also vary the message-loss weight over $\lambda_{\text{msg}} \in \{0.01, 0.03, 0.06\}$, which controls the strength of the message-embedding objective relative to scene preservation. Larger weights encourage stronger message recovery but may introduce larger changes to the SH coefficients. The benchmark includes 263 variants across the evaluated combinations.

C ADDITIONAL QUANTITATIVE RESULTS

We provide two stratified analyses that complement the aggregate results in the main paper. First, we report performance across the evaluated configurations of each attack. Second, we separate

Table 5: GaussTrap AUROC across attack configurations. Only S4 detectors are applicable because GaussTrap modifies the converged Gaussian representation. Each configuration varies one setting from the anchor ($\delta = 5°$, E_{2500}); the densification cutoff is fixed at 100k iterations. Bold denotes the best result in each configuration.

Method	$\delta = 3°$ E_{2500}	$\delta = 5°$ E_{2500}	$\delta = 15°$ E_{2500}	No VES E_{2500}	$\delta = 5°$ E_{800}	$\delta = 5°$ E_{500}	$\delta = 5°$ E_{100}
	S4: Final Model			*Gaussian parameters*			
SplatFormer	**0.983**	**0.980**	**0.981**	**0.894**	0.638	**0.681**	0.476
Gaussian-MAE	0.840	0.819	0.822	0.834	**0.700**	0.553	**0.505**
SceneSplat	0.536	0.527	0.470	0.450	0.493	0.479	0.441

Table 6: 3D-GSW AUROC across message lengths and embedding-loss weights. Only S4 detectors are applicable because 3D-GSW modifies the converged Gaussian representation. Bold denotes the best result in each configuration.

Method	32 bits $\lambda = 0.2$	32 bits $\lambda = 0.4$	32 bits $\lambda = 0.8$	48 bits $\lambda = 0.4$	64 bits $\lambda = 0.4$	64 bits $\lambda = 0.8$
	S4: Final Model		*Gaussian parameters*			
SplatFormer	**0.590**	**0.566**	**0.585**	**0.555**	**0.547**	**0.552**
Gaussian-MAE	0.493	0.529	0.497	0.482	0.498	0.496
SceneSplat	0.481	0.508	0.486	0.499	0.507	0.481

performance between indoor and outdoor scenes. Unless stated otherwise, detector configurations, aggregation rules, and score orientations remain identical to the main evaluation. We report scene-level AUROC without attack-specific detector selection or adaptation.

C.1 Performance by Attack Configuration

We evaluate every applicable detector across the official configurations included for each attack. For StealthAttack, we vary the initial noise scale and decay schedule. For Poison-Splat, we vary the perturbation budget and poisoned-view ratio. For 3D-GSW and GuardSplat, we vary the message length and embedding-loss weight. The results are reported in Tables 3–7.

StealthAttack. A larger initial noise scale generally produces a stronger forensic signal at S4. With $\sigma_0 = 100$, SplatFormer reaches 0.696–0.758 AUROC, compared with 0.508–0.570 for $\sigma_0 = 30$. SceneSplat shows a similar increase from 0.483–0.498 to 0.616–0.652. The decay schedule has a smaller effect than the initial scale.

Poison-Splat. Stronger Poison-Splat configurations are generally more detectable at the training and final-model stages. For example, SplatFormer increases from 0.623 AUROC at $8/255$ to 0.725 at $24/255$ and 0.890 under unconstrained poisoning. This suggests that stronger perturbations leave more pronounced forensic signals in the trained Gaussian representation.

The trend differs at earlier stages. Under unconstrained 100%-view poisoning, MVSS-Net, VGGT, SparseViT, and NoPoSplat reach 0.113, 0.155, 0.174, and 0.227 AUROC, respectively. We interpret these reversals as a mismatch between the detectors' learned forensic cues and Poison-Splat's perturbations. Poison-Splat targets Gaussian over-densification, whereas these detectors respond to image-manipulation or geometric-consistency cues. Stronger poisoning therefore need not amplify the cues captured by their anomaly scores. Under unconstrained poisoning, the affected detectors instead tend to rank poisoned scenes as less anomalous than clean scenes. The below-chance AUROC thus indicates a reversed forensic signal rather than an absence of separability. We retain the fixed score orientation across all configurations, as required by our attack-agnostic protocol.

GaussTrap. Fine-tuning duration is associated with substantial variation in S4 detectability. At $E = 100$, all three detectors achieve only 0.441–0.505 AUROC, whereas at $E = 2500$, SplatFormer and Gaussian-MAE reach 0.980 and 0.819, respectively. This contrast is consistent with prolonged fine-tuning over clean and target views accumulating more detectable changes in the final Gaussian representation. However, intermediate-duration results are detector-dependent: SplatFormer reaches

Table 7: GuardSplat AUROC across message lengths and message-loss weights. Only S4 detectors are applicable because GuardSplat modifies the converged Gaussian representation. Bold denotes the best result in each configuration.

Method	48 bits $\lambda_{\text{msg}} = 0.01$	48 bits $\lambda_{\text{msg}} = 0.03$	48 bits $\lambda_{\text{msg}} = 0.06$	16 bits $\lambda_{\text{msg}} = 0.03$	16 bits $\lambda_{\text{msg}} = 0.06$	32 bits $\lambda_{\text{msg}} = 0.03$
			S4: Final Model *Gaussian parameters*			
SplatFormer	0.458	0.476	0.453	0.449	0.445	0.451
Gaussian-MAE	0.449	**0.507**	0.473	**0.493**	0.465	0.470
SceneSplat	**0.478**	0.495	**0.475**	0.477	**0.477**	**0.475**

0.681 at $E = 500$ and 0.638 at $E = 800$, while Gaussian-MAE increases from 0.553 to 0.700. SceneSplat remains near chance across durations. Thus, the results support a clear contrast between short and prolonged fine-tuning, rather than a uniformly monotonic relationship. By comparison, varying the VES angle yields little change in SplatFormer AUROC (0.980–0.983).

3D-GSW. Detectability remains modest across message lengths and embedding-loss weights. SplatFormer reaches 0.547–0.590 AUROC, while Gaussian-MAE and SceneSplat remain close to chance. Neither a larger payload nor a stronger embedding loss consistently improves detection. This weak variation is consistent with 3D-GSW fine-tuning multiple Gaussian attributes while constraining the modification to preserve rendering quality.

GuardSplat. GuardSplat is consistently difficult to detect: all results lie between 0.445 and 0.507 AUROC, with no clear trend across message lengths or loss weights. GuardSplat modifies only spherical-harmonic coefficients while preserving Gaussian geometry, leaving a subtle forensic signal that current Gaussian encoders do not reliably capture.

C.2 Indoor and Outdoor Performance

We further evaluate performance separately on the 50 indoor and 50 outdoor base scenes. For each environment, clean scenes from that environment serve as negatives, while the corresponding poisoned scenes serve as positives. This environment-matched protocol limits the influence of broad indoor–outdoor content differences on the anomaly scores.

Each environment defines a separate ROC evaluation, so its AUROC is not expected to average directly to the aggregate result. The results in Table 8 examine whether the observed stage-wise detectability patterns remain consistent across the two scene environments.

C.3 Cross-stage Ensemble Baseline

We examine whether detectors operating at different stages provide complementary forensic evidence. We fuse the scene-level scores already produced by the benchmark; no detector is rerun, and no detector parameters or fusion weights are learned. For each detector d, we standardize its score using the 100-scene clean reference bank:

$$z_d(x) = \frac{s_d(x) - \mu_d^{\text{clean}}}{\sigma_d^{\text{clean}}},$$

and combine the standardized scores using an unweighted mean.

For StealthAttack, the noise scheduler does not modify the S1 images or S2 geometry. Each variant therefore inherits the S1 and S2 scores associated with its scene and injection layout. Cross-stage fusion is evaluated only for StealthAttack and Poison-Splat because the pre-S4 artifacts of 3D-GSW and GuardSplat remain unchanged under our protocol.

Fixed cross-stage panel. We select one coverage-complete detector at each stage using macro-average benchmark AUROC: SparseViT at S1, MET3R at S2, GradNorm-style at S3, and SplatFormer at S4. The same panel and equal-weight fusion rule are applied to both attacks. Because panel membership is selected from the aggregate benchmark results, this experiment is a diagnostic analysis rather than part of the primary attack-agnostic evaluation.

Table 8: Indoor and outdoor scene-level AUROC using environment-matched clean negatives. Bold denotes the best result within each stage, attack, and environment column.

Method	StealthAttack		Poison-Splat		GaussTrap		3D-GSW		GuardSplat	
	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor
S1: Data *multi-view RGB images*										
SoftPatch+	**0.785**	**0.878**	**0.500**	0.439	N/A	N/A	N/A	N/A	N/A	N/A
SparseViT	0.727	0.814	0.498	**0.581**	N/A	N/A	N/A	N/A	N/A	N/A
MVSS-Net	0.656	0.759	0.320	0.312	N/A	N/A	N/A	N/A	N/A	N/A
IML-ViT	0.666	0.747	0.404	0.457	N/A	N/A	N/A	N/A	N/A	N/A
S2: 3D Information *multi-view geometry*										
MET3R	0.704	0.778	**0.541**	0.505	N/A	N/A	N/A	N/A	N/A	N/A
VGGT	0.608	0.661	0.473	0.465	N/A	N/A	N/A	N/A	N/A	N/A
MV-DUSt3R+	0.471	0.506	0.518	**0.535**	N/A	N/A	N/A	N/A	N/A	N/A
NoPoSplat	0.547	0.497	0.263	0.362	N/A	N/A	N/A	N/A	N/A	N/A
S2: 3D Information *sparse point cloud*										
Sonata	**0.889**	**0.866**	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Concerto	0.847	0.823	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
S3 Training*: training dynamics*										
GradNorm-style	**0.763**	**0.683**	**0.571**	0.591	N/A	N/A	N/A	N/A	N/A	N/A
Spectral-style	0.505	0.643	0.515	**0.602**	N/A	N/A	N/A	N/A	N/A	N/A
S4: Final Model *Gaussian parameters*										
SplatFormer	**0.665**	**0.675**	**0.610**	**0.798**	**0.781**	0.812	0.539	**0.605**	0.468	0.464
Gaussian-MAE	0.406	0.625	0.571	0.740	0.570	**0.829**	0.446	0.567	0.471	**0.510**
SceneSplat	0.576	0.612	0.521	0.528	0.499	0.483	**0.541**	0.457	**0.543**	0.433

Table 9: Fixed cross-stage ensemble using the same detector panel for both attacks. Δ denotes the AUROC difference between the ensemble and its best individual panel member. Bold denotes the better result for each metric.

Attack	Best member	Individual		Ensemble		Δ AUROC
		AUROC↑	FPR@95↓	AUROC↑	FPR@95↓	
StealthAttack	SparseViT	0.773	0.78	**0.891**	**0.40**	$+0.118$
Poison-Splat	SplatFormer	0.707	0.74	0.665	0.84	-0.042

As shown in Table. 9, the ensemble improves StealthAttack because several stages provide partly complementary evidence. For Poison-Splat, the strongest signal is concentrated at S4, and averaging the weaker S1–S3 scores reduces performance. These results suggest that simple cross-stage fusion can help when informative signals are distributed across stages, but may dilute detection when one stage dominates.

C.4 S4 Clean-reference Size Ablation

The S4 protocol scores each scene against a centroid constructed from clean reference embeddings. We examine how many clean reference scenes are required by randomly sampling banks of size $k \in \{2, 5, 10, 20, 30, 50, 75, 100\}$. For each $k < 100$, we perform 500 seeded draws and recompute the centroid directly from the stored per-scene embeddings; no detector is rerun.

The leave-one-base-scene-out constraint is preserved in every draw: if an evaluated scene's clean base appears in its sampled bank, that base is excluded before computing the centroid. The negative set remains fixed at the 100 clean scenes, making the results comparable across k. We report the mean AUROC and empirical 5th–95th percentile interval across sampled banks. The interval measures sensitivity to reference-bank composition rather than uncertainty over evaluation scenes. As an implementation check, the full-bank evaluation reproduces every per-scene leave-one-base-scene-out score within 10^{-9} and all corresponding headline AUROCs.

Table 10: Effect of clean-reference-bank size on SplatFormer detection of Poison-Splat. For $k < 100$, results report the mean AUROC and empirical 90% interval across 500 randomly sampled clean banks. The $k = 100$ row is the deterministic full-bank result.

Clean scenes k	**Mean AUROC**	**90% interval**
2	0.666	[0.461, 0.789]
5	0.685	[0.534, 0.776]
10	0.698	[0.622, 0.763]
20	0.702	[0.652, 0.747]
30	0.704	[0.664, 0.739]
50	0.707	[0.680, 0.730]
75	0.707	[0.692, 0.721]
100	0.707	–

The mean AUROC approaches the full-bank result as k increases, while the empirical interval narrows substantially. With 10 clean references, the mean AUROC reaches 0.698, compared with 0.707 using the full bank, although the result remains sensitive to bank composition. At $k = 50$, the mean matches the full-bank AUROC and the interval narrows to $[0.680, 0.730]$. Thus, small clean banks provide competitive average performance, whereas larger banks yield more stable detection.

D Additional Qualitative Results

We provide additional qualitative examples from Poison-3DGS to illustrate the diversity of poisoning artifacts covered by the benchmark. These examples complement Fig. 2 in the main paper and show how different attack families leave visually and structurally distinct forensic signals across the 3DGS pipeline.

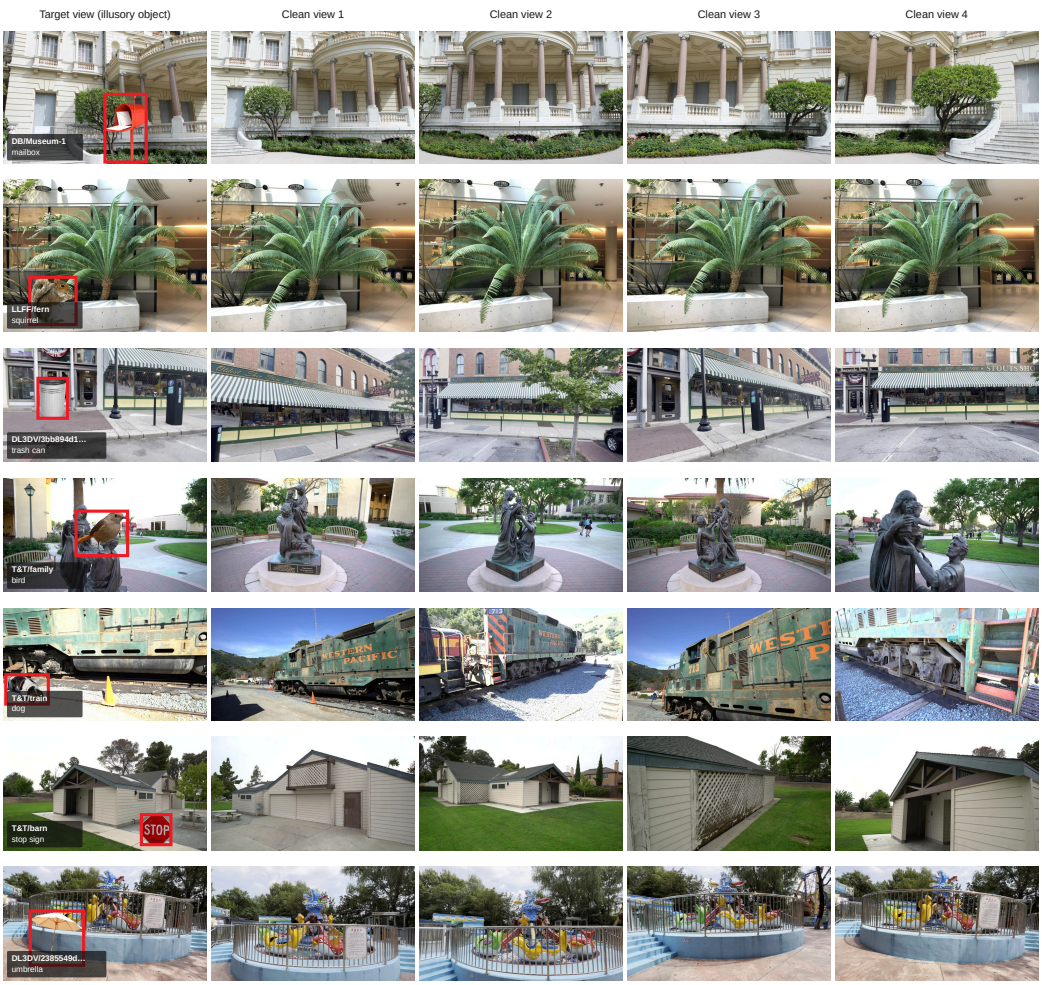

Figure 3: Additional qualitative examples of StealthAttack in Poison-3DGS. Each row shows one poisoned scene variant. The first column is the target training view containing an injected illusory object, highlighted by a red box, while the remaining columns show clean views from the same scene.



Figure 4: Additional qualitative examples of GaussTrap in Poison-3DGS. Each row shows one poisoned scene variant. The first column is the attacker's target view containing the injected illusory object, highlighted by a red box, while the remaining columns show clean training views of the same scene from other viewpoints.

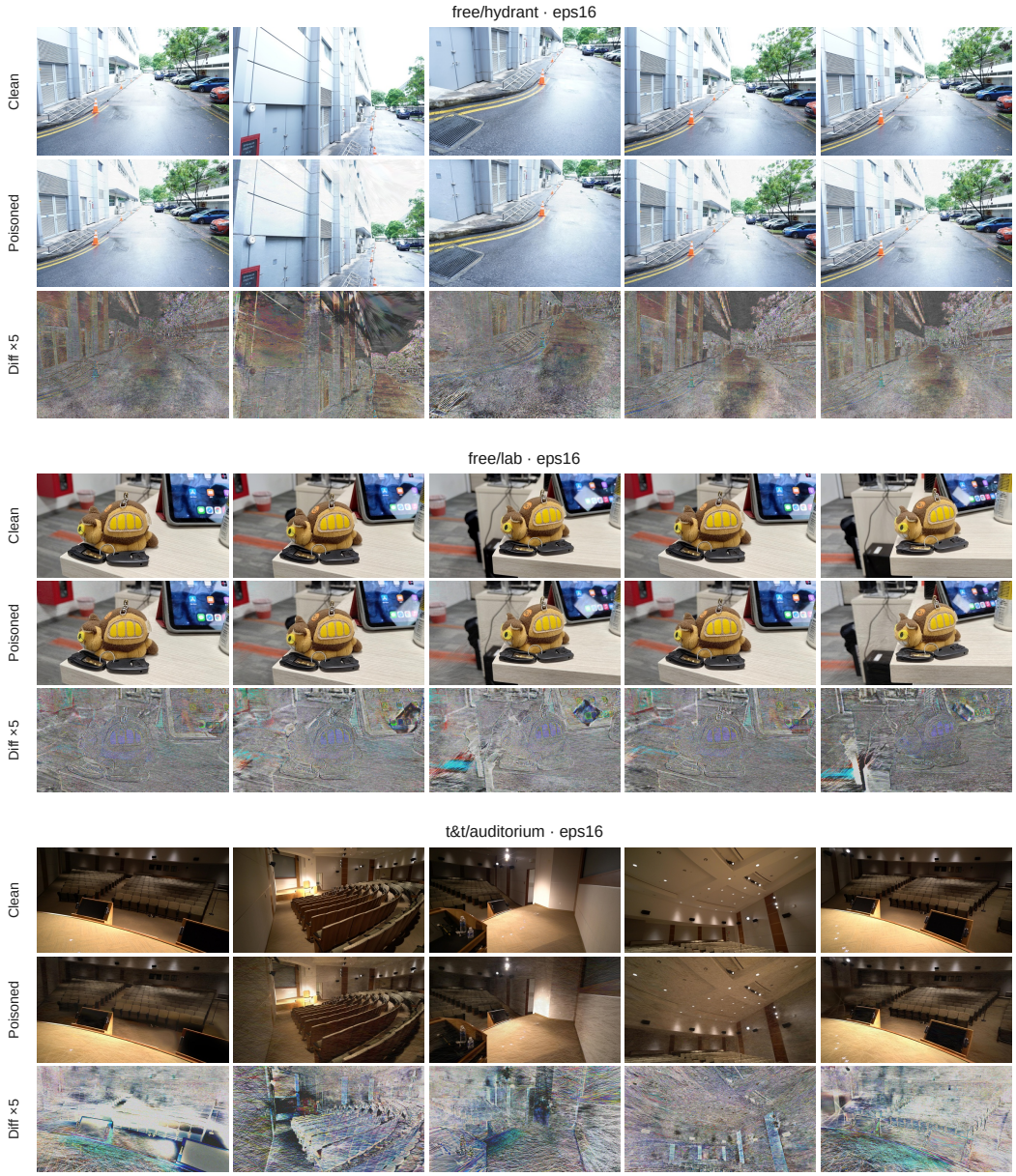

Figure 5: Additional qualitative examples of Poison-Splat in Poison-3DGS. Each block shows clean views, poisoned views, and amplified residuals for one scene. Although the perturbations are visually subtle at the image stage, the residuals reveal structured image-space changes that can be amplified during 3DGS optimization and become more detectable in later-stage artifacts.

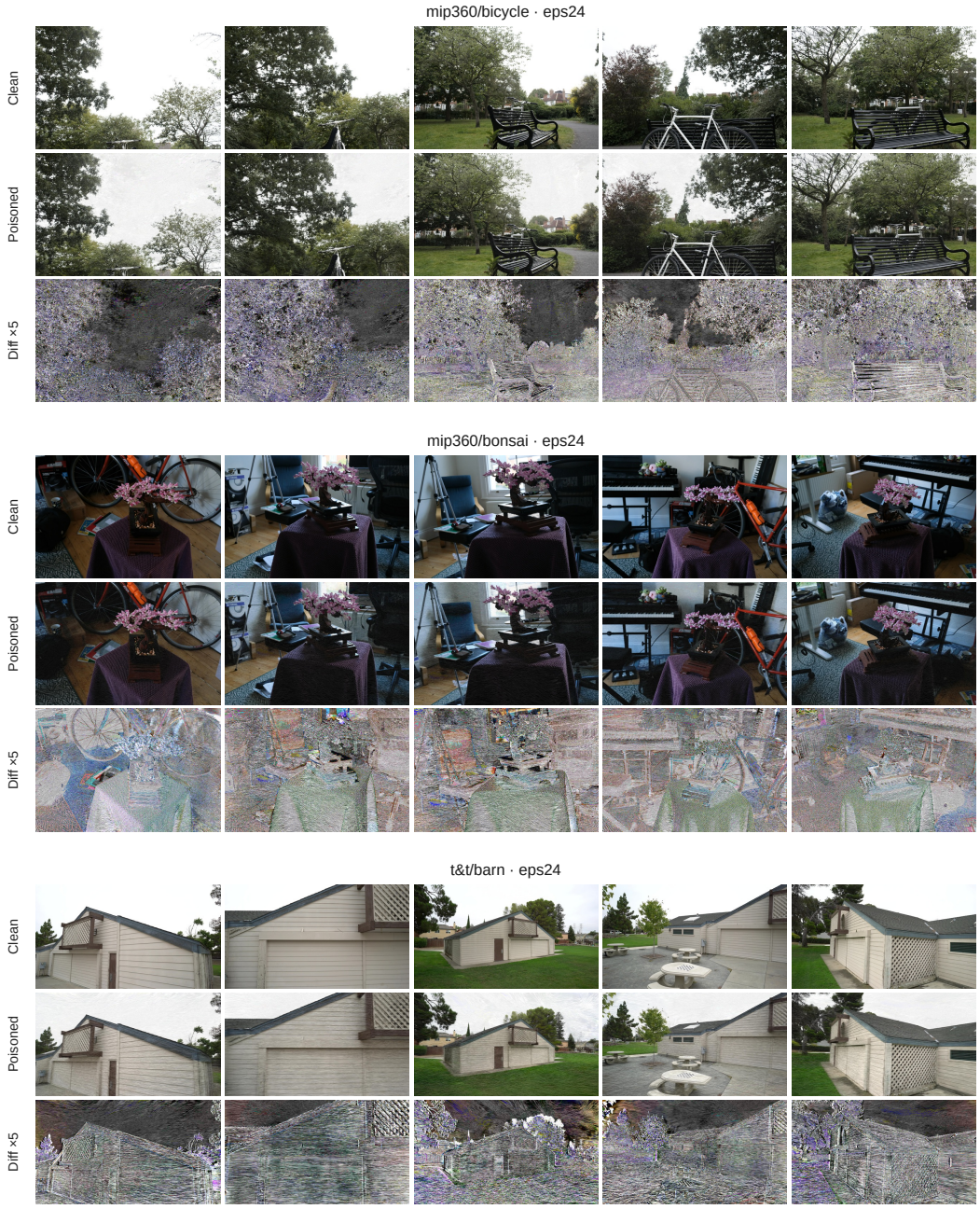

Figure 6: Additional qualitative examples of Poison-Splat in Poison-3DGS. Each block shows clean views, poisoned views, and amplified residuals for one scene. Although the perturbations are visually subtle at the image stage, the residuals reveal structured image-space changes that can be amplified during 3DGS optimization and become more detectable in later-stage artifacts.



Figure 7: Additional qualitative examples of 3D-GSW in Poison-3DGS. Each block shows clean renders, watermarked renders, and amplified residuals from five views of the same scene. The visual differences are subtle in the rendered images but become visible in the amplified residuals, illustrating final-model modifications that preserve rendering quality while encoding hidden information.

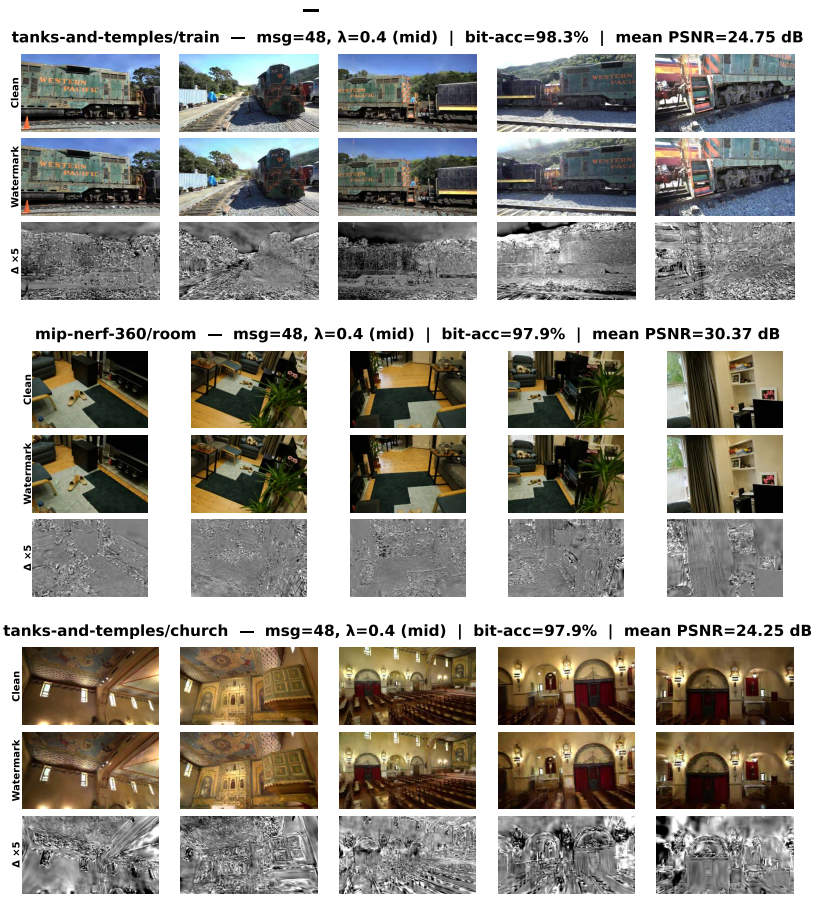

Figure 8: Additional qualitative examples of 3D-GSW in Poison-3DGS. Each block shows clean renders, watermarked renders, and amplified residuals from five views of the same scene. The visual differences are subtle in the rendered images but become visible in the amplified residuals, illustrating final-model modifications that preserve rendering quality while encoding hidden information.

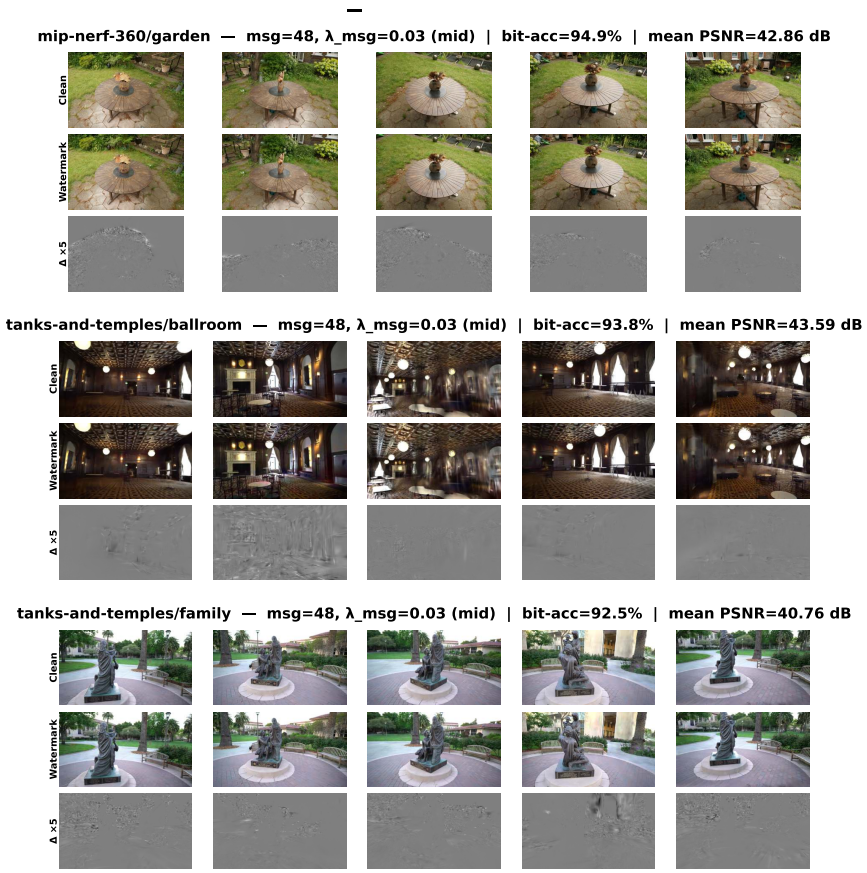

Figure 9: Additional qualitative examples of GuardSplat in Poison-3DGS. Each block shows clean renders, watermarked renders, and amplified residuals from five views of the same scene. The rendered images remain visually close to the clean references, while the amplified residuals reveal subtle final-model modifications introduced by message embedding in the Gaussian representation.

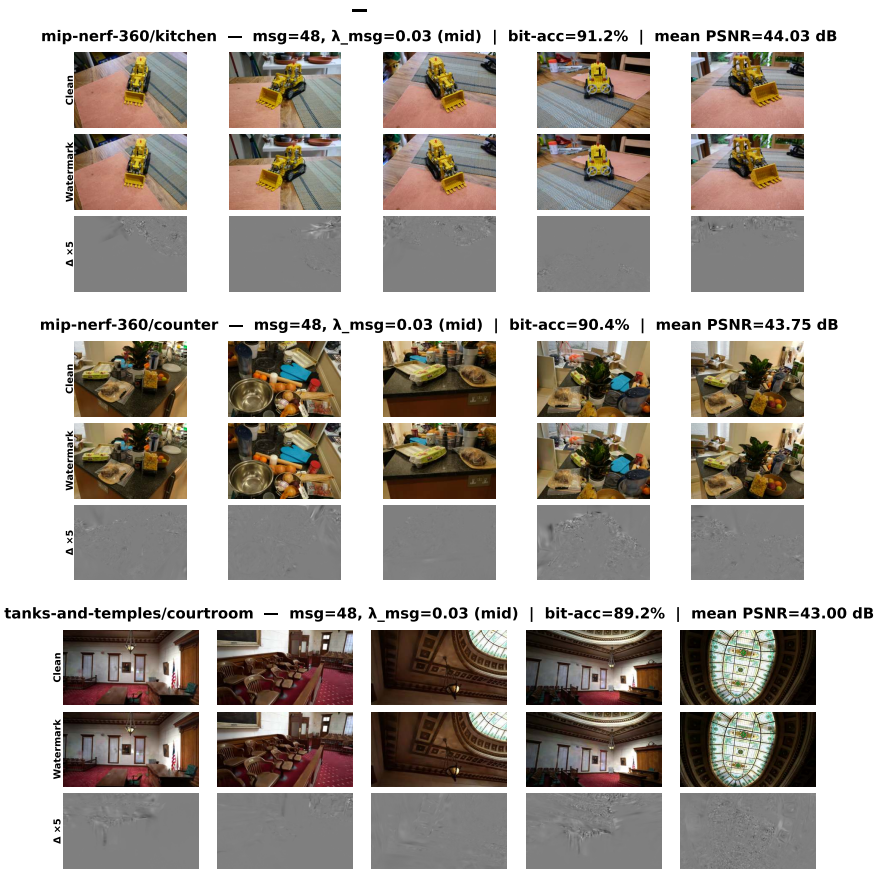

Figure 10: Additional qualitative examples of GuardSplat in Poison-3DGS. Each block shows clean renders, watermarked renders, and amplified residuals from five views of the same scene. The rendered images remain visually close to the clean references, while the amplified residuals reveal subtle final-model modifications introduced by message embedding in the Gaussian representation.