SHIELD: WATERMARKING 4D GAUSSIAN SPLATTING VIA SPHERICAL HARMONIC COEFFICIENT PERTURBATION

Anonymous authors
Paper under double-blind review

ABSTRACT

4D Gaussian Splatting (4DGS) models are computationally expensive to train but distributed as easily copied parameter files. Existing watermarks protect these assets by fine-tuning the scene against a neural decoder to extract bit strings from rendered video, a heavy process vulnerable to the temporal dynamics of dynamic scenes. However, if the threat model targets the redistributed file itself, verification can be vastly simplified. Because the canonical Spherical-Harmonic (SH) coefficients of 4DGS are time- and view-invariant, parameter-space perturbations can be recovered via deterministic arithmetic. We propose **SHield**, the first non-blind, decoder-free watermarking approach that natively embeds full-resolution RGB images into 4DGS files. SHield tiles a zero-mean payload across the SH coefficients of a key-selected Gaussian subset, applying a brief anchored optimization to preserve visual fidelity. To verify ownership, it geometrically re-associates the key using canonical Gaussian centres to survive file tampering and recovers the payload in closed form. Across 30 scenes (D-NeRF, HyperNeRF, DyNeRF), SHield guarantees near-perfect payload recovery (Pearson $r \geq 0.991$) on all 150 tested models with a mean rendering penalty of ≤ 0.13 dB. Held-out evaluation yields perfect detection on 400 true trials and zero false positives across $1{,}600$ wrong-subset null trials. We also strictly bound the method's robustness: SHield survives INT8 quantization and 30% opacity pruning for the image payload, and on aggressive SH-degree-1 truncation bit payloads are still decodable.

1 INTRODUCTION

4D Gaussian Splatting (4DGS) (Wu et al., 2024; Yang et al., 2024) achieves real-time dynamic scene rendering by displacing explicit Gaussians via a learned deformation field (Kerbl et al., 2023). Because training these models requires immense data and compute, the resulting parameter files are valuable intellectual property. However, they are also trivially easy to copy and redistribute, creating an urgent need for robust ownership verification.

Most Gaussian splatting watermarks operate in the render domain: they fine-tune the scene against a learned decoder until a short bit string survives the rendering process (Chen et al., 2025; Jang et al., 2025; Huang et al., 2024). In 4DGS, this introduces a severe temporal challenge. A perturbation that is invisible at one timestamp may become glaringly obvious at the next timestamp as the Gaussians move, this phenomenon is known as "FVD collapse" (Zhou et al., 2026). Existing methods counter this with heavy, motion-specific machinery. For example, Mark4D (Lee et al., 2026) uses trajectory-aligned offsets and a video-text decoder, while 4D-GSW relies on curvature-based gating. This forces thousands of fine-tuning iterations (e.g., $4{,}000$ steps for Mark4D) just to embed a few dozen bits. While this complexity is necessary if the watermark must be extracted from a rendered video, it is vastly over-engineered if the adversary is simply redistributing the underlying model file.

Our approach leverages the fundamental architecture of 4DGS appearance storage. Each Gaussian maintains a single, canonical set of spherical-harmonic (SH) coefficients that the deformation field queries across all timestamps and viewpoints. By writing a perturbation directly into these canonical coefficients, we create a time-invariant watermark. Shifting the verification target from the rendered

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.

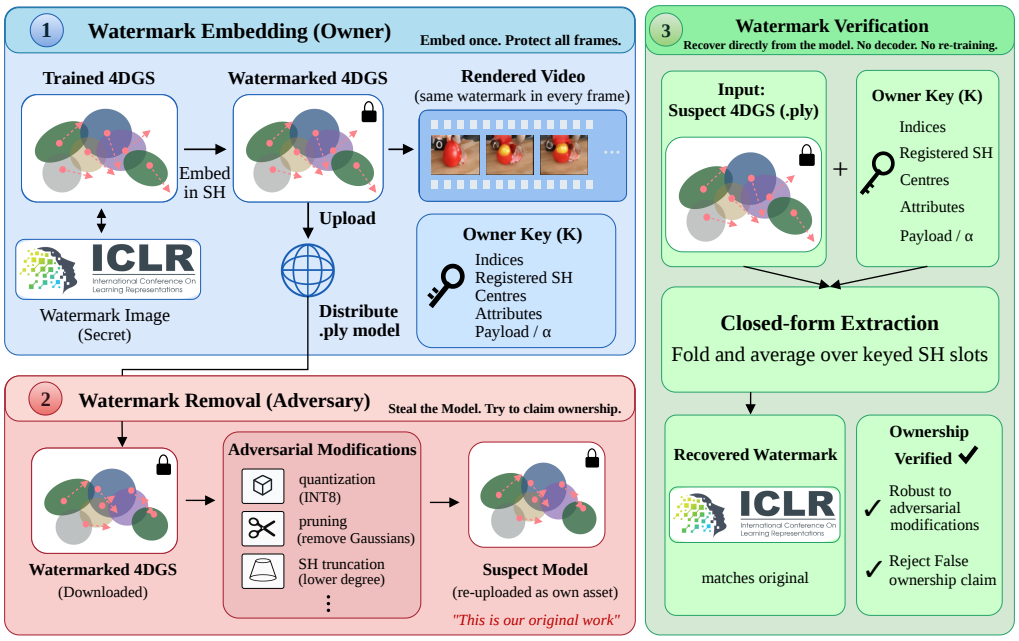

Figure 1: **File-centric ownership verification with SHield.** The owner embeds a payload into the SH coefficients of key-selected Gaussians. Under adversarial file degradation, verification geometrically re-associates the key via canonical centres, enabling closed-form extraction directly from the parameters without learned decoders or rendering loops.

video back to the redistributed file itself eliminates the need for learned decoders and rendering loops, allowing extraction via deterministic arithmetic while completely sidestepping temporal fragility.

We introduce **SHield** (Figure 1), the first decoder-free watermarking framework that natively embeds full-resolution RGB images into 4DGS parameter files and recovers them in closed form. SHield operates as a non-blind watermark, meaning verification requires the owner's private key rather than the full original `.ply` file. The embedding process tiles a zero-mean payload across the SH coefficients of a key-selected subset of Gaussians. Because naive tiling ignores whether a Gaussian is actually visible, SHield employs a brief T-step anchored optimization. This step adapts the perturbation to minimize visual rendering error while tightly anchoring each coefficient to its target value. During verification, the algorithm geometrically re-associates the owner's key with the suspect file using canonical centres (resolving ties via unchanged attributes), subtracts the stored reference coefficients, and averages the redundant copies. Crucially, SHield modifies only the SH coefficients; the scene geometry and deformation network remain strictly frozen. Our main contributions can be summarized as follows:

- **First full-resolution image watermark for 4DGS.** To our knowledge, SHield is the first method to embed and recover a full-resolution RGB image directly from a 4DGS model file. Prior dynamic scene watermarks rely on learned decoders without reporting image resolution (Liu et al., 2025), and full-resolution steganography has previously been restricted to static 3DGS (Meng et al., 2026).
- **Decoder-free, non-blind verification.** Extraction is reduced to closed-form arithmetic on canonical SH coefficients. By introducing centre-based re-association and a coprime stride, the watermark survives adversarial row reordering, opacity pruning, and SH-degree truncation.
- **Anchored optimization.** We propose a brief optimization strategy that adapts the tiled perturbation to the renderer's sensitivity. This reduces the visual footprint of plain tiling by $1.6\times$, $5.6\times$, and $3.0\times$ across the three benchmark datasets, trading a modest drop in WM-PSNR for significantly higher rendering fidelity.
- **Rigorous robustness evaluation.** We evaluate SHield across 150 models from 30 scenes (Pumarola et al., 2021; Park et al., 2021; Li et al., 2022). We thoroughly map both its capabilities and

operational limits, testing against non-adaptive post-processing (e.g., pruning, quantization) as well as adaptive attacks including targeted fine-tuning, duplication, and rigid transformations.

2 Related Work

Image and video watermarking. Classical spread-spectrum watermarking redundantly embeds messages across host coefficients, verifying authenticity by correlating the recovered signal against a registered reference. While blind detectors operate solely on suspect content, informed (non-blind) detectors leverage auxiliary owner-side data (Cox et al., 2007). Modern deep watermarking subsequently replaced hand-crafted embeddings with neural encoder–decoder pairs trained against synthetic distortions for images (Zhu et al., 2018; Tancik et al., 2020; Ma et al., 2022; Fernandez et al., 2022) and video (Zhang et al., 2019; Fernandez et al., 2024). SHield adapts this classical informed-correlation approach, shifting the host domain from pixel space to the internal parameter file of a 4DGS asset.

Radiance fields and static 3DGS. Neural asset protection initially targeted continuous coordinate fields, where CopyRNeRF (Luo et al., 2023) and StegaNeRF (Li et al., 2023) fine-tune Neural Radiance Fields to allow message recovery from rendered novel views. A similar render-centric paradigm dominates explicit 3D Gaussian Splatting: GuardSplat (Chen et al., 2025) optimizes spherical-harmonic (SH) offsets against a CLIP-guided decoder, 3D-GSW (Jang et al., 2025) incorporates frequency-guided densification, and GaussianMarker (Huang et al., 2024) anchors watermarks using per-Gaussian uncertainty. To withstand compression, CompMarkGS (In et al., 2025) and Robust3DGSW (Wang et al., 2026) integrate quantization-aware training into the pipeline. Broader steganographic frameworks extend these ideas to hide complex content, ranging from GS-Hider's (Zhang et al., 2024) coupled feature substitution and KeySS's (Ren et al., 2026) optimized key-secured decoders to Splats in Splats (Guo et al., 2026b;a) encrypting secondary scenes into SH attributes. Notably, WhisperSplat (Meng et al., 2026) avoids model modification entirely, learning an SH noise key that renders a secret high-resolution image from a single viewpoint without an explicit decoder.

Dynamic Gaussians. Adapting watermarks to 4DGS requires marks to survive continuous motion. Mark4D (Lee et al., 2026) learns time-dependent offsets decoded via video encoders, while 4D-GSW (Zhou et al., 2026) gates optimization gradients by trajectory curvature. Guard4D (Shi et al., 2026) also utilizes SH coefficients with frozen geometry, yet still relies on text-supervised decoders operating on rendered clips. Hide-in-Motion (Liu et al., 2025) embeds multimodal payloads into composite attributes but requires a private U-Net decoder and omits post-processing evaluation. Splats in Splats++ Guo et al. (2026a) similarly depends on a render-and-recover framework.

Where SHield differs. While prior approaches bind verification to rendering loops or neural decoders, SHield treats the redistributed model file itself as the verification object. Eliminating this overhead enables full-resolution image recovery via direct, closed-form arithmetic. This file-centric reliability requires a non-blind design where the owner matches a registered private key.

3 Method

3.1 Background and Threat Model

3DGS (Kerbl et al., 2023) parameterizes a scene using N Gaussians $\mathcal{G} = \{(\boldsymbol{\mu}_i, \mathbf{q}_i, \mathbf{s}_i, o_i, \mathbf{h}_i)\}_{i=1}^{N}$, defined by their centre, rotation, scale, opacity, and canonical spherical harmonic (SH) coefficients $\mathbf{h}_i \in \mathbb{R}^{C_{\text{tot}} \times 3}$ (C_{tot}=16 for degree 3). SHield exclusively modifies the SH coefficients, explicitly preserving the geometric and opacity attributes $\mathbf{a}_i = (o_i, \mathbf{q}_i, \mathbf{s}_i)$ to serve as invariant geometric anchors for key re-association. In 4DGS (Wu et al., 2024), a learned deformation network predicts time-dependent geometric offsets. Crucially, the stored SH coefficients $\mathbf{h}_i$ remain canonical and time-invariant; even in architectures like DyNeRF that inject time-dependent SH offsets during rendering, the underlying file parameters remain completely unmodified.

The owner embeds an RGB image $\mathbf{W} \in [0,1]^{H_0 \times W_0 \times 3}$ or an L-bit message $\mathbf{m} \in \{0,1\}^L$ into a trained 4DGS model, distributes the watermarked file, and retains a private key $\mathcal{K}$. For non-

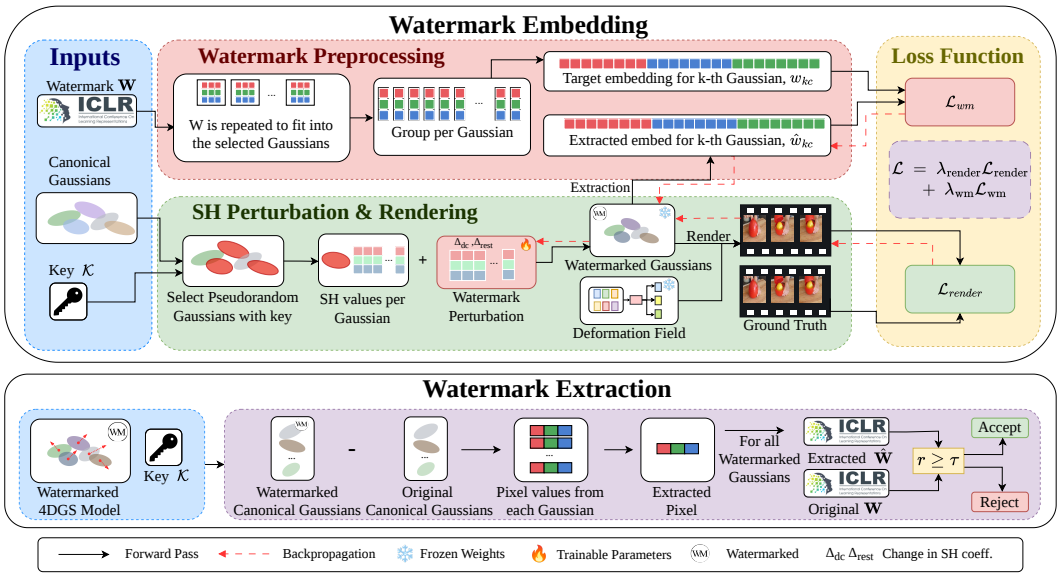

Figure 2: **Overview of SHield.** *(a) Embedding:* A zero-mean payload is tiled across the SH slots of key-selected Gaussians (red). Learnable offsets are applied to these SH coefficients while the scene geometry, unselected Gaussians, and deformation network remain strictly frozen (green). A joint loss anchors the offsets to the tiled target while preserving rendering fidelity (yellow). *(b) Verification:* Centre-based re-association locates the target Gaussians even if the file is reordered or pruned. Subtracting the stored reference SH cleanly isolates the perturbation, and the redundant copies are averaged to reconstruct the payload $\hat{\mathbf{W}}$. Ownership is verified if the correlation $r \geq \tau$.

blind extraction, $\mathcal{K}$ stores the selected Gaussian indices, original SH coefficients, canonical centres, static tie-break attributes, and payload metadata. Ownership is verified via Pearson correlation $r(\hat{\mathbf{W}}, \mathbf{W}) \geq \tau$ for images, or direct decoding accuracy for bit payloads. We assume the owner registers $\mathcal{K}$ and the payload prior to redistribution to establish provenance.

We assume a white-box adversary holding the `.ply` file who can modify any parameter and knows the algorithm, but lacks the private key $\mathcal{K}$ (Kerckhoffs' principle (Kerckhoffs, 1883)). We evaluate non-adaptive post-processing (e.g., SH quantization, reordering, pruning, truncation, and noise) alongside adaptive attacks targeting model re-optimization or the key matcher (§ 4.4). We assume the redistributed file retains its float32 canonical centres in the original coordinate frame.

3.2 Watermark Embedding

SHield embeds the watermark into the canonical SH coefficients (Figure 2). This provides two advantages: a single time-invariant perturbation propagates to all timestamps, and SH coefficients offer a robust carrier with minimal visual sensitivity. At matched relative strength, modifying spatial positions degrades rendering PSNR by $\sim$50 dB more than modifying higher-order SH bands (Appendix D). The mathematical protocol is detailed below (full pseudocode in Algorithm 2).

Tiled SH embedding. The key $\mathcal{K}$ pseudorandomly selects an ordered subset of K Gaussians $\mathcal{S} = (k_1 < \cdots < k_K) \subset [N]$. K scales to ensure at least $n_t^{\min}$ complete payload copies (tiles), strictly bounded between population fractions $\rho_{\min}$ and $\rho_{\max}$:

$$K = \min\Big(\max\big(\lfloor \rho_{\min} N \rfloor,\ \lceil n_t^{\min} n_{\text{wm}} / C_{\text{tot}} \rceil\big),\ \lfloor \rho_{\max} N \rfloor\Big), \tag{1}$$

where $n_{\text{wm}} = H_0 W_0$ and $n_t = \lfloor K C_{\text{tot}} / n_{\text{wm}} \rfloor$ is the number of complete tiles. For each channel c, the zero-meaned payload $\mathbf{z}_c$ (storing mean $\bar{w}_c$ in $\mathcal{K}$) is tiled across the KC_{tot} slots to form the target perturbation matrix $\mathbf{E}_c$. Coefficients are initialized as $\mathbf{h}^{\text{wm}}_{k_j,c} = \mathbf{h}_{k_j,c} + \alpha_0\, \mathbf{E}_c[j,:]$.

For L-bit messages, the payload is whitened via a key-derived sequence and mapped to ± 1. Because C_{tot}=16 cleanly divides standard message lengths ($L \in \{32, 48, 64\}$), plain tiling would destructively

concentrate all copies of a specific bit onto a single SH coefficient index. To prevent complete message erasure during SH-degree truncation, SHield shifts each tile t by $t\sigma$, where the stride σ is strictly coprime to both L and C_{tot} (Appendix E).

Reducing the rendering cost. Plain tiling applies a uniform amplitude regardless of a Gaussian's visibility; thus, a large α_0 degrades visual quality while a small α_0 risks quantization vulnerability. SHield resolves this by fixing α_0 at a robust level and optimizing learnable SH offsets (initialized at $\alpha_0\mathbf{E}$) for T steps across n_v training views via the frozen deformation network, minimizing:

$$\mathcal{L} = \lambda_{\text{render}}\,\mathcal{L}_{\text{render}} + \lambda_{\text{wm}}\,\mathcal{L}_{\text{wm}}, \tag{2}$$

$$\mathcal{L}_{\text{render}} = \ell_1(\hat{I}, I) + \lambda_{\text{ssim}}(1 - \text{SSIM}(\hat{I}, I)), \tag{3}$$

$$\mathcal{L}_{\text{wm}} = \tfrac{1}{3}\textstyle\sum_c \text{MSE}\big(\mathbf{e}_c[1{:}n_t n_{\text{wm}}],\ \alpha_0 \cdot \text{tile}(\mathbf{z}_c,\ n_t)\big), \tag{4}$$

where $\hat{I}$ and I are the watermarked and ground-truth renders. $\mathcal{L}_{\text{render}}$ adapts offsets to the scene's topology, while $\mathcal{L}_{\text{wm}}$ anchors the $n_t n_{\text{wm}}$ active slots to their tiled targets. This minor departure from plain tiling (averaging just 1.8% on D-NeRF) drastically improves visual fidelity.

Algorithm 1 Verification (images)

Input: suspect file (centres $\tilde{\boldsymbol{\mu}}$, attributes $\tilde{\mathbf{a}}$, SH $\tilde{\mathbf{h}}$); key $\mathcal{K}$; τ
Output: decision, confidence r

▷ *Phase 1: Geometric Re-association*

$\pi \leftarrow \textsc{ReAssociate}(\boldsymbol{\mu}[\mathcal{S}], \mathbf{a}[\mathcal{S}], \tilde{\boldsymbol{\mu}}, \tilde{\mathbf{a}})$
$\mathbf{E}_k \leftarrow (\tilde{\mathbf{h}}_{\pi_k} - \mathbf{h}_{\text{orig}}[\mathcal{S}]_k)/\alpha_0$ ▷ 0 if $\pi_k{=}\varnothing$

▷ *Phase 2: Fold-and-Average*

for $c \in \{R, G, B\}$ **do**
  $\mathbf{e}_c^{(t)} \leftarrow$ tile t of $\textsc{Flatten}(\mathbf{E}[:, :, c])$
  $\hat{\mathbf{z}}_c \leftarrow \frac{1}{n_t}\sum_t \textsc{Rot}\big(\mathbf{e}_c^{(t)}, +t\sigma\big)$ ▷ Eq. 5; undoes the $-t\sigma$ embedding shift, $\sigma{=}0$ for images
  $\hat{\mathbf{W}}_c \leftarrow \text{clip}(\hat{\mathbf{z}}_c + \overline{\mathbf{W}_c},\ 0,\ 1)$
end for

▷ *Phase 3: Statistical Verification*

$r \leftarrow \text{Pearson}(\hat{\mathbf{W}}, \mathbf{W})$
return $r \geq \tau$ ▷ $\tau{=}0.6$

3.3 Verification Interface

Verification proceeds completely in the parameter space without requiring any rendering loops or learned decoders (Algorithm 1).

Key re-association. Because adversarial redistribution often involves Gaussian pruning or row permutation (Figure 2b), the verification key cannot safely rely on static matrix indices. Instead, SHield geometrically re-associates each key entry by matching it one-to-one with a suspect Gaussian whose canonical centre falls within a tight tolerance ($\epsilon = 0.01\, s_{\text{nn}}$), where s_{nn} is the original model's median nearest-neighbour spacing. If multiple candidates fall within ϵ, the static attributes (opacity, scale, and rotation) act as strict tie-breakers. Any entry lacking a unique, valid candidate is securely treated as missing rather than guessed, preventing geometric noise from corrupting the extraction array.

Fold-and-average extraction. Following re-association, the perturbation for key entry k is isolated via $\mathbf{E}_c^{\text{rec}}[k] = (\tilde{\mathbf{h}}_{\pi_k, c} - \mathbf{h}_{\mathcal{S}_k, c})/\alpha_0$ (where π_k is the matched Gaussian), then flattened and averaged across all n_t tiles:

$$\hat{\mathbf{z}}_c = \frac{1}{n_t}\sum_{j=0}^{n_t-1} \mathbf{e}_c[j\, n_{\text{wm}} : (j{+}1)\, n_{\text{wm}}]. \tag{5}$$

The payload is reconstructed as $\hat{\mathbf{w}}_c = \text{clip}(\hat{\mathbf{z}}_c + \bar{w}_c, 0, 1)$ (bit messages are first inverse-shifted by $t\sigma$ and un-whitened). Because subtraction exactly cancels the scene coefficients, errors on clean files stem only from optimizer departures, clipping, and 8-bit rounding. Unmatched or pruned Gaussians contribute zero, gracefully attenuating the payload rather than corrupting the data stream.

Table 1: **Rendering cost and extraction** (α_0=0.1, five payloads per scene, 20 test views). Δ values are against the unwatermarked model, with the worst scene bracketed; min r is the worst payload recovery over models. The comparison against plain tiling at the same α_0 and selection is in Appendix H.

Dataset (models)	$\text{PSNR}_{\text{clean}}$	ΔPSNR [worst]	SSIM (Δ)	LPIPS (Δ)	WM-PSNR	min r
D-NeRF (35) (Pumarola et al., 2021)	35.31	−0.048 [−0.113]	0.9842 (−0.0004)	0.0164 (+0.0002)	44.77	0.9990
HyperNeRF (85) (Park et al., 2021)	26.92	−0.047 [−0.150]	0.7761 (−0.0012)	0.2949 (+0.0009)	36.27	0.9910
DyNeRF (30) (Li et al., 2022)	31.19	−0.127 [−0.314]	0.9353 (−0.0011)	0.1099 (+0.0012)	36.59	0.9952

4 Experiments

4.1 Experimental Setup

We evaluate standard 4DGS (Wu et al., 2024) across 30 scenes (7 D-NeRF, 17 HyperNeRF, 6 DyNeRF). Embedding five RGB payloads (50k–217k pixels) yields 150 watermarked models. Rendering quality (PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018)) is measured against ground truth on 20 test views, and extraction via WM-PSNR and Pearson r. Because parameter-space extraction shares no protocol with render-domain methods, our primary baseline is SHield's plain-tiling initialization (at matched amplitude and selection) to strictly isolate the impact of anchored optimization. Implementation details are in Appendix A.

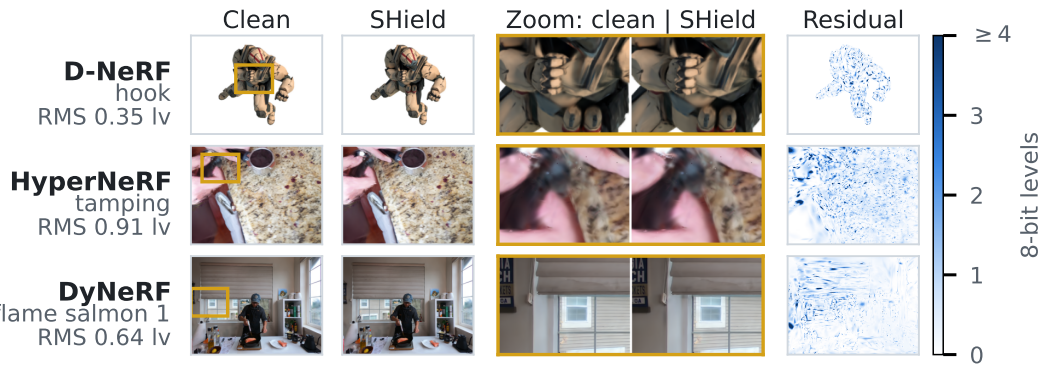

Figure 3: **Rendering residual** (α_0=0.1, wm1): each dataset's median-residual scene at its median test view, to avoid cherry-picking; per-pixel RMS difference to the clean render in 8-bit levels (lv), one shared scale, zoom box at the largest local residual.

4.2 Rendering Quality and Extraction

Mean rendering penalties are just 0.048, 0.047, and 0.127 dB on D-NeRF, HyperNeRF, and DyNeRF, with no scene degrading > 0.314 dB and extraction remaining near-perfect ($r \geq 0.991$) across all 150 models (Table 1). Compared to plain tiling, anchored optimization reduces rendering costs by 1.6×, 5.6×, and 3.0× (improving 149/150 pairs). This trades raw extraction fidelity for visual quality (e.g., DyNeRF WM-PSNR drops from 56.2 to 36.6 dB), yet correlation remains exceptionally robust ($r \geq 0.995$).

For wm1, the per-view RMS residual across 830 test views is ≤ 1.85 8-bit levels and spatially diffuse (Figure 3). The SH distribution shifts by a 3D-Sinkhorn distance (Ren et al., 2026) of < 0.02, near the 0.008 quantization floor. Parameter-space operation guarantees view-independent extraction: while a 2D baseline applied to training frames fails on novel views ($r = 0.004$), SHield reliably reads $r = 0.998$ directly from the file (Appendix M).

Table 2: **Post-processing of the model file** (α_0=0.1, all 150 image-payload models): mean WM-PSNR, mean r and passes at τ=0.6.

Opacity pruning				SH quantization (whole file)			
	WM-PSNR	r	$r \geq \tau$		WM-PSNR	r	$r \geq \tau$
0 %	38.32	0.998	150	FP32/16	38.32	0.998	150
10 %	23.38	0.965	150	INT12	38.25	0.998	150
30 %	18.21	0.892	150	INT10	37.35	0.998	150
50 %	15.38	0.790	150	INT8	32.08	0.994	150
80 %	12.70	0.544	52				

4.3 Verification and Key Re-association

Held-out verification. $\tau = 0.6$ was chosen by an earlier in-sample analysis of all 150 models and then evaluated unchanged on a scene-disjoint split of 14 calibration and 16 held-out scenes. On the held-out scenes (80 models) SHield accepted all 640 true trials – the distributed file, whole-file INT8, 30% opacity pruning, and 400 fresh trials under 30% random deletion, row permutation and key re-association (minimum $r = 0.815$) – with no false positive in 2,320 null trials of three kinds: another owner's payload on the same scene (320, maximum $r = 0.317$), a wrong subset of Gaussians (1,920, maximum 0.498) and an unmarked model (80, maximum 0.368; undefined r counts as a non-detection). Nominal one-sided 95% Clopper–Pearson bounds, treating models or scenes as independent, are 3.68% and 17.1%; the independence is imperfect, so these are not deployment-level rates (Appendix C). The in-sample choice of τ does not carry the result: every decision is identical for any $\tau \in (0.498, 0.815]$, and a threshold from the calibration scenes alone (0.657) reproduces them exactly. A centred statistic, which removes the payload-dependent residual of Figure 4, makes the same held-out decisions at a calibration-only threshold ($\tau_c = 0.5834$; 240/240 accepted, no false positive in 720) but does not repair the SH-truncation behaviour below, so we keep the uncentred one.

Re-association. Across all 192 models, re-association reproduces extraction exactly on unmodified and permuted files, and after 30% lowest-opacity or 20% random deletion every surviving marked Gaussian finds its correct row (0.008% and 0.116% of removed entries falsely matched). Geometric noise is the weak point: all marked Gaussians match under isotropic centre noise up to 0.001 s_{nn} but only 19.9% at 0.01 s_{nn}, where image verification fails on 103 of 150 models, and a float16 cast leaves 3.75% matched and fails 135 of 150, though bit payloads still decode (Appendix B).

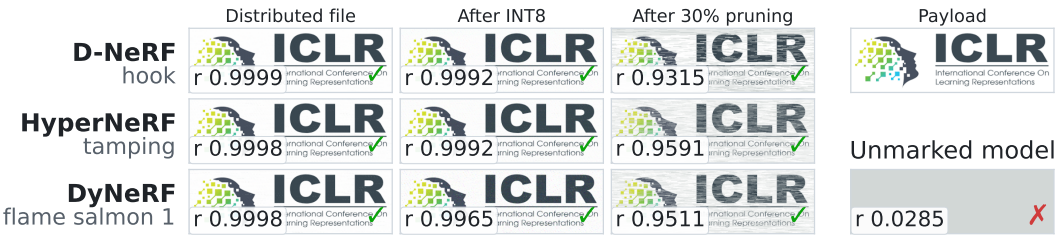

Figure 4: **Payloads recovered from the model file**, for the scenes of Figure 3: as distributed, after whole-file SH INT8, and after 30% opacity pruning, with Pearson r against τ=0.6. On an unmarked model the recovery collapses to a flat block of the payload's per-channel mean colour, whose residual correlation is payload-dependent.

4.4 Robustness to Post-processing

Non-adaptive post-processing. FP16 quantization is lossless for extraction, and after whole-file SH INT8 (every Gaussian's SH tensors, per-tensor min–max, geometry untouched) all 150 models verify with mean $r = 0.994$ (Table 2, Figure 4); quantizing the centres as well is a different attack, and the float16 cast above shows that SHield does not survive it. Removing the lowest-opacity Gaussians degrades WM-PSNR sharply but r gradually: all models pass τ at 50% pruning and 52 of 150 at

Table 3: **Adaptive fine-tuning attack.** ΔPSNR: quality on *held-out* views against the distributed model, i.e. the adversary's cost (training-view quality rises throughout). r after re-association and at the key's stored indices (*oracle*); passes at $\tau = 0.6$.

			r re-associated		r oracle indices			
Fine-tuned	steps	ΔPSNR	mean	min	mean	min	matched (%)	passes
SH only	100	+0.01	0.9615	0.9282	0.9615	0.9282	100.0	5/5
SH only	500	−0.04	0.8994	0.8458	0.8994	0.8458	100.0	5/5
SH only	2,000	−0.18	0.7740	0.6634	0.7740	0.6634	100.0	5/5
SH only	5,000	−0.39	0.5257	0.4312	0.5257	0.4312	100.0	1/2
SH only	10,000	−0.63	0.3946	0.3109	0.3946	0.3109	100.0	0/2
SH only	20,000	−0.95	0.3034	0.2367	0.3034	0.2367	100.0	0/2
SH + geometry	2,000	−1.00	0.4857	0.2971	0.8449	0.7428	12.6	1/5

80%, while costing the adversary at most 1.7 dB at 30% and 9.6–18.0 dB at 80%. Near-threshold outcomes are seed-sensitive.

A white-box adversary can also target parameter bands (Table 7, Appendix F). For wm1, truncation-aware extraction passes on all 30 scenes at degree 2 but fails at degree 1 and 0, making truncation to degree ≤ 1 an effective and sometimes cheap removal attack, costing 1.36 dB on average at degree 1. Over all payloads 114 of 150 models fail at degree 1, and none of the 36 passes establishes image recovery: 30 are wm2 models lifted by that payload's high null floor and 6 are wm3 models carrying two tiles, both consequences of the tiling index map. The coprime stride keeps bit payloads decodable – every bit decodes at every degree across all 42 runs (margin 0.062 at degree 0) – whereas a noise-free simulation of plain tiling gives 52–81%, near chance at degree 0.

Adaptive attacks. An adversary with the file and comparable training data can re-optimize the model under the ordinary 4DGS photometric loss, with no key and no watermark term. We run this on five scenes for 100–20,000 steps, over the SH alone or over SH and geometry, and extract both after re-association and at the key's stored indices, which separates an erased payload from an unmatched one. Both variants defeat verification (Table 3). Fine-tuning the SH erases the payload: every entry still matches, but mean r falls from 0.774 at 2,000 steps to 0.395 at 10,000. Fine-tuning geometry as well defeats verification at 2,000 steps on four of five scenes *without* erasing the mark ($r = 0.845$ at the stored indices): the marked centres move 0.32–0.66 s_{nn}, far beyond ϵ, and the fraction still inside ϵ predicts the match rate to within 0.0002 on every run. The attack costs 0.39 dB on held-out views at the first SH-only budget that breaks verification and 1.00 dB with geometry, a price many would pay, so we do not claim robustness against adaptive fine-tuning; the geometry variant shows that a defence needs position-robust matching, not a stronger payload.

Attacks on the matcher. Three cheaper attacks target re-association rather than the payload (Appendix P). Duplicating Gaussians in place makes key entries ambiguous, and duplicating all of them defeats the shipped matcher ($r = 0.0285$, $0/30$) at the cost of doubling the file; taking the lowest-numbered bit-identical candidate instead restores $r = 0.9997$ at every duplication rate, and requiring a lone candidate to be bit-identical removes all tolerance-induced false matches. A global translation or rotation matches nothing, but aligning the file to the original model first recovers translation on all 30 scenes and rotation on 25. We report the shipped matcher throughout so that results stay comparable.

4.5 Reference Comparison with Mark4D

Table 4 places SHield beside Lee et al. (2026) on the same scenes and message lengths; baselines are quoted and the underlying D-NeRF protocol differs (Appendix J). SHield decodes every bit in all 42 runs at α_0=0.1, rendering at 44.16–44.40 dB against the original model. SHield holds the best bit accuracy and PSNR in Table 4 and Mark4D the best SSIM and LPIPS, but the accuracy in particular follows from the access model rather than from a better watermark: closed-form subtraction of the stored coefficients on an unmodified file recovers the symbols exactly, so anything below 100% would indicate a bug. The comparison is a cross-threat-model reference, not a ranking, and it measures fidelity and capacity rather than robustness; under a strict instantiation of Mark4D's own model-level

Table 4: **Cross-threat-model reference with Table 1 of Mark4D (Lee et al., 2026).** Baselines are quoted; SHield is measured on our base models with the same message lengths but a different D-NeRF protocol. **Bold** marks the best value in each column; the row groups answer different threat models, so this is a cross-threat-model reference rather than a ranking. Acc.: bit accuracy (%); "–": unsupported. Baseline decoders: (Ma et al., 2022; Zhu et al., 2018; Tancik et al., 2020; Fernandez et al., 2022; Zhang et al., 2019; Fernandez et al., 2024).

	32 bits				48 bits				64 bits			
Method	Acc.	PSNR	SSIM	LPIPS	Acc.	PSNR	SSIM	LPIPS	Acc.	PSNR	SSIM	LPIPS
Frame-level, render domain: decode rendered frames												
CIN	67.66	35.62	.9814	.0261	64.22	34.96	.9794	.0290	59.64	34.49	.9780	.0278
HiDDeN	82.21	31.48	.9550	.0425	79.11	31.07	.9566	.0375	76.85	31.67	.9556	.0478
StegaStamp	85.12	33.03	.9793	.0472	80.59	32.64	.9718	.0554	76.09	32.32	.9707	.0540
SSL	85.24	30.73	.9651	.0436	81.52	29.98	.9620	.0486	78.37	29.56	.9596	.0504
GaussianMarker	86.36	32.94	.9627	.0405	82.44	32.17	.9597	.0459	80.92	33.59	.9688	.0440
GuardSplat	88.33	38.69	.9934	.0082	84.74	37.48	.9921	.0105	81.30	37.49	.9916	.0118
3D-GSW	90.46	31.98	.9651	.0321	87.49	32.09	.9604	.0411	83.36	30.87	.9504	.0512
Mark4D	96.34	42.32	**.9960**	**.0018**	94.54	41.45	**.9954**	**.0023**	92.71	41.27	.9954	**.0022**
Video-level, render domain: decode rendered clips												
RivaGAN	80.51	30.41	.9556	.0506	–	–	–	–	–	–	–	–
VideoSeal	88.21	36.29	.9811	.0103	82.72	35.55	.9780	.0122	77.98	36.16	.9810	.0211
Mark4D	96.96	42.32	**.9960**	**.0018**	95.02	41.45	**.9954**	**.0023**	92.77	41.27	.9954	**.0022**
Model file, non-blind: read the file with the owner's key												
SHield (ours)	**100.00**	**44.40**	.9958	.0036	**100.00**	**44.16**	**.9954**	.0039	**100.00**	**44.32**	**.9958**	.0036

distortion (SH noise, 20% of Gaussians removed at random), SHield still decodes all 32 bits on all eight D-NeRF scenes.

5 LIMITATIONS

SHield protects the model file; content that circulates only as rendered video calls for a render-domain watermark, and the two are complementary. Verification is non-blind: the owner keeps a key of 0.5–18.1 MB (median 2.8 MB) per model and registers it, with the payload, before distribution. Re-association expects essentially unchanged centres; rigid transforms can be undone by alignment, but noisy or half-precision centres cannot, so position-robust matching, which would also blunt the geometry fine-tuning attack, is the most useful next step.

The uncentred statistic has a payload-dependent floor: on an unmarked model the recovery collapses to the payload's per-channel mean colour, which correlates with the payload at $\sqrt{\eta^2}$ (0.029–0.368 for our payloads, undefined for greyscale wm3). Payloads therefore need η^2 well below $\tau^2 = 0.36$, checked once at registration, or the centred statistic. Our robustness claims cover non-adaptive post-processing (for SH truncation, bit payloads only); we report the adaptive and truncation attacks on image payloads without claiming robustness to them. Dedicated 3DGS compression pipelines (In et al., 2025; Wang et al., 2026) remain untested, temporal stability is measured with a rendering-cost proxy rather than a video benchmark, and the anchored optimization uses the training images, although anchoring to the owner's own renders works about as well.

6 CONCLUSION

SHield watermarks redistributed 4DGS assets without a decoder: a key-selected, tiled payload in the canonical SH coefficients is time- and view-invariant, and extraction is closed-form arithmetic on the model file. It resists INT8 quantization and opacity pruning, while the evaluation maps where it stops – unchanged centres are required, SH truncation removes image payloads, and an adversary who re-trains the model can remove the mark at a small quality cost. Those limits name the next steps: position-robust re-association, statistics without payload-dependent null floors, and pairing file-level marks with render-domain ones, which protect a different surface.

AI USE STATEMENT

In this work, we used large language model (LLM) assistants to help with writing and editing portions of the paper text. All AI-assisted text has been reviewed, verified, and revised by the authors. All experimental results, claims, and conclusions are the sole responsibility of the authors. We take full responsibility for the final content of this work.

ETHICS STATEMENT

This work proposes a watermarking method for 4D Gaussian Splatting models intended to help content creators establish ownership of their assets. It does not involve human subjects, personally identifiable information, or sensitive datasets. As with other watermarking methods, the same mechanism could be used to hide unrelated data in shared models, and a description of the carrier could inform removal attempts; we report the attacks that defeat the image payload (§ 4.4) so that users can judge its limits.

REPRODUCIBILITY STATEMENT

The experimental setup, hyperparameters and dataset configurations are described in § 4 and Appendix A. The Mark4D comparison setup, including its differences from the Mark4D protocol, is detailed in Appendix J, the re-association validation in Appendix B, and the verification protocol, including the in-sample choice of τ, the scene-disjoint split and its seed, and the fresh trials, in Appendix C. All experiments use the publicly available D-NeRF (Pumarola et al., 2021), HyperNeRF (Park et al., 2021) and DyNeRF (Li et al., 2022) datasets. Per-scene, per-payload and per-seed breakdowns are provided in the appendix.

REFERENCES

Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-Gaussian embedding-based deformation for deformable 3D Gaussian splatting. In *European Conference on Computer Vision (ECCV)*, pp. 321–335, 2024.

Zixuan Chen, Guangcong Wang, Jiahao Zhu, Jianhuang Lai, and Xiaohua Xie. Guardsplat: Efficient and robust watermarking for 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Ingemar Cox, Matthew Miller, Jeffrey Bloom, Jessica Fridrich, and Ton Kalker. Digital watermarking and steganography. *Morgan Kaufmann Publishers*, 2007.

Pierre Fernandez, Alexandre Sablayrolles, Teddy Furon, Hervé Jégou, and Matthijs Douze. Watermarking images in self-supervised latent spaces. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

Pierre Fernandez, Hady Elsahar, I. Zeki Yalniz, and Alexandre Mourachko. Video seal: Open and efficient video watermarking. *arXiv preprint arXiv:2412.09492*, 2024.

Yijia Guo, Wenkai Huang, Tong Hu, Gaolei Li, Yang Li, Yuxin Hong, Liwen Hu, Xitong Ling, Jianhua Li, Shengbo Chen, Tiejun Huang, and Lei Ma. Splats in splats++: Robust and generalizable 3D Gaussian splatting steganography. *arXiv preprint arXiv:2604.15862*, 2026a.

Yijia Guo, Wenkai Huang, Yang Li, Gaolei Li, Hang Zhang, Liwen Hu, Jianhua Li, Tiejun Huang, and Lei Ma. Splats in splats: Robust and effective 3D steganography towards Gaussian splatting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(6):4485–4493, 2026b.

Xiufeng Huang, Ruiqi Li, Yiu-ming Cheung, Ka Chun Cheung, Simon See, and Renjie Wan. Gaussianmarker: Uncertainty-aware copyright protection of 3D gaussian splatting. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Sumin In, Youngdong Jang, Utae Jeong, MinHyuk Jang, Hyeongcheol Park, Eunbyung Park, and Sangpil Kim. CompMarkGS: Robust watermarking for compressed 3D Gaussian splatting. *arXiv preprint arXiv:2503.12836*, 2025.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5938–5948, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.

A. Kerckhoffs. La cryptographie militaire. *Journal des Sciences Militaires*, pp. 161–191, 1883.

Jaejin Lee, Minjae Jeong, Joonhyuk Park, Yechan Hwang, Seunghun Baek, and Won Hwa Kim. Mark4D: Temporally-consistent watermarking for 4D Gaussian Splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39393–39402, 2026.

Chenxin Li, Brandon Y. Feng, Zhiwen Fan, Panwang Pan, and Zhangyang Wang. StegaNeRF: Embedding invisible information within neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, and Zhaoyang Lv. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5521–5531, 2022.

Hengyu Liu, Chenxin Li, Wentao Pan, Zhiqin Yang, Yifeng Yang, Yifan Liu, Wuyang Li, and Yixuan Yuan. Hide-in-motion: Embedding steganographic copyright information into 4D Gaussian splatting assets. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2694–2700, 2025. doi: 10.1109/ICRA55743.2025.11128285.

Ziyuan Luo, Qing Guo, Ka Chun Cheung, Simon See, and Renjie Wan. CopyRNeRF: Protecting the Copyright of Neural Radiance Fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Rui Ma, Mengxi Guo, Yi Hou, Fan Yang, Yuan Li, Huizhu Jia, and Xiaodong Xie. Towards blind watermarking: Combining invertible and non-invertible mechanisms. In *Proceedings of the ACM International Conference on Multimedia (MM)*, 2022.

Nicole Meng, Ronak Sahu, Miao Yin, Faysal Hossain Shezan, and Yingjie Lao. WhisperSplat: Lossless steganography in 3D Gaussian splatting. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *PMLR*, 2026.

Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M. Seitz. HyperNeRF: A higher-dimensional representation for topologically varying neural radiance fields. *ACM Transactions on Graphics*, 40(6):1–14, 2021.

Gabriel Peyré and Marco Cuturi. *Computational Optimal Transport*. Foundations and Trends in Machine Learning, 2019.

Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-NeRF: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10318–10327, 2021.

Yan Ren, Shilin Lu, and Adams Wai-Kin Kong. All that glitters is not gold: Key-secured 3D secrets within 3D gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2026.

Yingying Shi, Zhong Zhou, and Bin Zhou. Guard4D: Robust watermarking for 4D Gaussian splatting via decoupled decoding. In *Proceedings of the 35th International Joint Conference on Artificial Intelligence (IJCAI-ECAI)*, 2026.

Matthew Tancik, Ben Mildenhall, and Ren Ng. StegaStamp: Invisible hyperlinks in physical photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2117–2126, 2020.

Boyu Wang, Jun Xia, and Mingsong Chen. Robust3DGSW: Toward robust watermarking for quantization-aware 3D Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

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

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4D Gaussian Splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20310–20320, 2024.

Zeyu Yang, Hongye Yang, Zijie Pan, Xiatian Zhu, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4D Gaussian Splatting. In *International Conference on Learning Representations*, 2024.

Kevin Alex Zhang, Lei Xu, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Robust invisible video watermarking with attention. *arXiv preprint arXiv:1909.01285*, 2019.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Xuanyu Zhang, Jiarui Meng, Runyi Li, Zhipei Xu, Yongbing Zhang, and Jian Zhang. GS-Hider: Hiding messages into 3D Gaussian splatting. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Sifan Zhou, Hang Zhang, Yuhang Wang, and Ming Li. 4D-GSW: Kinematic-aware spatio-temporal consistent watermarking for 4D Gaussian Splatting. *arXiv preprint arXiv:2605.22342*, 2026.

Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. HiDDeN: Hiding data with deep networks. In *European Conference on Computer Vision*, pp. 682–697, 2018.

A Implementation Details

Algorithm 2 SHield embedding

Input: 4DGS model $\mathcal{G}$ with SH $\mathbf{h}$; payload $\mathbf{W}$; seed κ; scale α_0; training views $\mathcal{V}$
Output: watermarked model $\tilde{\mathcal{G}}$, key $\mathcal{K}$

$K \leftarrow$ adaptive count (Eq. 1)
$\mathcal{S} \leftarrow \textsc{Sample}(N, K; \kappa)$ ▷ sorted, no replacement
$n_{\text{slots}} \leftarrow K\,C_{\text{tot}}$; resize $\mathbf{W}$ if $n_{\text{wm}} > n_{\text{slots}}$; $n_t \leftarrow \lfloor n_{\text{slots}}/n_{\text{wm}} \rfloor$
for $c \in \{R, G, B\}$ **do**
  $\mathbf{z}_c \leftarrow \mathbf{W}_c - \overline{\mathbf{W}_c}$ ▷ store $\overline{\mathbf{W}_c}$ in $\mathcal{K}$
  for $s = 0 \ldots n_{\text{slots}} - 1$ **do**
    $t \leftarrow \lfloor s/n_{\text{wm}} \rfloor$; $j \leftarrow \big((s \bmod n_{\text{wm}}) + t\sigma\big) \bmod n_{\text{wm}}$ ▷ σ=0 for images
    $\mathbf{E}_c[\lfloor s/C_{\text{tot}} \rfloor,\ s \bmod C_{\text{tot}}] \leftarrow \mathbf{z}_c[j]$
  end for
end for
$\Delta \leftarrow \alpha_0 \mathbf{E}$ ▷ warm start
for $i = 1 \ldots T$ **do**
  $\mathbf{h}' \leftarrow \textsc{Detach}(\mathbf{h})$; $\mathbf{h}'[\mathcal{S}] \leftarrow \mathbf{h}[\mathcal{S}] + \Delta$
  render n_v views of $\mathcal{V}$ with $\mathbf{h}'$ ▷ grads to Δ
  $\Delta \leftarrow \textsc{Adam}\big(\Delta, \nabla_\Delta[\mathcal{L}_{\text{render}} + \lambda_{\text{wm}}\mathcal{L}_{\text{wm}}]\big)$ ▷ Eq. 2
end for
$\mathbf{h}[\mathcal{S}] \leftarrow \mathbf{h}[\mathcal{S}] + \Delta$ ▷ $\mathbf{a}$: opacity, scale, rotation
$\mathcal{K} \leftarrow (\mathcal{S}, \mathbf{h}_{\text{orig}}[\mathcal{S}], \boldsymbol{\mu}[\mathcal{S}], \mathbf{a}[\mathcal{S}], \alpha_0, \sigma, \overline{\mathbf{W}}, \mathbf{W})$

SH Tensor Layout. In the official 4DGS implementation (Wu et al., 2024), the SH appearance coefficients for each Gaussian are maintained as two `nn.Parameter` tensors: `_features_dc` $\in \mathbb{R}^{N\times 1\times 3}$ (representing the degree-0 DC component) and `_features_rest` $\in \mathbb{R}^{N\times(C_{\text{tot}}-1)\times 3}$ (representing degrees 1 to d_{max}). During rendering, these tensors are concatenated to form a single $[N, C_{\text{tot}}, 3]$ tensor. Because SHield embeds the payload independently into each colour channel, the total available slot count per channel is KC_{tot}.

Key Format and Storage. The verification key file (`watermark_info.npz`) is designed to be self-contained. It strictly stores:

(i) The selected Gaussian indices $\mathcal{S}$.

(ii) The original, unmodified SH coefficients of the selected Gaussians.

(iii) The canonical centres of the selected Gaussians, alongside the model's median nearest-neighbour spacing. To resolve tie-breaks during re-association, we also store their original opacity, scale, and rotation.

(iv) Payload metadata: shape, channel count, and per-channel means. For bit messages, this includes the message itself, the coprime stride σ, and the whitening sequence.

(v) Hyperparameters: the perturbation scale α_0, the effective selection ratio, and the random seed.

Because the stored reference coefficients dominate the file footprint, the key size scales linearly with K and remains completely independent of the training image resolution or video sequence length. Across all 192 models evaluated in this paper, the compressed key sizes range from 0.514 to 18.116 MB (median 2.849 MB). Broken down by payload type, the median size is 4.175 MB for the 150 image-payload keys, and 1.756 MB for the 42 bit-payload keys. The identity fields stored for tie-breaking (iii) add approximately 24.3% to the median key size.

SHield Settings. All experiments utilize the following fixed hyperparameter configuration unless otherwise specified:

Hyperparameter	**Value**
Iterations T	500
Adam learning rate	10^{-3}
Loss weights (λ_{wm}, λ_{ssim})	100, 0.2
Views per step n_v (pool per model)	4 (32)
Perturbation scale α_0	0.1
Selection bounds (ρ_{min}, ρ_{max}); Min tiles $n_t^{\min}$	0.10, 0.50; 2
Key seed (selection, views; bits: message, whitening)	42
Re-association tolerance ϵ	0.01 s_{nn}

Anchored Slots. In Eq. 4, tile$(\mathbf{z}_c, n_t)$ repeats $\mathbf{z}_c$ exactly n_t times, which is shorter than the KC_{tot} slots of $\mathbf{e}_c$ whenever n_{wm} does not divide KC_{tot}. For image payloads the loss therefore anchors the first $n_t n_{\text{wm}}$ slots, the ones fold-and-average reads (Eq. 5); the trailing partial tile is written but never read, and is left to $\mathcal{L}_{\text{render}}$ alone. For bit payloads the target is expanded over all KC_{tot} slots instead, because the coprime stride makes it more than a plain repeat of the symbol vector.

Runtime Overhead.

- **Tiled Embedding:** < 1 s (requires tensor arithmetic only).
- **Optimization (SHield):** 37–179 s (measured across the 12 logged seed-study re-embeddings detailed in Appendix I). This assumes 500 steps $\times$ 4 views on a shared NVIDIA RTX 4090, inclusive of loading the model, training cameras, and ground-truth images.
- **Extraction:** < 1 s plus re-association time. Extraction requires only a PLY file read, nearest-centre nearest-neighbor queries, and basic arithmetic, with zero rendering overhead.

Scenes and Evaluation Details. Our evaluation on the 17 HyperNeRF scenes spans three distinct capture splits: *interp* (aleks-teapot, chickchicken, cut-lemon1, hand1-dense-v2, slice-banana, torchocolate); *misc* (americano, cross-hands1, espresso, keyboard, oven-mitts, split-cookie, tamping); and *vrig* (broom2, vrig-3dprinter, vrig-chicken, vrig-peel-banana).

For robustness evaluations:

- **Quantization:** Whole-file INTb quantization is applied as per-tensor min–max uniform quantization. Specifically, the DC and higher-order SH tensors of the entire file are individually rounded to 2^b uniformly spaced levels between that tensor's minimum and maximum values. FP16 is applied as a standard half-precision cast. During extraction, the original, unquantized reference stored in the key is used.
- **Pruning Cost (§ 4.4):** The rendering cost of the adversary's pruning attack is measured on the wm1 models of *bouncingballs*, *hellwarrior*, *hook*, *jumpingjacks*, *mutant*, *standup*, *trex* (D-NeRF), and *coffee_martini*, *cook_spinach*, *flame_steak* (DyNeRF). We simulate physical deletion by forcing the opacity of the targeted Gaussians to zero, and then compare the PSNR against ground truth on 20 evenly spaced test views.
- **Rendering Residual (§ 4.2):** Residual evaluations utilize all 20 test views for D-NeRF, and 30 evenly spaced test views per scene for HyperNeRF and DyNeRF (yielding 830 total evaluated views).

B KEY RE-ASSOCIATION

Re-association Algorithm. To securely re-associate the verification key with a suspect file, we employ a geometric matching strategy. For a given key entry k with stored canonical centre $\boldsymbol{\mu}_k$, the algorithm proceeds as follows:

- **Candidate Retrieval:** Identify all suspect Gaussians j where $\|\tilde{\boldsymbol{\mu}}_j - \boldsymbol{\mu}_k\| \leq \epsilon$. We strictly define $\epsilon = 0.01\, s_{\text{nn}}$, where s_{nn} is the median nearest-neighbour spacing of the watermarked model (stored in the key).
- **Unique Match:** If exactly one candidate satisfies the distance threshold, the entry is assigned to it.
- **Tie-Breaking:** If multiple candidates fall within the ϵ tolerance, we consult static attributes (opacity, scale, and rotation). The entry is assigned *only* if exactly one candidate possesses attributes bit-identical to the key's stored values. Otherwise, it is left unmatched.
- **Conflict Resolution:** If multiple key entries claim the same suspect Gaussian, only a single claimant with bit-identical attributes retains it; all others are left unmatched. Note that static attributes are only used for tie-breaking and never override a unique geometric centre match.
- **Fallback:** An unmatched entry substitutes the stored reference coefficients in place of the suspect coefficients. This yields a residual of zero in the extraction math, mathematically equivalent to a removed Gaussian.

This tolerance ϵ was fixed prior to the experiments described below and was not iteratively tuned.

Baseline Validation (Topological Robustness). Table 5 applies four conditions across all 192 models. Here, "Exact" denotes an extraction outcome perfectly identical to the index-based reference (e.g., the unmodified file for Conditions A and B, and in-place removal of the exact same Gaussians for C and D).

Without re-association, a simple row permutation completely destroys extraction (dropping image r to between -0.018 and 0.075, and bit accuracy to near-chance levels of 34–66%). With our algorithm applied to unmodified files, 99.66% of key entries found exactly one candidate, while 0.34% found several (up to 40). Our attribute tie-breaker successfully resolved all $17{,}647$ ambiguous entries; notably, no model contained two Gaussians with perfectly identical centres and attributes. The rare false matches observed among removed entries were entirely tolerance-induced (i.e., a removed Gaussian happened to have a surviving neighbor within the ϵ radius).

Table 5: **Geometric re-association performance across all 192 models** (150 image, 42 bit payloads). Our centre-based matching algorithm demonstrates mathematically flawless structural recovery: across all adversarial conditions, surviving marked Gaussians were *never* assigned to an incorrect row (0 mis-assignments). **Metrics:** Δ denotes the maximum observed change in extraction fidelity (WM-PSNR and Pearson r) relative to theoretical, perfectly aligned in-place removal. **Baseline Alignment:** Under Condition C (30% opacity pruning), the geometrically recovered mean r=0.892 and WM-PSNR of 18.21 dB perfectly reproduce the in-place baseline established in Table 2, proving our re-association module incurs virtually zero mathematical penalty.

Condition	exact	mis-assigned	removed entries falsely matched	Δ WM-PSNR / r	bits
A Unmodified file	192/192	0	–	0 / 0	100%
B Rows permuted	192/192	0	–	0 / 0	100%
C 30% lowest opacity deleted, permuted	117/192	0	123 (0.008%)	0.006 dB / 1.5×10^{-4}	100%
D 20% random deleted, permuted	28/192	0	1,189 (0.116%)	0.017 dB / 2.1×10^{-4}	100%

Stress-Testing Geometric Limits. To locate the exact breakdown point of the matcher, we applied isotropic Gaussian noise (per-axis s.d. σ) to every centre across all 192 models (with rows permuted, but SH and other attributes unchanged). Thresholds for material degradation were declared prior to scoring (Figure 5).

All marked Gaussians are successfully matched up to $\sigma \leq 0.001\, s_{\text{nn}}$. At $0.003\, s_{\text{nn}}$ (roughly 0.8% of the median Gaussian radius), 98.88% of Gaussians remain matched. At $\sigma = \epsilon$, matching drops to 19.90%, causing image-payload verification to fail on 103 of the 150 models. (These measured rates follow theoretical χ_3 predictions to within 0.03 percentage points). Importantly, incorrect matches remain safely below 0.1% of key entries at every noise level (peaking at 0.093%). Bit payloads prove more resilient: they only begin losing bits at $\sigma = 0.03\, s_{\text{nn}}$ (on a single model) and fall below 90% mean accuracy at $0.1\, s_{\text{nn}}$.

Finally, casting the model centres to float16 leaves only 3.75% of entries matched (none incorrectly). While this defeats image verification on 135 of 150 models, bit payloads still decode successfully. As a diagnostic test, relaxing the tolerance to $0.1\, s_{\text{nn}}$ matches 71.0% of Gaussians after a float16 cast, but unacceptably increases false matches of removed entries to 1.79% under 20% deletion; no single tested tolerance securely handles both. When matching fails entirely, extraction defaults to the flat mean-colour block of an unmarked model. While this falls below τ for our tested payloads, it may not for payloads with high η^2 (Appendix C). Therefore, we explicitly state that this re-association regime does not provide security against targeted, adaptive geometric manipulation.

C Ownership Verification: Held-out Evaluation and Null Floor

Formalizing the Evaluation. To rigorously evaluate the watermark, we define two hypothesis states:

- H_1 **(True Positive):** The correct key is applied to a watermarked model (either exactly as distributed or after post-processing).
- H_0 **(Null Hypotheses):** Verification is attempted without the correct mark. We model this under three adversarial or coincidental conditions: *other payload* (owner i's key used on the same scene carrying owner j's mark), *wrong subset* (using the stored coefficients and payload with a randomly drawn selection of Gaussians, simulating a partially compromised key), and *unmarked* (applying the owner's key to a clean, unmodified model).

Strict Evaluation Protocol. An initial in-sample analysis computed $1{,}800$ extractions across all 150 models ($1{,}770$ yielded a defined r; the 30 unmarked-model scores for the greyscale wm3 are mathematically undefined). From this, we selected a decision threshold of $\tau = 0.6$.

To prevent data leakage, we subsequently split the 30 scenes into 14 calibration scenes (70 models) and 16 evaluation scenes (80 models), stratified by dataset using a fixed seed (20260916). We locked $\tau = 0.6$ unchanged. (The calibration scores naturally leave a wide margin around this threshold: max $H_0 = 0.4972$, min $H_1 = 0.8169$). The split, the decision rule ($r \geq \tau$, treating undefined r as a non-detection), and the fresh-trial design were all fixed before any fresh trial was scored. We make

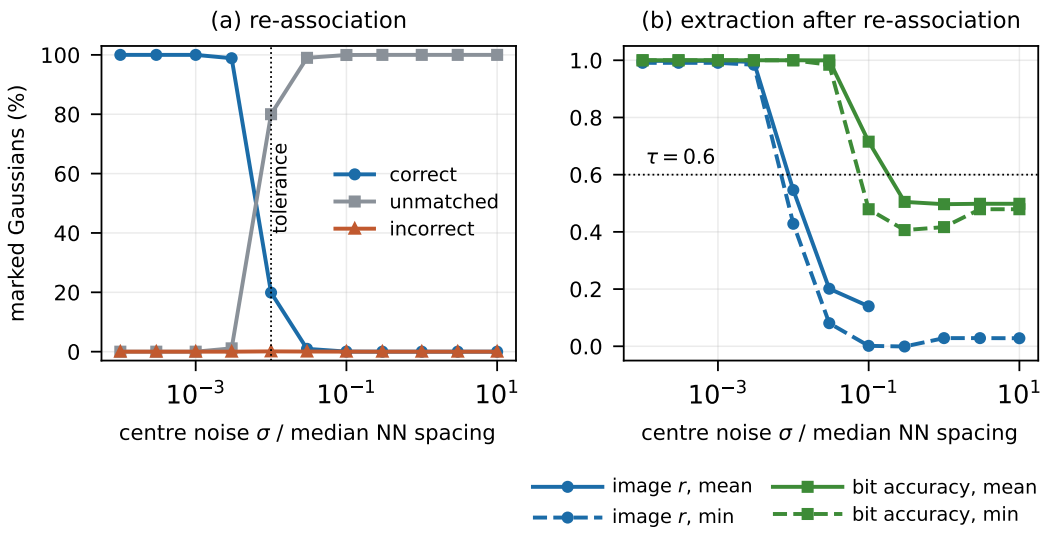

Figure 5: **Stress-testing geometric re-association under spatial noise.** We evaluate the breakdown point of the matcher across all 192 models by injecting isotropic Gaussian noise into the canonical centres. The fixed matching tolerance (ϵ=0.01 s_{nn}) is indicated by the dotted line. **(a) Matching Dynamics:** The proportion of marked Gaussians that are correctly matched, safely left unmatched, or falsely matched to an incorrect Gaussian. Crucially, the false-match rate remains strictly near zero across all noise levels. **(b) Extraction Fidelity:** The resulting image-payload Pearson r (mean and minimum over 150 models) and bit accuracy (mean and minimum over 42 runs) following the noisy re-association. **Mathematical Bound:** The mean image r becomes mathematically undefined beyond σ=0.1 s_{nn}. At this extreme noise level, every single entry is safely left unmatched, causing the greyscale payload's recovery to collapse into a flat mean-colour block with exactly zero variance.

no such claim for τ itself: it was chosen in-sample, as stated above, and only then carried unchanged onto scene-disjoint data.

The final evaluation consists of *retrospective* scores (the original conditions on the evaluation scenes) and *fresh* trials drawn exclusively for this analysis. The fresh trials include 400 true H_1 trials (5 per model, under 30% random Gaussian deletion, row permutation, and key re-association) and 1,600 null H_0 trials (20 per model, drawing wrong subsets from new random seeds).

Statistical Bounds and Assumptions. Our results demonstrate zero false positives across the 1,600 fresh wrong-subset null trials from the 80 evaluation models. While the exact per-trial Clopper–Pearson upper bound is nominally 0.19%, we emphasize that this assumes fully independent trials. Because the 20 null trials for a given model share the same payload and selection, they are correlated; thus, this is not a strict bound on the per-trial false-positive rate.

Treating models or scenes as independent units, the nominal one-sided 95% Clopper–Pearson upper bounds on producing any false positive are 3.68% per model and 17.07% per scene. Again, in the interest of complete statistical honesty, this independence assumption is imperfect because models share scenes and payloads (e.g., the five models of a scene share its base model). These figures represent our rigorous empirical test bounds rather than generalized deployment-level false-positive rates, especially given that the null distribution is heavily payload-dependent.

Sensitivity to the Threshold. Because τ was chosen in-sample, the held-out result could in principle depend on that choice. It does not, in a concrete sense. Across the whole evaluation set (640 H_1 and 2320 H_0 trials, 16 of the latter undefined), the largest null score is 0.4981 and the smallest true score 0.8147. Every decision reported above is therefore unchanged for any threshold in $(0.498, 0.815]$, a band 0.317 wide; $\tau = 0.6$ sits 32% into it, with 0.102 of headroom below and 0.215 above. Repeating the evaluation with a threshold taken from the calibration scenes alone, the midpoint 0.6570 of their own H_0/H_1 gap, which uses no evaluation data, accepts the same 640 true

Table 6: **Ownership verification on the held-out evaluation set** (16 scenes, 80 models, threshold τ=0.6). SHield achieves a perfect 100% detection rate across all true positive (H_1) conditions while successfully registering zero false positives across all null (H_0) conditions, including 1,600 fresh adversarial trials. **Decision Rule:** An undefined Pearson r strictly defaults to a non-detection. **Statistical Bounds & Transparency:** Because multiple trials within a single model inherently share payloads and selections, they are statistically correlated. Consequently, naive per-trial bounds (e.g., 0.19% for the 1,600 fresh nulls) mathematically underestimate the true false-positive rate. By strictly treating models or scenes as independent units, we derive nominal one-sided 95% Clopper–Pearson upper bounds of 3.68% per model (n=80) and 17.07% per scene (n=16). We explicitly report these boundaries, acknowledging their imperfect independence assumptions, to ensure absolute statistical rigor (§ C).

	Condition	n	min	median	max	decisions at τ
H_1, retrospective	correct key, distributed model	80	0.9910	0.9991	1.0000	80/80 detected
	after INT8	80	0.9766	0.9963	0.9995	80/80 detected
	after 30% pruning	80	0.8147	0.9103	0.9737	80/80 detected
H_0, retrospective	other payload, same scene	320	−0.0951	0.0060	0.3170	0 false positives
	wrong subset	320	0.0382	0.1417	0.4858	0 false positives
	unmarked model (64 defined)	80	0.0285	0.1400	0.3681	0 false positives
H_1, fresh	30% random deletion, permuted, re-associated	400	0.8192	0.9056	0.9677	400/400 detected
H_0, fresh	wrong subset, new seeds	1,600	0.0410	0.1413	0.4981	0 false positives

trials and yields the same zero false positives. The in-sample choice of τ therefore does not carry the result: any threshold in a band roughly a third of the score range wide gives the identical outcome. This does not remove the payload-dependence of the null floor below, which is what bounds how wide that band is for a given payload.

The Payload-Dependent Null Floor. During extraction, SHield adds back the stored per-channel mean. If the file carries no watermark information (e.g., an unmarked model, or all entries remain unmatched due to geometric noise), the recovery mathematically collapses into a flat block of that mean colour. The Pearson correlation of this flat block against the original payload is exactly $\sqrt{\eta^2}$, where η^2 represents the fraction of the payload's pixel variance that lies between its channel means.

For our tested payloads, $\sqrt{\eta^2}$ is 0.029 (wm1), 0.368 (wm2), 0.214 (wm4), and 0.067 (wm5), which exactly matches the measured unmarked-model scores (wm3 is greyscale, rendering the floor undefined). This variance floor shapes the other nulls as well: all 318 fresh wrong-subset trials that scored $r \geq 0.4$ utilized the wm2 payload. Crucially, a payload with $\eta^2 \geq \tau^2 = 0.36$ would be automatically falsely accepted from an unmarked model. This high floor also artificially inflates the scores of partially destroyed marks (e.g., wm2 crossing the threshold at SH degree 1; Table 7), which does not indicate successful visual recovery. (The 6 wm3 passes at SH degree 1 are not a channel-mean effect: they are the six scenes whose wm3 embedding carries two tiles, where the tiling index map attenuates pixels rather than erasing them (Appendix E). They likewise do not establish visual recovery.)

Therefore, our false-positive guarantees apply solely to the tested payloads and our specific protocol. For practical deployments, payloads must be explicitly screened to ensure η^2 is well below τ^2. We deliberately do not alter the statistic in this paper to mask this property. Figure 4 illustrates extractions from the models of Figure 3, visually confirming the flat recovery from an unmarked model.

D Carrier Selection

Theoretical Advantage vs. Empirical Cost. As established in Section 3, embedding the watermark into canonical SH coefficients provides a powerful theoretical advantage: because these coefficients are queried at every timestamp, a single write covers the entire dynamic sequence. However, identifying where a watermark *can* be mathematically placed does not dictate where it is visually *cheapest* to place.

Cost Isolation Experiment. Figure 6 addresses this by systematically isolating the empirical rendering cost of each potential carrier. In this ablation, the payload signal and the selected subset of Gaussians are held strictly constant. We then individually apply the perturbation across different parameter groups (position, scale, rotation, opacity, and SH bands). To ensure a fair comparison that does not artificially penalize parameters with narrow natural ranges, each perturbation is normalized, scaled to the same relative multiple (ε) of that specific parameter group's own standard deviation.

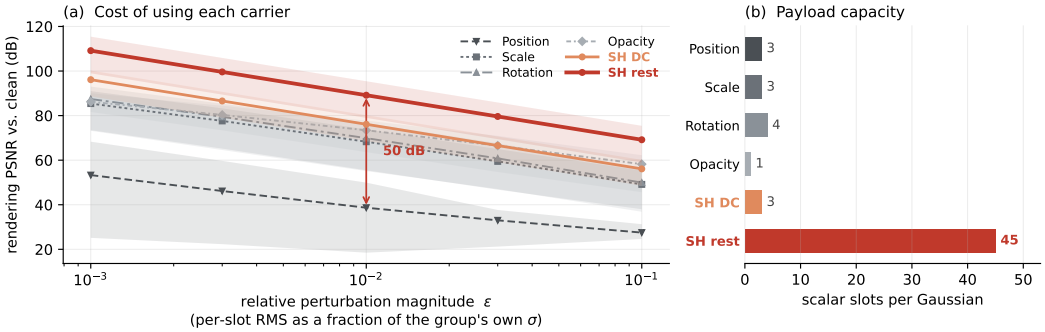

Figure 6: **Cost and capacity of Gaussian parameter groups as watermark carriers.** This ablation systematically isolates the rendering penalty of embedding into different parameter spaces, decisively justifying our selection of higher-order SH bands. **Ablation Setup:** The identical tiled, zero-mean payload is injected into each respective parameter group for the exact same key-selected subset (K=10%). To ensure a mathematically fair comparison across disparate parameter ranges, the perturbation is normalized such that its per-slot RMS strictly equals ε times that specific group's natural standard deviation. **(a) Empirical Rendering Cost:** PSNR measured against the clean render (where higher indicates a less visually destructive carrier), averaged over five scenes (four D-NeRF, one DyNeRF) with min–max variance shaded. At ε=0.01, perturbing spatial positions triggers a catastrophic rendering penalty (38.7 dB). In stark contrast, the higher-order SH bands achieve a near-lossless 89.2 dB, massively outperforming the DC band (76.1 dB) and geometric attributes like scale, rotation, and opacity (68–73 dB). **(b) Embedding Capacity:** The total available scalar slots per Gaussian, highlighting the immense capacity advantage (45 slots) of the higher-order SH bands.

Results and Temporal Stability. The empirical ordering of rendering costs remains perfectly consistent across every scene and every ε value in our sweep. While the absolute PSNR values for most parameter groups remain well above the threshold of human perceptibility, the relative gaps between them are substantial and dictate our design choice.

Crucially, perturbing canonical positions incurs a massive PSNR penalty compared to perturbing higher-order SH bands. Beyond static image quality degradation, geometry introduces a severe temporal vulnerability in 4DGS: because the deformation network continuously propagates canonical positions to every timestamp, a static spatial perturbation would inadvertently cause the rendered content to warp or shift over time. Consequently, higher-order SH bands provide the optimal combination of high embedding capacity, minimal static rendering penalty, and reliable temporal stability.

E Coprime Stride Interleaving

The Structural Threat of SH Truncation. Unlike operations that introduce statistical noise (e.g., quantization), SH-degree truncation structurally removes entire coefficient indices simultaneously. Figure 7 illustrates how this deterministic erasure affects the embedded payload, and demonstrates how a single integer stored in the key, the coprime stride, mathematically guarantees that bit messages remain decodable.

The Vulnerability of Plain Tiling. For bit messages, standard payload lengths ($L \in \{32, 48, 64\}$) are evenly divisible by the total number of SH coefficients ($C_{\text{tot}} = 16$). Under plain tiling, this divisibility causes every single copy of a specific bit to map to the exact same SH coefficient index. If

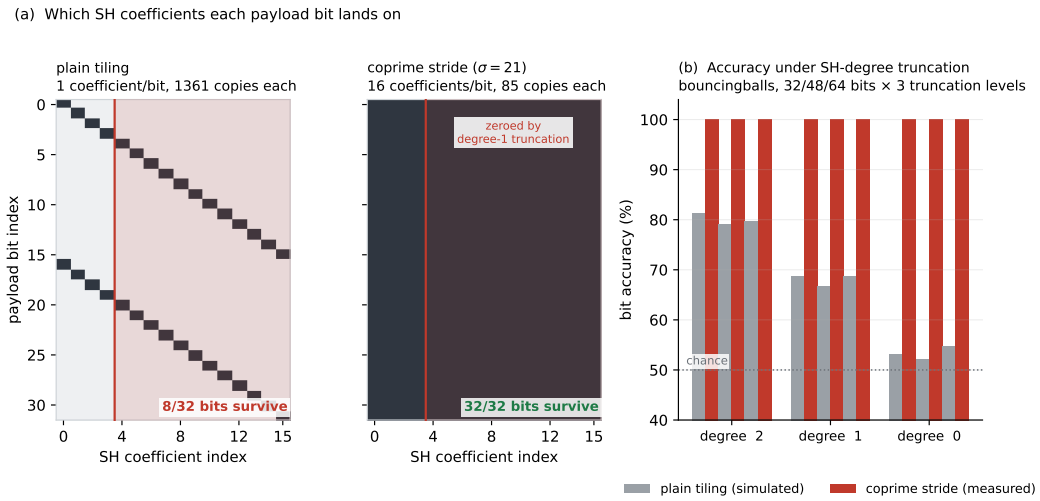

Figure 7: **Defeating targeted SH band erasure via coprime stride interleaving.** This figure mathematically demonstrates how our coprime stride prevents the catastrophic erasure of bit payloads when adversaries systematically truncate higher-order SH bands. In our flattened 1D formulation, slot s inherently maps to SH coefficient $s \bmod C_{\text{tot}}$ (where C_{tot}=16). **(a) Vulnerability vs. Defense:** We visualize a 32-bit message embedded into the *bouncingballs* scene (dark slots denote at least one embedded copy). *Naive Tiling:* Because C_{tot}=16 perfectly divides all baseline message lengths ($L \in \{32, 48, 64\}$), plain tiling collapses all ≈1,360 redundant copies of a specific bit into a *single* SH coefficient. Consequently, truncating the model to SH degree 1 (red) deterministically erases every single copy of 24 of the 32 bits. *Coprime Stride:* By shifting each tile t by $t\sigma$ (where σ is strictly coprime to both L and C_{tot}), SHield uniformly distributes the redundant copies across all 16 coefficients. Under the identical degree 1 truncation attack, the stride merely sacrifices 75% of the copies globally, safely preserving ≈340 redundant copies per bit to cleanly resolve the final sign decision. **(b) Empirical Resilience:** Validating this mathematical guarantee on *bouncingballs*: SHield maintains a flawless 100% measured bit accuracy across all truncation levels and message lengths. In stark contrast, a simulated noise-free naive tiling degrades proportionally with band erasure, plunging to near-chance recovery (52–55%) when truncated down to the DC band (degree 0).

an adversary truncates the model's SH degree and drops that specific index, the bit is permanently and entirely erased.

In our noise-free simulation of plain tiling, accuracy under truncation plummeted to near-chance levels (52–81%) across all 42 runs. Because this erasure is driven purely by structural arithmetic rather than signal degradation, increasing the embedding amplitude α_0 provides absolutely no defense against it.

The Coprime Stride Defense. To counter this vulnerability, SHield shifts each tile by an integer σ that is coprime to both L and C_{tot}: 21, 31 and 41 for $L = 32$, 48 and 64. Such a σ exists for every L, since σ=1 satisfies both conditions, and the search starts near L/φ so that consecutive tiles land far apart. This interleaving completely breaks the index alignment, spreading the copies of every individual bit evenly across all 16 SH coefficients.

When an adversary truncates the model to degree d, every bit safely retains a fraction $(d{+}1)^2/C_{\text{tot}}$ of its original copies. Consequently, the decoding accuracy remains at 100% across all degrees, while the decoding margin shrinks predictably in exact proportion to the surviving copies (0.561, 0.249, and 0.062 at degrees 2, 1, and 0, respectively; Table 7).

Impact on Image Payloads. Image payloads keep plain tiling ($\sigma = 0$), so the copies of payload index j occupy the SH coefficient indices $(j + t\, n_{\text{wm}}) \bmod C_{\text{tot}}$ for $t = 0, \ldots, n_t{-}1$, and each pixel therefore reaches $\min\big(n_t,\ C_{\text{tot}}/\gcd(n_{\text{wm}}, C_{\text{tot}})\big)$ distinct indices. Which failure mode applies is thus fixed by n_{wm} and n_t, not by signal strength. For wm1, $n_{\text{wm}} = 128 \times 394 = 50{,}432$ is an exact

Table 7: **SH-degree truncation: attack cost and survival, by payload type** (α_0=0.1). *Image payload wm1* (30 scenes): ΔPSNR against the untruncated render (mean [worst scene]); mean r and threshold passes with truncation-aware extraction (same bands zeroed in the reference) and naive extraction. *All five payloads*: passes of 150 models, aware (naive); †the 36 passes at degree 1 are 30 wm2 and 6 wm3, both accounted for by the tiling index map and, for wm2, its high null floor (Appendix E); none establishes image recovery. *Bit payloads*: 42 runs, accuracy, decoding margin (1 = unattacked) and PSNR of the 14 truncated L=32 models against the untruncated clean render. Per dataset: Appendix F.

	Image payload wm1 (30 scenes)				All image payloads	Bit payloads (42 runs)		
SH degree	ΔPSNR [worst]	r aware	passes aware	passes naive	passes aware (naive)	bit accuracy	margin	PSNR vs. clean (L=32)
3	0.00	0.9997	30/30	30/30	150 (150)	100%	0.998	44.40
2	−0.76 [−3.41]	0.749	30/30	19/30	150 (51)	100%	0.561	38.56
1	−1.36 [−4.54]	0.500	0/30	0/30	36† (0)	100%	0.249	36.69
0	−2.27 [−9.66]	0.251	0/30	0/30	0 (0)	100%	0.062	33.75

multiple of $C_{\text{tot}} = 16$, so every copy of a pixel lands on the *same* index, exactly as for bit payloads under plain tiling. Truncation to degree d then does not attenuate all pixels alike: it erases a fraction $1 - (d{+}1)^2/C_{\text{tot}}$ of the pixels outright, whose recovery falls back to the stored channel mean, and returns the remaining ones exactly. Writing $p = (d{+}1)^2/C_{\text{tot}}$ for the surviving fraction, the Pearson correlation of such a recovery against the payload is $\sqrt{p}$, giving 0.750, 0.500 and 0.250 at degrees 2, 1 and 0 and matching the measured 0.749, 0.500 and 0.251 (Table 7). Uniform attenuation would instead leave r unchanged, Pearson correlation being scale-invariant.

For the other four payloads n_{wm} is not a multiple of C_{tot} ($\gcd = 2$), so once $n_t \geq 2$ each pixel's copies reach several indices and truncation attenuates every pixel instead of erasing some, which holds r well above $\sqrt{p}$; at $n_t = 1$ there is one copy per pixel and the erasure behaviour returns. Replaying this index arithmetic over the stored payloads reproduces the measured truncation-aware r of every wm1 and wm2 model to within 0.005, and accounts for all 36 threshold passes at degree 1 (Table 7): the 30 wm2 passes, which carry $n_t \geq 2$ on every scene and are lifted further by that payload's 0.368 channel-mean floor, and the 6 wm3 passes, which are exactly the six scenes whose wm3 embedding carries two tiles ($r \approx 0.705$, against $r \approx 0.500$ on the 24 single-tile scenes). Neither group indicates visual recovery; both are structural artefacts of the index map and the channel-mean floor (§ 5).

F SH Truncation: Details

Simulation Methodology and DyNeRF Nuance. To evaluate the visual impact of SH truncation, we simulate the attack by limiting the 4DGS rasterizer to SH degree d during rendering. For D-NeRF and HyperNeRF, this rasterizer limit is mathematically identical to zeroing the stored SH bands above degree d directly in the file (yielding a maximum pixel difference of 0). However, on DyNeRF, simply zeroing the stored bands is insufficient (yielding pixel differences up to 0.37) because the deformation network dynamically applies time-dependent offsets to all 16 coefficients.

Crucially, because SHield's extraction is strictly a file-level operation, it operates directly on the stored parameter bands, bypassing these render-time offsets entirely. As expected, evaluating extraction at degree 3 reproduces the baseline untruncated evaluations exactly. Table 8 provides a comprehensive dataset-level breakdown of these truncation effects.

Adversarial Rendering Cost. From an adversarial perspective, degree-1 truncation is a highly accessible removal attack for image payloads. The rendering penalty incurred by the attacker is remarkably low, costing less than 0.25, 0.5, and 1.0 dB of PSNR on 3, 10, and 17 of the 30 evaluated scenes, respectively.

Verifying Bit Payload Survival (Sanity Check). Because maintaining 100% bit accuracy at degree 0 might initially appear counterintuitive, and could theoretically arise from a decoding math artifact, we conducted a rigorous sanity check. We attempted to decode two runs from entirely *clean* (unmarked) models using the exact same registered keys.

Table 8: **Dataset-level breakdown of robustness against SH-degree truncation** (α_0=0.1). This detailed ablation isolates the extraction fidelity and rendering penalties across D-NeRF, HyperNeRF, and DyNeRF under progressive SH band erasure. **Image Payloads (wm1):** We report the rendering penalty (ΔPSNR) against ground truth relative to the untruncated render, alongside the Pearson r and threshold passes for both truncation-aware and naive extraction. **Artifact Isolation:** Crucially, when an unmarked clean model is truncated in the exact same manner, its rendering score falls within a negligible 0.07 dB of the watermarked model. This mathematically proves that the observed visual degradation is fundamentally caused by the loss of higher-order SH detail, rather than the adversarial exposure of watermarking artifacts. **Bit Payloads:** We report the PSNR of the truncated L=32 models evaluated against the clean untruncated render (using Mark4D test views). Empowered by the coprime stride, bit decoding accuracy remains a flawless 100% on every single run across all datasets and all truncation degrees.

Dataset	degree	ΔPSNR (dB)	r aware / naive	passes aware / naive	bit models: PSNR (dB)
D-NeRF	2	−0.523	0.749 / 0.628	7/7 / 5/7	43.24
	1	−1.142	0.500 / 0.302	0/7 / 0/7	40.89
	0	−2.410	0.251 / 0.076	0/7 / 0/7	37.40
HyperNeRF	2	−0.243	0.749 / 0.619	17/17 / 10/17	–
	1	−0.785	0.500 / 0.300	0/17 / 0/17	–
	0	−1.420	0.251 / 0.079	0/17 / 0/17	–
DyNeRF	2	−2.476	0.749 / 0.590	6/6 / 4/6	32.33
	1	−3.240	0.500 / 0.279	0/6 / 0/6	31.09
	0	−4.535	0.251 / 0.071	0/6 / 0/6	28.88

On these clean models, the decoding accuracy correctly collapsed to near-chance levels (53.1% and 45.3%) with negligible decoding margins (0.018 and 0.017). By contrast, the watermarked models maintained 100% accuracy with margins of 0.064 and 0.062. This verifies that the degree-0 survival is a genuine structural recovery rather than a decoder artifact, though the surviving margin drops to roughly $3.6\times$ that of an unmarked model. We note that this sanity check is designed strictly to validate survival mechanics and does not serve as a formal false-positive estimate for bit payloads.

G Results per Scene and Payload

Consistent Extraction Fidelity. We provide a comprehensive breakdown of extraction performance by scene (Table 9) and by payload (Table 10), alongside visual examples of recovered payloads on a single scene (Figure 8). Regardless of dataset complexity or payload density, SHield's extraction remains exceptionally robust: all five evaluated payloads achieve a Pearson correlation of $r \geq 0.995$ across all three datasets.

The Real-World Optimization Trade-off. While r remains highly stable, the Watermark PSNR (WM-PSNR) exhibits greater variance and is systematically lower on the two real-world datasets (HyperNeRF and DyNeRF) compared to the synthetic D-NeRF scenes.

Because our extraction math subtracts the stored reference coefficients, the underlying scene content cancels out exactly. Therefore, this WM-PSNR gap is not an artifact of the complex SH values inherent to real-world scenes. Instead, it reflects the deliberate behavior of our anchored optimization. To maintain high rendering fidelity on complex, real-world topologies, the optimizer is forced to depart further from the initial tiled perturbation target. Our fold-and-average extraction suppresses this intentional residual, but does so only partially when the departure is large.

Tile Averaging Dynamics. This suppression mechanic is empirically validated by examining the tile count. Across the 17 HyperNeRF scenes, we observe a strong positive correlation between the recovered WM-PSNR and the mean tile count n_t ($r = 0.78$, $p = 0.0002$). This statistically confirms our algorithmic design: increasing the number of embedded payload copies (tiles) directly enhances the fold-and-average mechanism's ability to smooth out the optimization residual and cleanly recover the signal.

Table 9: **Comprehensive per-scene evaluation of SHield (α_0=0.1).** This granular breakdown demonstrates SHield's exceptional consistency, achieving near-perfect payload recovery (Pearson $r \geq 0.997$) across all individual scenes and datasets regardless of topological complexity. **Aggregation Setup:** Each row reports the mean performance across all five distinct payloads; HyperNeRF rows further aggregate the scenes within each respective capture split. **Metric Isolation:** Absolute metrics (PSNR, SSIM, and LPIPS) are measured against ground-truth images, primarily reflecting the natural reconstruction difficulty of each specific scene. Crucially, ΔPSNR isolates the pure mathematical rendering penalty incurred by the watermark, cleanly separating our defense's visual footprint from pre-existing baseline artifacts.

Scene / Split	$\text{PSNR}_{\text{clean}}$	PSNR_{wm}	ΔPSNR	SSIM_{wm}	LPIPS_{wm}	WM-PSNR	Pearson r
D-NeRF (7 synthetic monocular scenes)							
Bouncingballs	40.378	40.265	−0.113	0.9940	0.0112	41.39	0.9994
Hellwarrior	28.731	28.709	−0.022	0.9720	0.0296	42.54	0.9995
Hook	32.774	32.742	−0.032	0.9760	0.0201	45.29	0.9998
Jumpingjacks	35.497	35.462	−0.035	0.9855	0.0159	45.20	0.9998
Mutant	37.523	37.458	−0.065	0.9875	0.0106	46.28	0.9998
Standup	37.913	37.859	−0.054	0.9894	0.0095	46.16	0.9998
Trex	34.361	34.345	−0.016	0.9852	0.0176	46.54	0.9998
mean	35.311	35.263	−0.048	0.9842	0.0164	44.77	0.9997
HyperNeRF (17 real monocular scenes, by capture split)							
interp (6)	26.909	26.857	−0.052	0.7601	0.2933	36.45	0.9973
misc (7)	28.025	27.985	−0.040	0.8478	0.2392	36.08	0.9973
vrig (4)	25.000	24.950	−0.049	0.6745	0.3947	36.34	0.9974
mean	26.919	26.873	−0.047	0.7761	0.2949	36.27	0.9973
DyNeRF (6 real multi-view scenes)							
Coffee martini	28.012	27.890	−0.123	0.9031	0.1372	37.23	0.9980
Cook spinach	32.525	32.280	−0.246	0.9470	0.1017	36.61	0.9977
Cut roasted beef	32.119	32.090	−0.029	0.9467	0.1047	36.57	0.9976
Flame salmon	28.921	28.894	−0.027	0.9108	0.1295	37.51	0.9981
Flame steak	32.267	32.243	−0.024	0.9503	0.0940	35.78	0.9974
Sear steak	33.308	32.995	−0.314	0.9537	0.0922	35.85	0.9974
mean	31.192	31.065	−0.127	0.9353	0.1099	36.59	0.9977

H TILING VERSUS OPTIMIZATION, AND THE PERTURBATION SCALE

The Value of Anchored Optimization. This appendix compares each optimized model directly against its plain-tiling initialization, holding the perturbation scale ($\alpha_0 = 0.1$), the payload, and the selected subset strictly constant.

Across D-NeRF and HyperNeRF, the anchored optimization successfully reduces the rendering cost on all 120 model pairs, shrinking the mean $|\Delta\text{PSNR}|$ footprint from $0.079 \rightarrow 0.048$ dB and $0.263 \rightarrow 0.047$ dB, respectively. On the highly dynamic DyNeRF scenes (Table 11), the optimization reduces the mean rendering cost from $0.3914 \rightarrow 0.1302$ dB (a $3.01\times$ reduction, median per-pair ratio $2.73\times$). Notably, 29 of the 30 signed comparisons improved, with two optimized models marginally exceeding the rendering quality of the clean model.

This rendering gain naturally incurs an extraction trade-off. The optimizer's necessary departure from the tiled target lowers the mean WM-PSNR (e.g., dropping from 56.17 dB to 36.59 dB on DyNeRF). However, because the Pearson correlation remains exceptionally robust ($r \geq 0.9952$), the optimization presents a highly favorable trade-off for visual imperceptibility. While plain tiling is not strictly dominated across every metric, the optimized variant is strictly preferable for preserving asset fidelity.

Justifying the Perturbation Scale (α_0). Table 12 characterizes the watermark across all 150 image-payload models at varying values of α_0. We observe a clear scaling law: rendering cost remains negligible up to $\alpha_0 = 0.05$ but subsequently grows superlinearly, while extraction robustness yields diminishing returns.

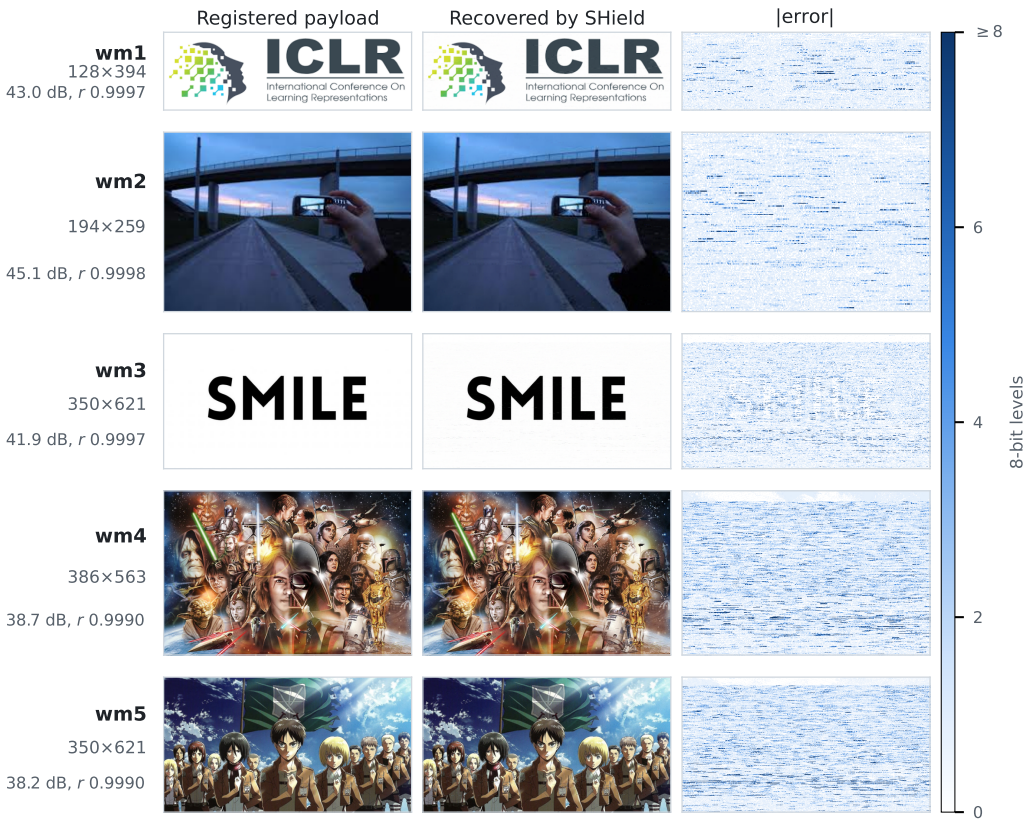

Figure 8: **Visual payload recovery under worst-case rendering conditions.** This figure qualitatively demonstrates that SHield successfully reconstructs highly recognizable images across a massive range of payload capacities, even on the dataset's most challenging scene. **Worst-Case Evaluation:** We deliberately visualize extractions from BOUNCINGBALLS, which represents the mathematical worst-case D-NeRF scene for Watermark PSNR (41.4 dB, compared to the 44.8 dB dataset average). Each payload is recovered directly from the distributed `.ply` utilizing the registered key. **Error Metric:** The *Error* column visualizes the absolute per-pixel RMS difference across RGB channels between the ground-truth payload and the reconstructed extraction, normalized to a single shared intensity scale for fair comparison. **Capacity vs. Redundancy:** The spatial resolution of the payload strictly dictates its spread-spectrum redundancy. The two compact 50k-element payloads securely embed two redundant copies per element, whereas the three massive 217k-element payloads are constrained to a single copy, directly accounting for the observed spread in WM-PSNR. Furthermore, the high-contrast, flat text payload naturally recovers 3–4 dB higher than the dense, high-frequency images of identical size.

We establish $\alpha_0 = 0.1$ as our universal operating point based on the following boundary constraints:

- **The Lower Bound (Vulnerability):** We explicitly avoid smaller scales like $\alpha_0 = 0.01$ because they leave the watermark vulnerable to quantization. Under whole-file SH INT8, the plain tiling's WM-PSNR at $\alpha_0 = 0.01$ drops to an unacceptably low 18.3 and 16.2 dB on D-NeRF and HyperNeRF, respectively (though the mean r does survive at 0.903 and 0.845). Furthermore, two highly sensitive DyNeRF scenes (*sear_steak* and *cook_spinach*) lose 0.27 and 0.19 dB even at $\alpha_0 = 0.01$, proving that certain scene topologies incur a baseline perturbation cost regardless of the scale.

- **The Upper Bound (Diminishing Returns):** Pushing the scale higher yields severe rendering penalties for minimal extraction gains. Doubling α_0 to 0.20 effectively quadruples the rendering cost while only gaining 1.85 dB in WM-PSNR and a marginal 0.0007 in r.

Table 10: **Extraction fidelity across diverse payload capacities.** This ablation demonstrates that SHield maintains near-perfect payload recovery ($r \geq 0.995$) regardless of the semantic content, spatial resolution, or complexity of the embedded 2D image. **Evaluation Scope:** Mean Watermark PSNR (WM-PSNR) and Pearson correlation r evaluated at the standard operating point (α_0=0.1) across all 150 models. **The Capacity-Redundancy Trade-off:** The mathematically lower WM-PSNR observed for the three massive 217k-element payloads (wm3, wm4, wm5) is a direct consequence of tile repetition limits. Because our adaptive selection rule strictly caps the marked subset K at $\rho_{\max}N$ to preserve rendering fidelity, these larger payloads permit only n_t=1 redundant copy per element on sparser datasets like D-NeRF, compared to n_t=2 copies for the compact 50k-element payloads (wm1, wm2). **The Unmarked Null ($\sqrt{\eta^2}$):** We explicitly report the payload-dependent null floor, the theoretical correlation of a flat, per-channel mean-colour recovery against the target payload, as formally established in our baseline analysis (§ C).

		D-NeRF (7)		HyperNeRF (17)		DyNeRF (6)		
Payload ($H\times W$)	Content	WM-PSNR	r	WM-PSNR	r	WM-PSNR	r	$\sqrt{\eta^2}$
wm1 (128×394)	ICLR logo, flat colour	47.81	0.9999	42.02	0.9996	42.48	0.9997	0.029
wm2 (194×259)	Road/bridge at dusk	48.61	0.9999	42.28	0.9995	42.56	0.9996	0.368
wm3 (350×621)	Bold "SMILE" on white	44.02	0.9998	32.41	0.9963	32.97	0.9972	–
wm4 (386×563)	Detailed colour poster	41.70	0.9995	32.59	0.9956	32.85	0.9961	0.214
wm5 (350×621)	Anime group scene	41.71	0.9995	32.06	0.9955	32.10	0.9959	0.067
Mean		44.77	0.9997	36.27	0.9973	36.59	0.9977	

Table 11: **Empirical justification of anchored optimization on DyNeRF** (α_0=0.1). This pairwise ablation demonstrates that SHield's optimization framework strictly dominates naive plain tiling, drastically reducing the visual rendering penalty across diverse dynamic scenes and payload capacities. **Metric Format:** Evaluated under identical subset selection and testing protocols, we report the rendering penalty (ΔPSNR in dB) for each scene–payload combination formatted as: *naive tiling / SHield optimization*. **Performance Multiplier:** The final column aggregates the per-scene improvement ratio of the mean absolute rendering penalty ($|\Delta$PSNR$|$). Crucially, the anchored optimization yields massive visual fidelity improvements, suppressing the watermark's visual footprint by up to $15.96\times$ on highly complex scenes like FLAME_STEAK and improving 29 of the 30 scene–payload pairs; COFFEE_MARTINI/wm2 is the single exception (-0.076 against -0.066 dB).

Scene	wm1	wm2	wm3	wm4	wm5	ratio
coffee_martini	−0.067 / −0.061	−0.066 / −0.076	−0.268 / −0.184	−0.152 / −0.146	−0.252 / −0.146	1.31×
cook_spinach	−0.111 / −0.106	−0.098 / −0.076	−1.084 / −0.395	−0.539 / −0.316	−0.557 / −0.337	1.94×
cut_roasted_beef	−0.220 / −0.009	−0.041 / +0.037	−0.593 / −0.092	−0.728 / −0.037	−0.549 / −0.045	9.64×
flame_salmon_1	−0.087 / −0.017	−0.062 / −0.020	−0.286 / −0.037	−0.176 / −0.006	−0.491 / −0.055	8.14×
flame_steak	−0.130 / −0.026	−0.083 / −0.011	−0.862 / −0.031	−0.457 / +0.010	−0.705 / −0.062	15.96×
sear_steak	−0.262 / −0.181	−0.138 / −0.133	−1.257 / −0.462	−0.756 / −0.388	−0.667 / −0.403	1.96×

At our optimal balance of $\alpha_0 = 0.1$, the per-scene ΔPSNR footprint remains highly contained, ranging from -0.113 to -0.016 dB on D-NeRF, -0.150 to -0.005 dB on HyperNeRF, and -0.314 to -0.024 dB on DyNeRF.

I SEED SENSITIVITY

Scope of the Seeded Pipeline. In our framework, the key seed governs multiple critical stochastic components: the selection of target Gaussians, the compilation of the training-view pool, the per-step view sampling during optimization, and—for bit payloads—the random message generation and whitening sequences. Consequently, this study evaluates the empirical sensitivity of the entire embedding and extraction pipeline to varying initialization conditions.

Experimental Design. We re-embedded the wm1 image payload using four independent seeds (1–4) on one representative scene per dataset, comparing them against our baseline model (seed 42, which was re-evaluated to confirm exact reproducibility). Additionally, we tested $L = 32$ bit messages on *hook* and *flame_salmon_1* (as no bit protocol is defined for HyperNeRF). As mathematically expected for independent draws without replacement, the independently seeded subsets share approximately

Table 12: **Ablation of the perturbation scale (α_0) on image payloads.** This ablation maps the fundamental trade-off between visual imperceptibility and extraction fidelity across all 150 evaluated models. **Optimization Target:** The hyperparameter α_0 establishes both the initial injection magnitude and the strict anchoring target for the watermarking loss ($\mathcal{L}_{\text{wm}}$). **Trade-off Dynamics:** As the perturbation scale increases, extraction fidelity (WM-PSNR and Pearson r) naturally strengthens, but at the direct expense of the rendering penalty (ΔPSNR). **Selected Operating Point:** The shaded row denotes our standard operating point (α_0=0.10), deliberately chosen as the optimal equilibrium. It guarantees near-perfect extraction (r=0.9980) while bounding the average visual penalty to a highly stable and mathematically minimal -0.063 dB.

α_0	ΔPSNR	WM-PSNR	Pearson r
0.01	-0.011	26.82	0.9770
0.05	-0.022	36.16	0.9969
0.10	-0.063	38.32	0.9980
0.20	-0.250	40.17	0.9986
0.50	-1.377	43.07	0.9994

Table 13: **Algorithmic stability across random seed initializations.** This ablation confirms that SHield's performance is fundamentally grounded in its rigorous mathematical design (anchored optimization and coprime stride) rather than a statistically fortunate initialization. **Metric Format:** We evaluate five independent seeds $(42, 1, 2, 3, 4)$, reporting the mean $\pm$ standard deviation, alongside specific [min, max] bounds where relevant. **Image Payloads (wm1):** Across three representative scenes, we observe exceptionally tight variance in both baseline rendering penalties (ΔPSNR) and extraction fidelity under the severe post-processing attacks established in Table 2. This strictly proves that our optimizer stably converges to an optimal equilibrium regardless of the specific Gaussian subset initially selected by the key. **Bit Payloads (L=32):** Evaluated under the Mark4D comparison framework. Strikingly, across the entire suite of extreme adversarial attacks detailed in Appendix J (including INT8 quantization, up to 80% opacity pruning, catastrophic truncation down to SH degree 0, $5\alpha_0$ parameter noise, and geometric distortion), SHield maintains a flawless 100% decoding accuracy on *every single seed*.

	Metric	*hook* (D-NeRF)	*tamping* (HyperNeRF)	*flame_salmon_1* (DyNeRF)
Image	ΔPSNR (dB)	-0.0226 ± 0.0034	-0.0145 ± 0.0027	-0.0038 ± 0.0135 [-0.0171, $+0.0152$]
	WM-PSNR (dB)	48.69 ± 0.21	44.15 ± 0.52	43.84 ± 0.23
	Pearson r	0.99994 ± 0.00000	0.99981 ± 0.00003	0.99979 ± 0.00001
	r after 30% pruning	0.9331 ± 0.0024	0.9604 ± 0.0014	0.9512 ± 0.0011
	r after 80% pruning	0.5951 ± 0.0107 [0.5875, 0.6133]	0.7550 ± 0.0080	0.7071 ± 0.0049
	r at SH degree 2 (aware)	0.7490 ± 0.0000	0.7489 ± 0.0000	0.7489 ± 0.0000
	r after INT8	0.9992 ± 0.0000	0.9992 ± 0.0000	0.9965 ± 0.0001
	WM-PSNR after INT8 (dB)	38.24 ± 0.11	38.24 ± 0.21	31.99 ± 0.10
Bits	bit accuracy	100% (5/5)	–	100% (5/5)
	PSNR vs. clean (dB)	44.68 ± 0.26	–	43.72 ± 0.48
	SSIM / LPIPS	0.9960 ± 0.0003 / 0.0022 ± 0.0002	–	0.9963 ± 0.0006 / 0.0029 ± 0.0004

K/N of the original seed-42 subset (16%, 10%, and 10% respectively). Table 13 reports the mean $\pm$ standard deviation across all five seeds.

Near-Threshold Variance. While overall fidelity and extraction metrics remain highly stable across seeds, we honestly report that outcomes resting extremely close to the decision threshold τ are inherently seed-dependent. For example, under the aggressive 80% pruning attack on *hook*, the watermark successfully passes the threshold only on the baseline seed 42 ($r = 0.6133$). For seeds 1–4, the correlation falls just short (0.5875–0.5960). Therefore, any per-scene extraction outcome falling within approximately 0.02 of τ must not be interpreted as a seed-independent survival guarantee. Similarly, the sign of the exceptionally small ΔPSNR footprint on *flame_salmon_1* fluctuates depending on the initial seed.

Security Disclaimer. We explicitly emphasize that this evaluates algorithmic variance. It is strictly an empirical sensitivity study and makes no formal claims regarding the cryptographic security, collision resistance, or predictability of the key generation process.

J BIT PAYLOADS: MARK4D COMPARISON SETUP AND PERTURBATION SCALE

Comparison Methodology. To provide a cross-threat-model reference, Table 4 quotes the baseline results from Table 1 of Lee et al. (2026) and positions SHield alongside them. While we align our scene identities and message lengths with their setup, there are five fundamental methodological differences between their evaluation protocol and ours:

(i) **Payload Mechanics and Energy:** SHield carries an L-bit message instead of an image. To prevent constant messages from being erased by the zero-mean step of the embedding (§ 3.2), the message is whitened via a key-derived sequence. It is then mapped to ± 1 symbols and tiled across the same $K \times C_{\text{tot}}$ SH slots using the coprime stride (Appendix E). Because ± 1 symbols have unit magnitude at every slot, whereas image pixel values are zero-mean and mostly small, a bit payload at $\alpha_0 = 0.1$ writes a per-slot RMS of 0.100 (42 runs, 10% of Gaussians). By contrast, the wm1 image payload writes an RMS of only 0.028–0.034 (on 10–26% of Gaussians). This means the bit payload writes $3.34\times$ the per-slot RMS amplitude on average – roughly $11\times$ the energy, which scales as the square – and writes each bit at least $1{,}824$ times (e.g., 608 times per channel on *jumpingjacks*, $L = 64$). We report decoding accuracy strictly; 100% accuracy does not implicitly establish a formal ownership decision rule or false-positive rate.

(ii) **Invisibility Metric:** Lee et al. (2026) measure invisibility by comparing renders of the watermarked model against the *original* model. In contrast, our main evaluation (Table 1) measures ΔPSNR directly against dataset ground truth. Thus, the PSNR values between the two tables are not strictly comparable.

(iii) **Base Models:** While we use the exact same eight D-NeRF and six DyNeRF scene identities, we do not use the authors' trained weights. Our base models were trained independently using the official 4DGS schedules (20k iterations for D-NeRF, 14k for DyNeRF).

(iv) **Viewpoint Construction:** Because D-NeRF natively provides only one view per timestamp, multi-view video must be artificially constructed. Following Mark4D's description, we render 21 fixed viewpoints on a ring around the scene (radius 4.0, elevation -30°) from the trained model, reusing this set across all 300 timestamps (on DyNeRF, we natively use the dataset's test cameras). However, Mark4D splits the D-NeRF ring (using 20 views for training and 1 for evaluation), whereas our base models are trained purely on the original monocular data. SHield optimizes against those original training views, and we average the resulting invisibility across all 21 ring viewpoints.

(v) **Dataset Breakdown:** Because our base models differ, the baseline numbers natively differ. Broken down by dataset, SHield's PSNR against the original models reaches 45.53, 45.68, and 45.30 dB on D-NeRF, and 42.88, 42.14, and 43.00 dB on DyNeRF (at 32, 48, and 64 bits, respectively). Mark4D reports only a single average over both datasets. Note that the model-level distortion attack in § 4.4 utilizes these exact same models.

Perturbation Scale for Bits. Table 14 characterizes bit payloads across three scales: $\alpha_0 \in \{0.01, 0.05, 0.1\}$ on all 42 runs. (While 0.1 remains our default operating point, we make no claim of optimality). At all scales, the measured per-slot RMS matches α_0 to within 0.3%, confirming that no SH values outside the selected subset are altered.

Remarkably, extraction decodes at a perfect 100% on all 42 runs across all three scales, yielding PSNRs of 62.48, 50.32, and 44.29 dB, respectively. Furthermore, this 100% accuracy holds completely firm across our entire standard `watermark_bit_robustness` suite (FP16, INT8, 30–80% opacity pruning, SH degree 2 and 0, and relative noise of $1\times$ and $5 \times \alpha_0$), as well as our strict instantiation of Mark4D's distortion (SH noise, 20% random Gaussian removal).

Because these standard attacks fail to separate the three α_0 settings, we demonstrate that only large *absolute* SH noise successfully degrades the bit accuracy (Table 14, right). Finally, we explicitly note that these highly resilient bit-payload results do not transfer to image payloads, because the optimizer forces a much larger mathematical departure from the plain tiling when α_0 is set as low as 0.01.

Table 14: **Ablation of the perturbation scale (α_0) for bit payloads.** This ablation rigorously quantifies the fundamental trade-off between absolute visual imperceptibility and resilience against catastrophic structural distortion. **Evaluation Scope:** Aggregated across 42 independent runs (spanning 8 D-NeRF and 6 DyNeRF scenes, with message lengths $L \in \{32, 48, 64\}$ at seed 42). **Visual and Physical Footprint:** We report standard rendering metrics (PSNR, SSIM, LPIPS) against the original clean renders following the Mark4D protocol. To explicitly bound the physical magnitude of the modification, we additionally track the absolute per-slot RMS across all SH entries and the total relative perturbation energy ($\|\Delta\mathbf{h}\|/\|\mathbf{h}\|$). **Extreme Stress Testing:** The rightmost columns evaluate extraction under our aggressive Mark4D-style distortion, which simultaneously injects massive absolute SH noise (σ) while catastrophically pruning 95% of the Gaussians. (Note: Under milder $\sigma \leq 0.1$ noise, all configurations successfully decode at 100% with the sole exception of α_0=0.01, which drops to 95.1%). **Operating Point Justification:** While the weakest perturbation (α_0=0.01) achieves an astonishingly invisible 62.48 dB PSNR, its robustness fractures under heavy noise (plunging to 57.8% accuracy at σ=1.0). By contrast, our selected operating point (α_0=0.10, shaded) deliberately accepts a mild visual penalty to establish an incredibly durable payload—maintaining flawless 100% decoding up to σ=0.3, and a highly robust 95.1% even when completely submerged in extreme σ=1.0 noise.

α_0	bit acc.	PSNR (dB)	SSIM	LPIPS	per-slot RMS	file RMS	energy	σ=0.3	σ=1.0
0.01	100%	62.48 [56.99, 67.40]	0.99994	0.00006	0.00998	0.00316	1.2%	72.4%	57.8%
0.05	100%	50.32 [47.81, 53.63]	0.99875	0.00095	0.04988	0.01577	6.0%	99.2%	79.9%
0.10	100%	44.29 [41.57, 47.62]	0.99564	0.00373	0.09981	0.03156	12.0%	100%	95.1%

K Positioning against Hide-in-Motion

Scope of the Comparison. Table 15 strictly lists properties explicitly stated in the Hide-in-Motion paper (Liu et al., 2025) alongside the corresponding properties of SHield. Where their manuscript is silent on critical architectural details, such as whether their extraction is blind, the computational overhead required, or the exact resolution of the hidden image, we deliberately make no speculative comparisons.

Quantitative Reference (Cross-Paradigm). Hide-in-Motion evaluates quantitative performance on four HyperNeRF scenes (*Banana*, *Broom*, *Chicken*, and *3D-Printer*; corresponding to our naming convention: *vrig-peel-banana*, *broom2*, *vrig-chicken*, and *vrig-3dprinter*) and on the exact same six DyNeRF scenes that we utilize.

Tables 16 and 17 place their published numbers alongside ours for these specific scenes, mimicking the format of their Tables I and II. We emphasize that this is provided as a cross-paradigm reference rather than a direct ranking, as the underlying methodologies differ fundamentally across several axes:

- **Differing Base Architectures:** Hide-in-Motion builds upon E-D3DGS (Bae et al., 2024), whereas SHield is built on standard 4DGS. Consequently, absolute rendering scores (PSNR, SSIM, LPIPS) primarily reflect the quality of the base reconstruction rather than the watermarking technique. The only strictly comparable metric is the rendering cost relative to each method's own clean base model (ΔPSNR): Hide-in-Motion incurs -0.068 and -0.029 dB, while SHield incurs a highly comparable -0.049 and -0.127 dB.
- **Extraction Mechanics and Capacity:** Hidden-content recovery metrics are not directly comparable. Hide-in-Motion extracts its payload by rendering a feature map at a specific check viewpoint and decoding it through a learned U-Net, without stating the resolution of the hidden image. By contrast, SHield extracts dense image payloads (50k–217k pixels) directly from the model file in closed form, and its anchored optimization deliberately trades raw recovery fidelity (WM-PSNR) for superior rendering quality (Appendix H).
- **Evaluation Protocol:** The evaluation viewpoints differ (Hide-in-Motion utilizes E-D3DGS settings, whereas we evaluate across the first 20 test views). Furthermore, the baseline tables provided here represent clean, unattacked models; Hide-in-Motion does not provide corresponding data for adversarial post-processing attacks.

Table 15: **Architectural and security paradigm comparison with Hide-in-Motion.** This conceptual breakdown contrasts SHield against the recent Hide-in-Motion (ICRA 2025) baseline, highlighting fundamental differences in embedding mechanics, extraction requirements, and adversarial rigor. **Architectural Simplicity:** Hide-in-Motion requires altering the underlying 3DGS data structure (adding new composite attributes) and must embed the watermark jointly during the initial, expensive scene reconstruction. In stark contrast, SHield operates strictly *post hoc* on a fully trained, completely unmodified 4DGS asset. **Extraction Mechanics:** Hide-in-Motion relies on rendering customized feature maps and passing them through a private, learned U-Net decoder. SHield completely eliminates the need for rendering or proprietary network architectures, instead executing mathematically exact closed-form arithmetic directly against the static parameters. **Adversarial Rigor:** While Hide-in-Motion supports exotic multimodal payloads (audio, video), it reports no evaluation against adversarial post-processing. SHield deliberately trades modality diversity for robust security, providing a rigorously stress-tested defense against severe white-box file manipulations.

	Hide-in-Motion (ICRA 2025)	SHield
Representation	E-D3DGS with an extra composite per-Gaussian attribute that also drives the deformation	unchanged 4DGS model
When embedded	jointly during reconstruction training	post hoc on a trained model
Carrier	learned composite feature	existing canonical SH of key-selected Gaussians
Extraction	feature map rendered at a check viewpoint, private U-Net decoder	closed-form arithmetic against stored reference SH; no rendering or learned decoder
Payloads	2D images; text, QR codes, audio, video via modality-specific decoders	RGB images, bit messages
Attacks evaluated	none reported	quantization, reordering, removal, SH truncation, SH noise (bits), centre perturbation

Table 16: **Quantitative reference comparison against Hide-in-Motion on HyperNeRF.** This comparison isolates the specific visual penalty of watermarking across differing 3D representation paradigms. **Evaluation Protocol:** Hide-in-Motion and its E-D3DGS baseline metrics are quoted directly from Liu et al. (2025). SHield and its unmodified 4DGS baseline metrics are our empirical measurements (averaged across the four shared scenes and five payloads at $\alpha_0{=}0.1$, evaluating the first 20 test views). **Relative Visual Penalty (ΔPSNR):** Absolute rendering metrics (PSNR, SSIM, LPIPS) against ground truth inherently reflect the varying baseline reconstruction capacities of E-D3DGS versus 4DGS. Crucially, ΔPSNR cleanly isolates the specific rendering cost imposed by the watermark itself. Here, SHield imposes a strictly lighter visual footprint (-0.049 dB) than Hide-in-Motion (-0.068 dB). **Extraction Incomparability:** As established in Table 15, the recovery metrics are fundamentally incomparable; Hide-in-Motion leverages a heavy, learned U-Net decoder to reconstruct payloads from rendered feature maps (artificially inflating raw PSNR), whereas SHield strictly reports the fidelity of closed-form, parameter-space arithmetic without neural post-processing.

		Dynamic rendering				Recovery	
Method	Base	PSNR	SSIM	LPIPS	ΔPSNR	PSNR	SSIM
E-D3DGS (Bae et al., 2024)	–	25.641	0.714	0.283	–	–	–
Hide-in-Motion	E-D3DGS	25.573	0.709	0.297	-0.068	50.341	0.998
4DGS (Wu et al., 2024)	–	25.000	0.676	0.394	–	–	–
SHield (ours)	4DGS	24.950	0.674	0.395	-0.049	36.345	0.921

L Temporal Stability (Proxy Analysis)

Theoretical Foundation vs. Rendered Reality. Because SHield is a parameter-space defense, it embeds the payload exactly once into the canonical SH coefficients. Unlike render-domain watermarks that must be carefully synchronized across frames, our method inherently avoids per-frame embedding inconsistencies. However, this is a guarantee about the *storage* of the watermark.

Table 17: **Quantitative reference comparison against Hide-in-Motion on DyNeRF.** This evaluation extends our comparative analysis to the highly dynamic DyNeRF (Neural 3D Video) dataset, contrasting SHield's parameter-space approach against Hide-in-Motion's rendering-based paradigm. **Evaluation Protocol:** As established in Table 16, baseline and Hide-in-Motion metrics are quoted directly from Liu et al. (2025). SHield metrics represent our exact empirical measurements averaged across all six shared scenes and five distinct payloads. **Relative Visual Penalty (ΔPSNR):** While Hide-in-Motion reports a marginally smaller rendering penalty on this specific dataset (-0.029 dB vs. SHield's -0.127 dB), SHield achieves this highly competitive, near-lossless visual fidelity entirely *post hoc*. We secure the model without structurally altering the underlying 3D representation or forcing an expensive joint reconstruction training phase. **Extraction Incomparability:** We strictly reiterate that recovery metrics are fundamentally incomparable; Hide-in-Motion's recovery PSNR is artificially elevated by passing rendered feature maps through a proprietary, learned neural decoder, whereas SHield guarantees recovery via rigorous, closed-form arithmetic directly in the parameter space.

		Dynamic rendering				Recovery	
Method	Base	PSNR	SSIM	LPIPS	ΔPSNR	PSNR	SSIM
E-D3DGS (Bae et al., 2024)	–	31.500	0.938	0.149	–	–	–
Hide-in-Motion	E-D3DGS	31.471	0.935	0.153	-0.029	50.075	0.998
4DGS (Wu et al., 2024)	–	31.192	0.936	0.109	–	–	–
SHield (ours)	4DGS	31.065	0.935	0.110	-0.127	36.591	0.927

Because the deformation network continuously reads these static parameters, and, in architectures like DyNeRF, actively adds time-dependent SH offsets to them at render time, the visual footprint of our static perturbation could theoretically fluctuate as the scene evolves.

Empirical Proxy Analysis. To strictly bound the magnitude of this potential temporal fluctuation, we conduct a targeted proxy analysis. From a fixed camera viewpoint, we trace the rendering cost (the PSNR between the watermarked render and the clean render) continuously over 60 dynamic timestamps.

Results and Scope. The measured temporal variance remains remarkably low. The standard deviation of the PSNR trace is just 0.31 dB on HOOK (D-NeRF), 0.78 dB on TAMPING (HyperNeRF), and a negligible 0.015 dB on FLAME_SALMON (DyNeRF) (illustrated in Figure 9).

We explicitly scope this as a diagnostic proxy designed to verify parameter-level stability, rather than a comprehensive video-consistency benchmark. Consequently, we do not compute generalized temporal metrics like FVD or perceptual flicker over the entire dataset. Nevertheless, these traces strongly suggest that SHield's static parameter embedding maintains highly stable visual fidelity across the temporal domain without requiring motion-specific training machinery.

M Post-hoc 2D Frame Watermarking Baseline

Motivation and Implementation. When designing a defense for neural assets, a natural question arises: why modify the complex 3D/4D parameters of the model itself when one could simply watermark the 2D frames it renders? To rigorously isolate the specific advantage of representation-level embedding, we construct a post-hoc 2D pixel-space baseline.

This baseline directly adds the payload to the rendered training frames. To ensure a fair mechanical comparison, it utilizes the exact same tiling and fold-and-average mathematics as SHield, but mapped across 2D pixel slots rather than SH coefficients. Extraction is then performed by calculating the difference against the clean, unmarked frame. We evaluated this baseline across all 150 scene–payload pairs (Table 18).

Results and Failure Mode. While the 2D baseline successfully recovers the payload on the specific training frames it manually altered (pooled $r = 0.937$), it catastrophically fails on novel test viewpoints (pooled $r = 0.004$). Because a dynamic 4DGS asset allows users to generate novel camera

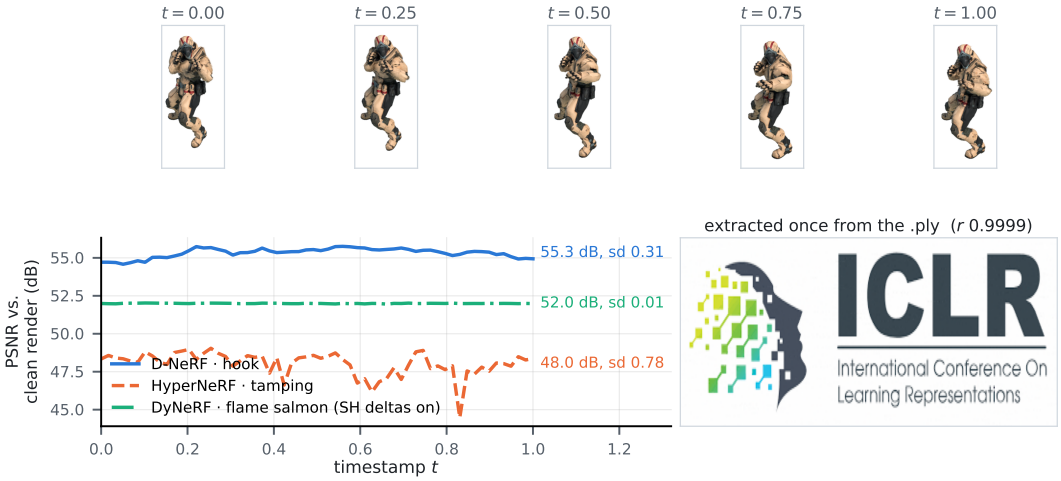

Figure 9: **Temporal stability and time-independent extraction.** This figure demonstrates that SHield successfully avoids the temporal flickering artifacts common to render-domain watermarks, while completely decoupling the extraction process from the time dimension. **Qualitative Consistency (Top):** Rendered frames spanning the watermarked HOOK sequence, demonstrating smooth, artifact-free temporal dynamics. **Quantitative Temporal Stability (Bottom Left):** We track the PSNR between the watermarked and clean renders across 60 sequential timestamps from a fixed camera viewpoint (evaluated on the median scene of each dataset). The exceptionally tight standard deviation (0.015–0.78 dB around means of 48–55 dB) mathematically proves the absence of high-frequency temporal flickering. Crucially, this stability holds even for DyNeRF, where the underlying deformation network actively injects time-dependent SH offsets during rendering. **Time-Invariant Extraction (Bottom Right):** The recovered image payload. Because SHield securely embeds the signal directly into the canonical parameter space, extraction is executed exactly once from the static `.ply` file and is fundamentally independent of the continuous time variable t.

trajectories and render Gaussians from previously unseen perspectives, post-hoc frame watermarking cannot generalize. By contrast, because SHield embeds the watermark directly into the canonical SH parameters, its extraction remains completely invariant to the camera viewpoint or the rendered sequence (pooled $r = 0.998$).

Table 18: **The catastrophic novel-view failure of post-hoc 2D frame watermarking.** This baseline comparison mathematically demonstrates why applying standard 2D image watermarks to training frames fails to secure underlying 3D/4D representations. **Extraction Paradigms:** The 2D baseline injects watermarks directly into the training images; "same-view" extraction is evaluated on these explicitly watermarked training cameras, while "novel-view" extraction targets unseen test cameras. **View-Synthesis Vulnerability:** While the 2D baseline achieves marginal recovery on explicitly memorized training views ($r \approx 0.92$–0.94), it completely collapses during novel-view synthesis (Pearson r plunging to near-zero noise floors of 0.000–0.011). Because 4DGS inherently entangles geometry and view-dependent colour, naive 2D signals do not structurally propagate to novel camera trajectories. **The Parameter-Space Advantage:** In stark contrast, SHield extracts directly from the static model file. By embedding directly into the canonical SH parameters, SHield renders the extraction process fundamentally view-agnostic, guaranteeing near-perfect recovery ($r \geq 0.997$) completely independent of the rendered camera path. **Evaluation Scope:** Averages are reported across the full suite of 30 scenes and 5 payloads.

		D-NeRF (7)		HyperNeRF (17)		DyNeRF (6)	
Method	View type	WM-PSNR	r	WM-PSNR	r	WM-PSNR	r
2D per-frame	Same-view (training)	14.34	0.917	15.28	0.943	15.30	0.943
2D per-frame	Novel-view (test)	6.49	0.010	7.78	0.000	8.30	0.011
SHield (ours)	model file	44.77	0.9997	36.27	0.9973	36.59	0.9977

Scope of the Baseline. We explicitly note that this experiment solely isolates the structural difference between writing into the 3D/4D representation versus marking the 2D output after the fact. It represents naive output-space watermarking and does not represent complex render-domain defenses (such as Mark4D or GuardSplat) that fine-tune the underlying 3D scene against a learned decoder, as those methods do successfully propagate marks to novel views.

N PARAMETER-DISTRIBUTION SHIFT

Motivation and Metric. A critical consideration for any parameter-space watermark is whether the embedding process noticeably shifts the macroscopic distribution of the model's attributes. To quantitatively assess this, we measure the distributional shift between the clean and watermarked models using the 3D-Sinkhorn distance, a metric recently proposed by Ren et al. (2026) for evaluating representation-level steganography.

Implementation Details. For each Gaussian attribute, we compute the per-Gaussian L2 norm (normalized to the $[0, 1]$ range) and bin both the clean and watermarked populations into 100-bin histograms. We then solve the entropy-regularized optimal-transport problem between these histograms using the Sinkhorn algorithm (Peyré & Cuturi, 2019). Note that the mathematical process of placing continuous values into discrete 100-bin histograms naturally introduces quantization error; on this specific scene collection, histogram quantization alone establishes a baseline distance floor of ≈ 0.008. Table 19 reports the mean distances computed across all 150 models.

Table 19: **Statistical imperceptibility of the parameter distribution.** This analysis proves that SHield's parameter-space injection does not introduce detectable statistical anomalies into the canonical 4DGS asset, effectively thwarting distribution-based steganalysis. **Distributional Stealth:** We quantify the global distributional shift by computing the 3D Sinkhorn distance (optimal transport cost) between the clean and watermarked spherical harmonic (SH) attributes. Across all 150 models evaluated at the α_0=0.1 operating point, the distributional deviation is microscopic (strictly < 0.02). This confirms that the watermarked SH coefficients remain statistically indistinguishable from their natural distribution. **Geometric Isolation:** SHield writes only into the SH coefficients, the DC band included: that band holds $1/C_{\text{tot}}$ of the tiled payload and carries its own learned offset $\mathbf{\Delta}_{\text{dc}}$, which is why it appears as a separate row above. The structural attributes (positions, opacities, scales and rotations) are untouched by construction, incur exactly zero distributional penalty, and are omitted from the table.

Attribute	D-NeRF (7)	HyperNeRF (17)	DyNeRF (6)
SH DC (f_{dc})	0.0084	0.0083	0.0085
SH rest (f_{rest})	0.0191	0.0123	0.0088
SH combined	0.0086	0.0083	0.0093

Results and Security Scope. The measured distributional shift remains exceptionally small. The combined SH distance peaks at just 0.0093 across all datasets, resting almost exactly on the quantization floor. Only the rest distance (representing the higher-order SH bands) registers clearly above this baseline, reaching up to $\approx 2.4\times$ the floor.

However, we explicitly scope the security implications of this metric. Because this statistic aggregates and compares macroscopic histograms of per-Gaussian norms, it is designed solely to detect *coarse*, global shifts in attribute distributions. It is not an adversarial subset-localization test. Therefore, while a low 3D-Sinkhorn distance confirms that SHield successfully avoids introducing massive, model-wide statistical artifacts, it does not mathematically guarantee that a determined white-box adversary could not isolate the marked subset through localized inspection.

O WHAT THE OPTIMIZATION CHANGES

Magnitude and Location of Departure. To understand the exact mechanical impact of the anchored optimization, we measure how far the learned perturbation deviates from its plain-tiling

initialization. At our standard operating point ($\alpha_0 = 0.1$), this departure remains remarkably small: on the seven D-NeRF scenes (wm1), the optimized offsets shift by an average of only 1.8% of the embedded magnitude (ranging from 0.7% to 6.5% per scene).

Crucially, this departure is not uniformly distributed. It heavily concentrates on high-opacity Gaussians, while the lowest-opacity eighth of the selected subset remains essentially unchanged. This aligns perfectly with 4DGS rendering mechanics: because nearly transparent Gaussians contribute minimally to the final rendered image, they receive negligible gradients from $\mathcal{L}_{\text{render}}$. Consequently, the optimizer naturally leaves the hidden watermark intact on transparent Gaussians, while carefully adjusting it on dense, highly visible structures to preserve rendering fidelity.

The Failure Mode of Low Amplitudes. We emphasize that this stable, minimal departure is highly specific to the carefully chosen perturbation scale of $\alpha_0 = 0.1$. If the initial amplitude is set too low, the optimizer is forced to aggressively overwrite the watermark to satisfy the rendering loss. At $\alpha_0 = 0.01$, the optimized image perturbation departs from its tiling target by a massive 44%, 229%, and 593% relative RMS on D-NeRF, HyperNeRF, and DyNeRF, respectively. This structural collapse mathematically explains why payloads at $\alpha_0 = 0.01$ fail to transfer robustly to complex real-world scenes, firmly validating our choice of a higher operating scale.

P Adaptive Attacks: Details

The attack is implemented in `attack_finetune.py`: the adversary loads the distributed watermarked file and re-optimizes it with the standard 4DGS photometric objective ($\ell_1 + 0.2(1 - \text{SSIM})$ against ground-truth training views, 4 views per step, 4DGS's own learning rates, deformation network frozen), then saves an ordinary checkpoint. No key, no watermark term, and no knowledge of which Gaussians are marked. Held-out quality is measured on 20 test views against the distributed watermarked model, which is what the adversary starts from; the training-view quality rises in every run, which is why it is not the figure we report.

Table 20 lists every run. With only the SH fine-tuned the centres do not move, re-association matches all key entries, and the fall in r measures erasure of the payload itself. With the geometry fine-tuned too, the marked centres move by 0.32–$0.66\, s_{\text{nn}}$ on average per run, and the fraction still within $\epsilon = 0.01\, s_{\text{nn}}$ matches the measured re-association rate to within 0.0002 on every run: the loss of verification is fully accounted for by centre drift, not by any change to the stored payload.

Attacks on the matcher. Three attacks leave the payload in place and target re-association instead; Table 21 lists them over the 30 wm1 models. *Duplication:* the adversary copies a random fraction of all Gaussians in place, with a seed independent of the key's, and appends the copies after the original rows. Each copy is bit-identical to its original, so the tie-break finds two candidates and the shipped rule drops the entry. Duplicating every Gaussian defeats verification outright at the cost of a file twice the size, while removing nothing; taking the lowest-numbered bit-identical candidate instead of giving up restores full matching at every rate. *Tolerance-induced false matches:* a lone candidate within ϵ is accepted on distance alone, so under 20% random deletion 113 deleted entries are matched to unrelated surviving Gaussians, and under isotropic centre noise 5 entries are mis-assigned at $0.003\, s_{\text{nn}}$ and 360 at $0.01\, s_{\text{nn}}$. Requiring the lone candidate to be bit-identical in opacity, scale and rotation as well removes all of them, and over the 180 cases (six conditions on 30 scenes) neither change loses a single correct match, so the two together give a strictly better matcher than the one evaluated in the main text. *Global rigid transforms:* a translation of $10\, s_{\text{nn}}$ along a fixed direction, a 15° rotation about a fixed oblique axis, or both, invalidates every stored centre. Because verification is already non-blind and SHield changes no geometry, the owner can align the suspect file to the original model first: centroid and principal axes, disambiguated over the four proper sign combinations, then refined by a few ICP steps. This recovers translation fully and rotation on 25 of 30 scenes (84.1% of key entries); on the other five the alignment itself fails, and every correctly aligned file extracts at $r = 0.9997$.

Table 20: **Adaptive fine-tuning attack, per run** (wm1, α_0=0.1).

Scene	params	steps	ΔPSNR	r re-assoc.	r oracle	matched (%)	verifies
bouncingballs	sh	100	+0.04	0.9808	0.9808	100.0	yes
bouncingballs	sh	500	−0.03	0.9257	0.9257	100.0	yes
bouncingballs	sh	2,000	−0.26	0.8035	0.8035	100.0	yes
bouncingballs	sh	5,000	−0.59	0.6202	0.6202	100.0	yes
bouncingballs	sh	10,000	−0.98	0.4782	0.4782	100.0	no
bouncingballs	sh	20,000	−1.51	0.3701	0.3701	100.0	no
bouncingballs	all	2,000	−0.77	0.5905	0.8862	22.7	no
hook	sh	100	+0.00	0.9727	0.9727	100.0	yes
hook	sh	500	−0.03	0.9171	0.9171	100.0	yes
hook	sh	2,000	−0.16	0.7616	0.7616	100.0	yes
hook	all	2,000	−0.97	0.2971	0.8289	9.0	no
flame_steak	sh	100	−0.06	0.9622	0.9622	100.0	yes
flame_steak	sh	500	−0.14	0.9051	0.9051	100.0	yes
flame_steak	sh	2,000	−0.35	0.8126	0.8126	100.0	yes
flame_steak	all	2,000	−2.09	0.4344	0.8805	8.6	no
misc/americano	sh	100	+0.05	0.9637	0.9637	100.0	yes
misc/americano	sh	500	+0.04	0.9031	0.9031	100.0	yes
misc/americano	sh	2,000	−0.02	0.8290	0.8290	100.0	yes
misc/americano	all	2,000	−0.49	0.6214	0.8862	9.4	yes
vrig/vrig-chicken	sh	100	0.00	0.9282	0.9282	100.0	yes
vrig/vrig-chicken	sh	500	−0.05	0.8458	0.8458	100.0	yes
vrig/vrig-chicken	sh	2,000	−0.13	0.6634	0.6634	100.0	yes
vrig/vrig-chicken	sh	5,000	−0.20	0.4312	0.4312	100.0	no
vrig/vrig-chicken	sh	10,000	−0.28	0.3109	0.3109	100.0	no
vrig/vrig-chicken	sh	20,000	−0.39	0.2367	0.2367	100.0	no
vrig/vrig-chicken	all	2,000	−0.66	0.4851	0.7428	13.5	no

Table 21: **Attacks on key re-association** (wm1, α_0=0.1, 30 models). Matched is pooled over key entries; r is the mean over models; passes at τ=0.6. *Shipped*: the matcher of the main results; *lowest-numbered*: ambiguous entries take the lowest-numbered bit-identical candidate; *aligned*: the suspect file is first aligned to the original model.

Attack	Matcher	Matched (%)	r	Passes
Duplicate 10%	shipped	89.96	0.9795	30/30
	lowest-numbered	100.00	0.9997	30/30
Duplicate 50%	shipped	49.91	0.8544	30/30
	lowest-numbered	100.00	0.9997	30/30
Duplicate 100%	shipped	0.00	0.0285	0/30
	lowest-numbered	100.00	0.9997	30/30
Translate	shipped	0.00	0.0285	0/30
	aligned	100.00	0.9997	30/30
Rotate, or rotate + translate	shipped	0.00	0.0285	0/30
	aligned	84.08	0.8390	25/30

Q A Centred Decision Rule

Extraction recovers a zero-mean signal $\hat{\mathbf{z}}_c$ and then adds the payload's stored per-channel mean $\bar{w}_c$ back before scoring (§ 3.3). Those means are part of the key, so when nothing is recovered the comparison is between the key's own stored means and the payload, which is exactly the payload-dependent null floor of Appendix C. The centred rule scores $\hat{\mathbf{z}}_c$ against the payload's zero-mean form directly, pooled over RGB, and centres the payload by its own channel means so that no value stored in the key enters the comparison. With $\hat{\mathbf{z}}_c$ identically zero the correlation is undefined, which we count as a non-detection.

Calibration and evaluation. Re-scoring the 1,800 in-sample trials under both statistics reproduces the uncentred scores exactly. On the 14 calibration scenes the centred statistic separates more widely

than the uncentred one: the largest null is 0.3597 and the smallest true score 0.8072, a gap of 0.4475 against 0.3197. Fixing τ_c at the midpoint of that gap, a rule declared before any evaluation score was computed, gives $\tau_c = 0.5834$. On the 16 held-out scenes (80 models, 240 H_1 and 720 H_0 trials) the centred rule accepts 240/240 true trials and produces 0 false positives, with a largest null score of 0.3577; the uncentred rule at $\tau = 0.6$ gives the same counts with a largest null of 0.4858. All 80 unmarked-model trials are undefined under the centred rule, against 16 under the uncentred one, because the recovery on an unmarked file is exactly zero.

Where it does not help. The centred rule does not remove the SH-truncation passes of Table 7. At degree 1 it passes 36 of 150 models, the same 36 as the uncentred rule and the same payloads. This is the expected outcome rather than a surprise: those passes are structural, produced by the tiling index map (Appendix E), not by the channel-mean floor. For greyscale wm3 the floor is exactly zero, so centring moves its degree-1 score only from 0.5401 to 0.5399. For wm2 the floor was in fact depressing the score, and removing it raises the degree-1 mean from 0.7471 to 0.8084; at degree 0 the centred rule passes 4 wm2 models that the uncentred rule rejects. A statistic that removes the null floor therefore does not by itself make a partially destroyed payload read as destroyed; screening payloads (§ 5) remains necessary under either rule.

R Two Improvements Not Used in the Main Results

Both are reported because they cost nothing and remove a stated limitation; neither is folded back into the numbers reported elsewhere, which use the shipped pipeline throughout.

Embedding without the training images. The anchored optimization minimizes rendering error against ground-truth training views (Eq. 3), so an owner holding the trained model but not its training set cannot run it. Anchoring instead to renders of the owner's own clean model needs nothing beyond the file and leaves the operating point essentially unchanged. Across 5 scenes spanning the three datasets (wm1, α_0=0.1) the mean rendering cost is 0.046 dB against 0.049 dB with ground-truth anchoring, and extraction is unchanged in practice (mean $r = 0.9994$ against 0.9997, minimum 0.9987). The perturbation is placed slightly less precisely, costing 2.7 dB of WM-PSNR (40.5 against 43.3), far from the margin that matters for verification.

Normalizing by the copies observed. Fold-and-average divides every payload element by n_t, the number of tiles written (Eq. 5). After pruning, the slots of entries that could not be matched read exactly zero, so an element is divided by more copies than contributed to it, and because the loss differs from element to element it is not a global scale that Pearson would ignore. Dividing each element by its own observation count recovers part of that loss at no cost: on the wm1 models, mean r rises from 0.9369 to 0.9797 at 30% pruning, from 0.8662 to 0.9252 at 50%, and from 0.6456 to 0.6854 at 80%, where it carries 24 of 30 scenes over τ against 20.

S Extended Limitations

Capacity Allocation Across SH Bands. Our tiling formulation distributes the payload uniformly across all 16 SH coefficients, inherently placing $1/16$ (6.25%) of the data in the degree-0 DC band and the remainder in the higher-order bands. While loading the DC band more heavily could theoretically improve the signal-to-noise ratio for a *blind* extraction network, our empirical ablation (Figure 6) strictly precludes this strategy. At a matched relative perturbation strength (ε), writing to the DC band incurs a massive 13.0 dB greater rendering PSNR penalty compared to the higher-order bands (76.1 vs. 89.2 dB, equating to a $\approx 20\times$ larger mean squared error). This substantial rendering gap remains locked within 0.03 dB across our entire ε sweep, decisively justifying our uniform 1:15 distribution limit.

Key Dependence and Verification Privacy. SHield's verification framework requires a registered, private payload rather than a published, public one. Because the payload itself, the selected Gaussian indices, and the reference SH coefficients all reside exclusively within the owner's private key, verification cannot be executed independently by a blind third party. This privacy is a structural necessity: keeping the selection index securely private is exactly what prevents a wrong-subset

null extraction from falsely matching the true score. Furthermore, we reiterate that our seed study (Appendix I) strictly evaluates algorithmic sensitivity to the random selection; it does not measure or claim cryptographic security regarding the computational difficulty an adversary might face in attempting to recover that hidden selection.

Vulnerability of Index-Based Verification. A naive verifier relying solely on matrix row indices cannot detect a pure row permutation of the model file. Because a permutation preserves the total row count and Gaussian population but entirely misaligns the target arrays, an index-based extraction will return completely meaningless noise. This specific failure mode underscores why our centre-based geometric re-association (Appendix B) is not merely an optional robustness feature, but a strict algorithmic requirement whenever the provenance of a redistributed 4DGS file is uncertain.