SPEND BITS WHERE THE RENDERER LOOKS: OBSERVABILITY-GUIDED RATE–DISTORTION OPTIMIZATION FOR DYNAMIC GAUSSIAN SPLATTING COMPRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Compressing a trained dynamic Gaussian representation requires understanding both how parameter perturbations affect rendered images and when a primitive's contribution can be transferred to a persistent substitute. We introduce **ORDO-GS** (*Observability-guided Rate–Distortion Optimization*), a post-training compression framework informed by *render-space observability*: the expected sensitivity of rendered images to perturbations of the trained representation. Our analysis identifies a pronounced storage–observability mismatch: positions occupy little storage but require high precision to preserve rendering quality, whereas appearance dominates storage and can be compressed more aggressively. ORDO-GS combines render-aware structural reduction with subsequent attribute-specific compression. Structural reduction combines reconstruction-aware importance with trajectory compatibility to determine which primitives to retain and which contributions to transfer. Persistent co-location motivates complementary prune-only and unified delete–merge reduction branches, selected by a lightweight no-fine-tuning probe. For attribute compression, we derive a position-precision prior from a local render-distortion model and adopt channel-specific compact representations for temporal and appearance attributes. At matched primitive counts, ORDO-GS outperforms OMG4 by 0.30 dB while reducing storage by 12.6% on N3DV, and by 0.73 dB with 3.4% less storage on SelfCap. ORDO-GS also transfers to Ex4DGS, achieving consistent rate–distortion improvements across representations.

1 INTRODUCTION

Building on 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023), dynamic representations enable photorealistic reconstruction and real-time rendering of dynamic scenes (Wu et al., 2024; Li et al., 2024; Lee et al., 2024b; Wang et al., 2025). However, representing geometry, motion, and appearance with large collections of primitives imposes substantial storage costs.

Existing compressors combine primitive reduction with compact attribute coding (Lee et al., 2024a; Fan et al., 2024; Chen et al., 2024), while dynamic methods additionally exploit temporal structure (Li et al., 2025; Hu et al., 2025; Lee & Baek, 2025; Lee et al., 2025b). However, storage cost does not necessarily reflect rendering sensitivity, and a primitive's importance does not establish whether its contribution can be preserved by a substitute. We introduce ORDO-GS, a post-training compression framework built on one principle—*spend bits where the renderer looks*. We therefore ask *which parameter changes are visible to the renderer, and when a primitive's contribution can be transferred to a persistent substitute.*

Perspective projection makes the effect of positional changes depend on the viewing direction, while temporal gating and visibility determine when and where a primitive contributes. Different appearance coefficients can also have substantially different effects on rendered images. We analyze these differences through *render-space observability*, which measures the expected sensitivity of rendered images to parameter perturbations across views and time. We use this metric to compare attribute sensitivities and derive a local model of the distortion caused by position quantization.

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.

Three properties inform our compression design. **O1: view-conditioned positional nullspace.** Distinct 3D positions along a viewing ray share the same projected center. This projection nullspace motivates assessing candidate substitutes through their screen-space trajectories across views and time rather than through 3D proximity alone. **O2: low-observability temporal parameters.** Rendered images exhibit low aggregate sensitivity to perturbations of temporal centers and durations. **O3: storage–observability mismatch.** Higher-order view-dependent appearance accounts for most attribute storage but contributes little to normalized observability, whereas positions occupy little storage yet dominate this measure. These attribute sensitivities motivate preserving position precision while allocating fewer bits to temporal and appearance attributes.

Informed by these observations, structural reduction combines reconstruction-aware importance with trajectory compatibility to determine which primitives to retain and which contributions to transfer. Persistent co-location motivates prune-only reduction when substitutes are scarce and delete–merge decimation when they are abundant; a no-fine-tuning probe selects the branch for each model. We then fine-tune the reduced model under teacher supervision. After recovery, attribute compression combines a position-precision prior derived from the render-space distortion model with codec calibration and compact temporal and appearance representations.

Our contributions are:

1. We formulate render-space observability for dynamic Gaussian representations and identify a mismatch between attribute storage and rendering sensitivity.
2. We introduce a structural-reduction strategy in which trajectory compatibility informs primitive retention as well as contribution transfer, with model-wise selection between complementary prune-only and delete–merge branches.
3. We derive an observability-based model of position-quantization distortion, complemented by codec calibration and compact temporal and appearance coding.

2 Related Work

Dynamic Gaussian scene representations. Dynamic Gaussian representations use deformation fields (Wu et al., 2024), native 4D primitives (Yang et al., 2024), or explicit motion trajectories (Li et al., 2024). FreeTimeGS (FTGS) (Wang et al., 2025) assigns temporal support and linear motion to each primitive. FreeTimeGS++ (Lee et al., 2026) improves motion consistency through gated marginalization and a shared neural velocity field. Ex4DGS (Lee et al., 2024b) interpolates positions and rotations between keyframes. Our experiments cover FTGS-family models and Ex4DGS.

Static 3D Gaussian compression. Static 3DGS compression reduces both primitive count and attribute cost. Compact 3DGS (Lee et al., 2024a) combines learned masking, neural appearance, and vector quantization; LightGaussian (Fan et al., 2024) uses significance pruning, SH distillation, and quantization; and EAGLES (Girish et al., 2024a) couples pruning with quantized embeddings. HAC (Chen et al., 2024) entropy-codes attributes from hash-grid context, and LocoGS (Shin et al., 2025) exploits local coherence with neural fields. OMG (Lee et al., 2025a) reduces both primitive count and attribute storage, while RDO-Gaussian (Wang et al., 2024) jointly optimizes pruning and entropy-constrained quantization.

Dynamic Gaussian compression. Dynamic compression can be integrated into the representation or applied after reconstruction. 4DGC (Hu et al., 2025) compresses motion grids and compensating Gaussians under rate–distortion supervision; 4DGCPro (Zheng et al., 2025) uses hierarchical representations and layer-wise rate–distortion training for progressive streaming. Other approaches exploit feature streams (Li et al., 2025), wavelet-transformed trajectories (Lee & Baek, 2025), lifespan-aware pruning (Yuan et al., 2025), or learned entropy models (Liu et al., 2026). Compact representations use deformation fields (Zhang et al., 2025), temporal layers (Xu et al., 2026), static–dynamic decomposition (Wu et al., 2025), or anchors (Huang et al., 2025; Cho et al., 2026; Ho et al., 2026). Frame-wise residuals (Girish et al., 2024b) and group-of-frames coding (Zhang et al., 2026) support streaming. OMG4 (Lee et al., 2025b) is the closest post-training approach, applying sampling, pruning, and merging sequentially before attribute compression. ORDO-GS incorporates trajectory compatibility into survivor selection and uses a model-wise probe to choose between prune-only and delete–merge reduction.

Render-space sensitivity and compression. Optimal Brain Damage and Optimal Brain Surgeon use loss curvature for neural-network pruning (LeCun et al., 1990; Hassibi & Stork, 1993). FisherRF (Jiang et al., 2024) applies Fisher information to radiance-field view selection. PUP 3D-GS (Hanson et al., 2025) approximates reconstruction-error curvature for static pruning. We analyze render sensitivity across dynamic parameter groups to identify which attributes require high precision and which admit stronger compression. We also derive an analytic position-precision prior from a local render-distortion model.

3 PRELIMINARIES

Dynamic Gaussian rendering. Consider a trained dynamic Gaussian representation $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N]$. At a query $q = (c, t)$, where c denotes the camera and t the time, primitive i has a time-dependent center $\boldsymbol{\mu}_i(t)$, covariance $\boldsymbol{\Sigma}_i(t)$, opacity $\alpha_i(t)$, and spherical-harmonic (SH) coefficients that produce view-dependent color $\mathbf{c}_i(\mathbf{d}_{i,c}, t)$. Following 3DGS (Kerbl et al., 2023), camera c projects primitive i to a 2D footprint with mean $\mathbf{u}_{i,q}$ and covariance $\boldsymbol{\Sigma}'_{i,q}$. Its opacity at pixel $\mathbf{p}$ is

$$a_{i,q}(\mathbf{p}) = \alpha_i(t) \exp\left[-\frac{1}{2}(\mathbf{p} - \mathbf{u}_{i,q})^\top \boldsymbol{\Sigma}'^{-1}_{i,q}(\mathbf{p} - \mathbf{u}_{i,q})\right]. \tag{1}$$

Front-to-back alpha compositing produces the rendered RGB value $\mathbf{I}_q(\mathbf{p})$ at pixel $\mathbf{p}$:

$$\mathbf{I}_q(\mathbf{p}) = \sum_{i \in \mathcal{V}_q(\mathbf{p})} \mathbf{c}_{i,q}\, a_{i,q}(\mathbf{p}) \prod_{j<i} (1 - a_{j,q}(\mathbf{p}))\,, \tag{2}$$

where $\mathcal{V}_q(\mathbf{p})$ is the depth-ordered set of overlapping primitives and $\mathbf{c}_{i,q} = \mathbf{c}_i(\mathbf{d}_{i,c}, t)$. We denote the resulting differentiable renderer by $\mathbf{I}_q = \mathcal{R}_q(\boldsymbol{\Theta})$.

Duration-based model. We instantiate this rendering interface with the duration-based parameterization of FreeTimeGS (Wang et al., 2025). For primitive i, $\boldsymbol{\mu}_i$ denotes its position at temporal center τ_i, $\mathbf{v}_i$ its velocity, s_i its temporal duration, and α_i its base opacity. Its time-dependent position and opacity are

$$\boldsymbol{\mu}_i(t) = \boldsymbol{\mu}_i + \mathbf{v}_i(t - \tau_i), \qquad \alpha_i(t) = \alpha_i \exp\left[-\frac{(t - \tau_i)^2}{2s_i^2}\right]. \tag{3}$$

Other dynamic representations can be handled similarly using their native temporal parameterizations, such as deformation fields or sampled trajectories.

4 METHOD

ORDO-GS compresses a trained dynamic Gaussian model through structural reduction followed by attribute compression (fig. 1). Render-space observability characterizes attribute sensitivities and provides an analytic prior for position precision. Structural reduction combines reconstruction-aware importance with multi-view trajectory compatibility to guide primitive retention and contribution transfer. After branch selection, we fine-tune the reduced model under teacher supervision before attribute compression.

4.1 RENDER-SPACE OBSERVABILITY AND COMPRESSION STRUCTURE

Metric and estimation. Let $\boldsymbol{\delta} = \widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}$ be a local compression perturbation, and let $\mathbf{J}_q = \partial \mathbf{I}_q / \partial \boldsymbol{\Theta}$ denote the render Jacobian, with image entries implicitly stacked. We measure the local render distortion by

$$\mathcal{D}(\boldsymbol{\delta}) = \mathbb{E}_q\left[\left\|\mathcal{R}_q(\widehat{\boldsymbol{\Theta}}) - \mathcal{R}_q(\boldsymbol{\Theta})\right\|_2^2\right] = \boldsymbol{\delta}^\top \mathbf{F} \boldsymbol{\delta} + \mathcal{O}(\|\boldsymbol{\delta}\|^3), \tag{4}$$

where $\mathbf{F} \coloneqq \mathbb{E}_q[\mathbf{J}_q^\top \mathbf{J}_q]$ measures how parameter perturbations propagate to rendered images. It is the Gauss–Newton metric of the render distortion and, under independent Gaussian pixel noise, is proportional to the Fisher information (Martens, 2020). Related render sensitivity has been studied for

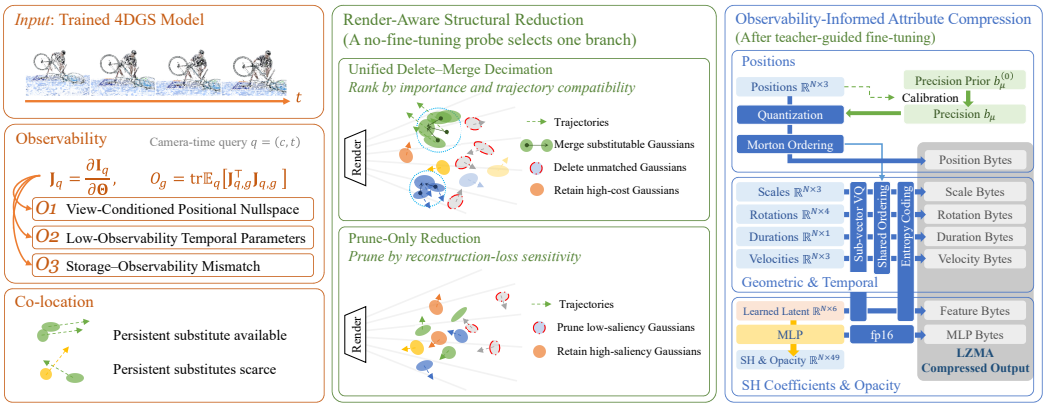


Figure 1: **Overview of ORDO-GS.** Left: observability motivates attribute coding, while co-location motivates complementary reduction regimes. Middle: a no-fine-tuning probe selects between prune-only and delete–merge reduction. Right: after teacher-guided fine-tuning, the reduced model is encoded using calibrated position precision and compact attribute representations.

Table 1: Shares (%) of IQR-scaled render sensitivity and uncompressed model storage, averaged over six FTGS base models per dataset.

Parameter group	N3DV	SelfCap	Storage (%)
Position	97.6	97.0	4.7
Time center	1.80	2.81	1.6
Velocity	0.19	0.06	4.7
Scale/rot./SH_0/opacity	0.39	0.18	17.2
Duration	2.9×10^{-3}	4.9×10^{-3}	1.6
Higher-order SH	7.1×10^{-4}	3.0×10^{-4}	70.3

radiance-field view selection (Jiang et al., 2024) and static 3DGS pruning (Hanson et al., 2025). We use render-space observability to expose the compression structure of dynamic parameter groups.

We summarize the render-space observability of parameter group g by the trace of its Fisher block. This trace can be estimated without constructing either $\mathbf{J}_q$ or $\mathbf{F}$. Let $\mathbf{J}_{q,g}$ denote the columns of $\mathbf{J}_q$ associated with group g. For an image-space Rademacher probe $\boldsymbol{\xi}$ satisfying $\mathbb{E}[\boldsymbol{\xi}\boldsymbol{\xi}^\top] = \mathbf{I}$, Hutchinson's stochastic trace identity (Hutchinson, 1990) gives

$$O_g = \mathrm{tr}(\mathbf{F}_{g,g}) = \mathbb{E}_{q,\boldsymbol{\xi}}\left[\left\|\mathbf{J}_{q,g}^\top \boldsymbol{\xi}\right\|_2^2\right]. \tag{5}$$

For each query–probe pair, the product $\mathbf{J}_{q,g}^\top \boldsymbol{\xi}$ is obtained by differentiating $\langle \mathbf{I}_q, \boldsymbol{\xi}\rangle$, requiring only one backpropagation. Since no target image or residual enters this estimate, O_g captures the intrinsic sensitivity of the trained renderer over the sampled queries. For the cross-group comparison in table 1, we scale each probed-gradient coordinate by the empirical interquartile range (IQR) of the corresponding parameter coordinate across primitives before computing the group energy. This rescaling is used only for cross-group diagnosis. Unlike range or standard deviation, the IQR remains representative under velocity's max-to-median ratios above 10^3 in both datasets.

O1: View-Conditioned Positional Nullspace. For a 3D point $\mathbf{x}$ in camera coordinates, the perspective-projection Jacobian satisfies $\mathbf{J}_\pi \mathbf{x} = \mathbf{0}$. Thus, displacement along the viewing ray leaves the projected position unchanged, although its rendered contribution may still change. This view dependence motivates assessing positional differences between candidate substitutes through their screen-space trajectories across views and time.

O2: Low-Observability Temporal Parameters. Temporal center and duration carry little aggregate observability, motivating low-rate coding of temporal attributes. Velocity is coded separately as a motion channel because of its strongly heavy-tailed distribution.

O3: Storage–Observability Mismatch. Position carries high observability with little storage, whereas higher-order SH coefficients consume most of the model but carry little observability. This mismatch motivates preserving position precision while using a compact shared appearance model.

4.2 Render-Aware Structural Reduction

Structural reduction must assess both a primitive's contribution to reconstruction and the availability of compatible substitutes. A primitive can remain important to rendering yet be redundant when another primitive can preserve its contribution. On the uncompressed teacher, we measure persistent co-location as $R_{\mathrm{col}} = \sum_i \bar{w}_i P_i / \sum_i \bar{w}_i$, where $\bar{w}_i$ is primitive i's accumulated squared footprint energy and P_i measures persistent co-location with the most compatible candidate neighbor, weighted by projected footprint energy. The diagnostic combines temporal co-activity with covariance-aware projected proximity (section A). Sparse persistent co-location motivates pruning based on individual reconstruction importance, whereas abundant co-location motivates jointly considering deletion importance and trajectory compatibility. These complementary regimes motivate our prune-only and delete–merge branches; a quick no-fine-tuning probe selects the branch for each model.

Prune-Only Reduction. The prune-only branch targets scenes where persistent substitutes are scarce. Let $\boldsymbol{\vartheta}_i$ collect the position, opacity, SH_0 appearance, scale, and rotation of primitive i. For query q, let $\ell_q = \mathrm{mean}\,|\mathbf{I}_q - \mathbf{I}_q^{\mathrm{gt}}|$ be the L_1 rendering loss averaged over pixels and color channels. We rank primitive i by

$$s_i^{\mathrm{prune}} = \mathbb{E}_q \left\| \nabla_{\boldsymbol{\vartheta}_i} \ell_q \right\|_2^2 . \tag{6}$$

With $\mathbf{g}_q = \partial \ell_q / \partial \mathbf{I}_q$, the chain rule gives $\nabla_{\boldsymbol{\vartheta}_i} \ell_q = \left(\partial \mathbf{I}_q / \partial \boldsymbol{\vartheta}_i\right)^\top \mathbf{g}_q$. Equation (6) measures the primitive-wise loss-gradient energy. A low score indicates that small perturbations of the primitive have little first-order effect on reconstruction loss over the sampled views and times. Squaring before averaging prevents cancellation across queries. We progressively remove low-scoring primitives, with intermediate recovery and score recomputation between reduction rounds.

Unified Delete–Merge Decimation. The delete–merge branch incorporates the availability of compatible substitutes into primitive selection. For primitive i and a deterministic grid $\mathcal{Q}$ of 24–30 camera–time queries $q = (c, t)$, let $\mathbf{U}_{i,q}$ be the normalized screen position of $\boldsymbol{\mu}_i(t)$. We concatenate these positions into the trajectory feature $\boldsymbol{\Phi}_i = [\mathbf{U}_{i,q}]_{q \in \mathcal{Q}}$ and find its nearest trajectory by

$$d_{ij} = \|\boldsymbol{\Phi}_i - \boldsymbol{\Phi}_j\|_2, \quad j^*(i) = \arg\min_{j \neq i} d_{ij}. \tag{7}$$

Motivated by O1, this feature measures multi-view co-movement through time-varying screen positions.

Let $e_{i,q}$ be the sum of squared RGB residuals bilinearly sampled at $\mathbf{U}_{i,q}$. Here, $V_{i,q}$ is the positive-depth indicator, and $\sigma_{i,q}$ is the normalized isotropic footprint radius. We define

$$\begin{aligned} \delta_i^{\mathrm{delete}} &= \mathbb{E}_{q \in \mathcal{Q}} \left[V_{i,q} \alpha_i(t)^2 \sigma_{i,q}^2 e_{i,q} \right], \qquad \delta_i^{\mathrm{merge}} = \alpha_i^2 d_{i,j^*(i)}^2, \\ \mathrm{cost}_i &= \min\left(\frac{\delta_i^{\mathrm{delete}}}{\mathrm{mean}_k\, \delta_k^{\mathrm{delete}}}, \frac{\delta_i^{\mathrm{merge}}}{\mathrm{mean}_k\, \delta_k^{\mathrm{merge}}} \right). \end{aligned} \tag{8}$$

The deletion cost approximates residual-weighted footprint energy by treating the residual as locally constant over each footprint. The factor $\alpha_i(t)^2 \sigma_{i,q}^2$ accounts for squared opacity and projected area, while $e_{i,q}$ weights this contribution by the local reconstruction error. The merge cost combines trajectory distance with base opacity, assigning low scores to primitives with nearby co-moving candidates. After scene-wise mean normalization, taking the minimum gives a low removal score when either deletion or substitution is favorable under the corresponding cost. We retain the K primitives with the largest cost_i. The original nearest neighbor may not survive reduction. We therefore recompute the nearest match for each removed primitive within the retained set. If their trajectory distance is below τ_{m} times the median nearest-survivor distance, we transfer its contribution to that survivor; otherwise, we delete it. Within each merge cluster, base opacities are combined by alpha union, while the remaining attributes use base-opacity-weighted aggregation following the rules in table S3.

Branch Selection and Recovery. The selection probe runs both branches to the same number of primitives without fine-tuning. We select the branch with lower distortion on the probe queries for the final reduction to K primitives. Both the branch probe and the position bit-depth search exclude held-out evaluation cameras, which are reserved for final evaluation.

We recover the reduced model by fine-tuning through distillation from the frozen uncompressed teacher. Interpolated views between randomly sampled pairs of training cameras use teacher renders as targets, whereas training views use the average of teacher renders and ground truth; the two query types are sampled equally.

4.3 Observability-Informed Attribute Compression

Observability-Guided Position Compression. We derive a position-precision prior from the local render-distortion model in eq. (4). Let $A_{\mu,k}$ be the range of position coordinate k and $O_{\mu,k} = \sum_i [\mathbf{F}]_{\mu_{i,k},\mu_{i,k}}$ its aggregate observability, computed from the unscaled metric. For a common bit depth b_μ, the high-rate uniform quantization model assumes zero-mean, uncorrelated coordinate errors with variance $A_{\mu,k}^2 2^{-2b_\mu}/12$ (Gersho & Gray, 1992). The expected position-induced render distortion $\mathcal{D}_\mu$ and the precision required to satisfy $\mathbb{E}[\mathcal{D}_\mu] \le \varepsilon_\mu$ are approximated by

$$\mathbb{E}[\mathcal{D}_\mu] \simeq \frac{2^{-2b_\mu}}{12} \sum_{k=1}^{3} O_{\mu,k} A_{\mu,k}^2, \qquad b_\mu^{(0)} = \left\lceil \frac{1}{2} \log_2 \frac{\sum_{k=1}^{3} O_{\mu,k} A_{\mu,k}^2}{12\varepsilon_\mu} \right\rceil. \tag{9}$$

This estimate provides an analytic precision prior. We calibrate the final bit depth by rendering the quantized model on training queries and selecting the lowest tested precision that satisfies the prescribed PSNR tolerance. The analytic prior predicts the precision requirement, while render-based calibration determines the final bit depth. Morton ordering and entropy coding reduce the serialized size without changing the quantized coordinates. All attribute streams share the same ordering, avoiding permutation side information.

Compression of Remaining Attributes. Consistent with O2, low-observability temporal centers use low-precision uniform quantization. Correlated geometry and motion vectors use standard sub-vector quantization (SVQ) (Jégou et al., 2011). The resulting index streams are entropy-coded. Motivated by O3, we replace the storage-heavy SH appearance with compact shared MLP decoders, which predict SH coefficients and opacity from positional encodings and per-primitive latent codes. The latents are sub-vector quantized. After teacher-guided fitting, the decoders are evaluated once at load time to recover attributes for the standard Gaussian renderer.

5 Experiments

5.1 Experimental Setup

Datasets and protocol. *N3DV* contains six synchronized multi-view videos with the center camera held out for evaluation (Li et al., 2022). On N3DV, we use the FTGS reproduction from FreeTimeGS++ (Lee et al., 2026), which closely matches the teacher PSNR reported by OMG4 (Lee et al., 2025b), and evaluate at 1352×1014. *SelfCap* comprises multi-view captures with large-amplitude motion (Wang et al., 2025). Following FreeTimeGS++ (Lee et al., 2026), we evaluate six 60-frame clips at 1920×1080, except for the 1080×1080 bike sequences. Our re-evaluations use identical held-out views and resolutions and report three-seed means unless noted. Evaluation views are excluded from branch and bit-depth selection; metrics use the decoded representation.

Baselines and metrics. We compare directly with OMG4 (Lee et al., 2025b) and GIFStream (Li et al., 2025); broader comparisons include Real-Time4DGS (Yang et al., 2024), MEGA (Zhang et al., 2025), ADC-GS (Huang et al., 2025), and RD4DGS (Lee & Baek, 2025) on Ex4DGS (Lee et al., 2024b). We report PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), primitive count, serialized size, and speed. Sizes are decimal MB for the complete LZMA container, including codebooks, decoder weights, and metadata. FPS is synchronized batch-one rendering over the held-out sequence after ten warm-up frames.

Table 2: Quantitative comparison on N3DV. The upper block uses the same FTGS backbone; the lower block gives values compiled from OMG4 under broader settings (Lee et al., 2025b). We measure FPS normalized by the FPS of the same uncompressed teacher. *Teacher reproduced by OMG4; our teacher has PSNR 32.77 dB and LPIPS 0.0324. †excludes Coffee Martini. K denotes thousands of primitives.

Method	PSNR↑	SSIM↑	LPIPS↓	FPS (rel.)↑	Size (MB)↓
FTGS-based models					
FTGS teacher (500K)*	32.80	0.9579	0.0398	1.00×	64.00
OMG4 (FTGS–L, 284K)	32.62	0.9562	0.0411	0.71×	5.87
OMG4 (FTGS–S, 90K)	32.22	0.9516	0.0491	0.87×	2.01
ORDO-GS–L (250K)	32.70	0.9556	0.0378	1.12×	4.361
ORDO-GS–S (90K)	32.52	0.9533	0.0453	1.37×	1.756
Other backbones or protocols (reported)					
MEGA	31.49	–	0.057	–	25.05
GIFStream	31.75	0.938	0.051	–	10.0
ADC-GS–L†	31.67	–	0.061	–	6.57
ADC-GS–S†	31.41	–	0.066	–	4.04
Real-Time4DGS*	31.96	0.946	0.051	–	2087
OMG4 (RT-4DGS)–L	31.99	0.943	0.056	–	5.75
OMG4 (RT-4DGS)–S	31.60	0.939	0.064	–	2.54

Table 3: SelfCap comparison on the 60-frame scenes used by FreeTimeGS++ (Lee et al., 2026).

Method	PSNR↑	SSIM↑	FPS↑	#Prim.↓	Size (MB)↓
GIFStream	24.67	0.8427	118	0.91M	5.77
FTGS++ (uncompressed)	27.14	0.9075	194	0.50M	128
OMG4+interp.	26.34	0.8917	187	0.25M	4.74
ORDO-GS	27.07	0.8963	321	0.25M	4.58

Implementation details. Experiments use one RTX 4090 and 500K-primitive FreeTimeGS-family teachers trained for 30K iterations. We set the merge threshold $\tau_{\mathrm{m}} = 1$. Reduction recovery uses Adam for 16K iterations with a 10^{-3} initial learning rate under cosine decay and $0.8L_1 + 0.2(1 - \mathrm{SSIM})$. Attribute compression uses 12K appearance fitting iterations followed by 6K iterations of joint codebook and decoder refinement. Position precision is codec-calibrated under a 0.05 dB training-view distortion tolerance. We use 8-bit time centers and SVQ-coded geometry and motion. The probe and 250K pipeline take about 40 seconds and 20 minutes per scene, respectively.

5.2 Quantitative Comparison

N3DV. We use OMG4 (Lee et al., 2025b) as the direct FTGS-based baseline, evaluating its official checkpoints and converting their sizes to decimal MB. Table 2 shows that at the matched 90K count, ORDO-GS outperforms OMG4-S by 0.30 dB with 12.6% less storage and $1.6\times$ the rendering throughput; its 250K model also exceeds the 284K OMG4-L model by 0.08 dB with 25.7% less storage.

SelfCap. We compare with the FTGS++ teacher (Lee et al., 2026), OMG4 (Lee et al., 2025b) at a matched 250K count, and the best of three GIFStream anchor densities (Li et al., 2025). For a controlled comparison, *OMG4+interp.* adapts OMG4 to the same teacher, primitive budget, and 16K-step interpolated-view recovery schedule. Table 3 shows that ORDO-GS exceeds OMG4+interp. by 0.73 dB with 3.4% less storage. The largest PSNR gains over OMG4+interp. occur on *dance1* and *dance2*, at 2.54 and 1.98 dB, respectively. ORDO-GS also exceeds the best-PSNR GIFStream setting by 2.40 dB with $3.6\times$ fewer primitives. Figure 2 shows corresponding held-out comparisons on the large-motion sequences. Structural rate–distortion curves across primitive budgets are provided in fig. S1. Full-sequence quality and temporal error variation are evaluated in section G.

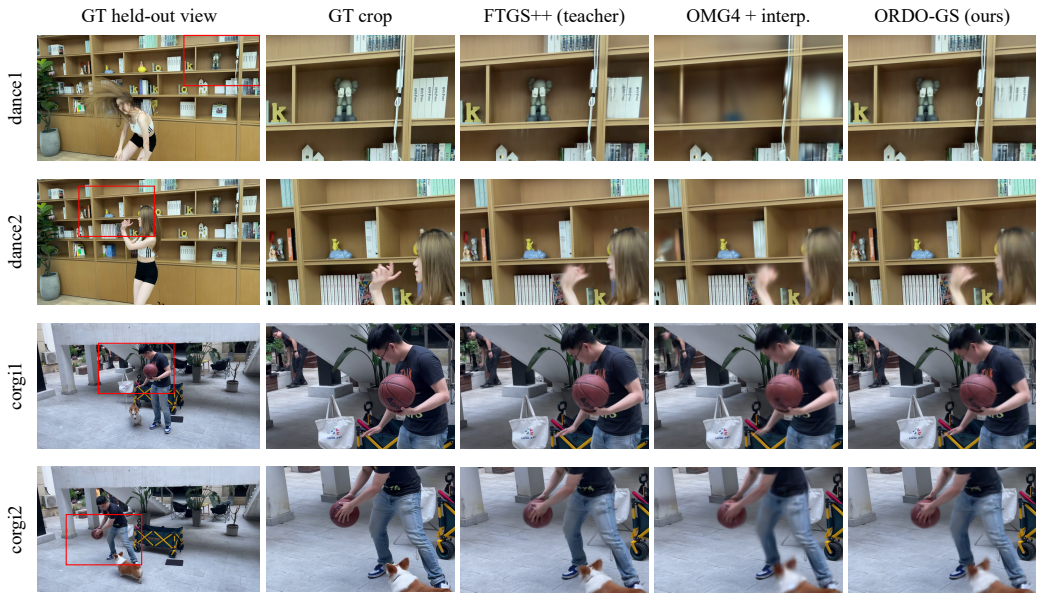


Figure 2: Held-out SelfCap comparisons using 500K primitives for the FTGS++ teacher and 250K for OMG4+interp. and ORDO-GS. The first column marks shared crops. ORDO-GS better preserves thin structures, object boundaries, and moving foreground details.



Figure 3: Complementary reduction regimes at 250K primitives with other settings fixed. Prune-only is selected for *flame steak* (low persistent co-location), and delete–merge for *dance1* (high persistent co-location). Held-out error maps show $E_{\text{prune}} - E_{\text{merge}}$ (per-pixel RGB MSE): red favors delete–merge and blue favors prune-only, using a shared symmetric-log scale.

5.3 Analysis and Ablations

Reduction branches. Figure 3 illustrates complementary reduction regimes. At 250K, delete–merge outperforms prune-only by $2.450/1.986$ dB on *dance1/2*, the two scenes with the highest R_{col}. At 90K, table 4 shows that loss-gradient pruning performs best on *cook spinach*, whereas footprint-based delete–merge performs best on *corgi1* and *dance1*. On *dance1*, footprint scoring improves prune-only PSNR by 3.59 dB, and the corresponding delete–merge configuration provides a further 0.73 dB gain. Screen-space trajectories remain within 0.28 dB of the best tested merge feature across four SelfCap scenes (table S5).

Branch selection. We test whether branch selection before fine-tuning predicts relative quality after recovery. Across 12 scenes at 250K primitives, probe-based selection improves mean PSNR by 0.360 dB over always using prune-only and by 0.061 dB over always using delete–merge (table 5). Its mean PSNR is only 0.021 dB below that obtained by recovering both branches and choosing the better one for each scene.

Table 4: Joint ablation of the deletion score and reduction operator at 90K primitives. All variants use the same 6K-step recovery schedule. PSNR is averaged over three seeds. Footprint denotes the residual-weighted footprint energy cost.

	Prune only		Delete–merge	
Scene	Loss-grad.	Footprint	Loss-grad.	Footprint
cook spinach	**33.370**	33.029	33.147	32.820
corgi1	27.122	27.118	27.164	**27.385**
dance1	22.419	26.005	22.916	**26.739**

Table 5: Component ablations relative to the full configuration; Δ denotes the PSNR difference (ablated minus full). The uniform-precision variant retains the same Morton ordering and entropy codec.

Full-design component	Ablated variant	Effect of Ablation	Evaluation
Probe-based branch selection	Always prune-only	-0.360 dB	12 scenes, 250K
	Always delete–merge	-0.061 dB	12 scenes, 250K
Interpolated-view recovery	Training views only	-0.216 dB	SelfCap
Teacher-guided recovery	Ground truth only	-0.540 dB	SelfCap
Calibrated position precision	Uniform 16-bit precision	-0.377 dB; 0.157 MB smaller	SelfCap
Position/time codec	FP16 streams	9.6–13.2% larger; $\lvert\Delta\rvert \leq 0.07$ dB	12 scenes, 250K
Shared appearance decoder	Explicit decoded SH	2.08–2.33$\times$ the size; $\lvert\Delta\rvert \leq 0.05$ dB	N3DV

Compression. Observability-guided precision gains 0.983 dB on *dance1* and 0.377 dB on average across SelfCap at a 0.157 MB size increase. With all non-position attributes fixed, predicted and measured position-quantization distortions have cross-scene Spearman correlations of 0.944–0.993 across 12 scenes at six precisions from 12 to 22 bits (section D). Table 5 also verifies compact position/time and appearance coding, interpolated views, and teacher guidance.

5.4 Cross-Representation Transfer

Under a shared reproduced Ex4DGS (Lee et al., 2024b) teacher and protocol, ORDO-GS gains 0.28 dB over OMG4 (Lee et al., 2025b) with 8.7% less storage (table 6), demonstrating portability across dynamic Gaussian representations. Published RD4DGS results are included for reference.

Table 6: Cross-representation transfer on N3DV. Ex4DGS and RD4DGS values follow RD4DGS (Lee & Baek, 2025) (L1/L6: aggressive/high-fidelity); OMG4 and ORDO-GS share our reproduced teacher.

Method	PSNR↑	Size (MB)↓
Ex4DGS	32.11	115.00
RD4DGS (L6)	29.66	11.06
RD4DGS (L1)	27.04	1.26
OMG4 (Ex4DGS)	31.57	15.42
ORDO-GS (Ex4DGS)	31.85	14.07

6 Conclusion

We presented ORDO-GS, a post-training framework for observability-guided rate–distortion optimization of dynamic Gaussian representations. Render-space observability characterizes attribute sensitivity and yields an analytic position-precision prior. Structural reduction combines reconstruction importance with trajectory compatibility, while a model-wise probe selects between complementary reduction branches. Together with compact attribute coding, these designs improve rate–distortion trade-offs on N3DV and SelfCap and transfer to Ex4DGS. Our reduction scores rely on fixed temporal sampling; adaptive query placement could better cover short-lived primitives and visibility changes.

AI USE STATEMENT

We used generative AI tools to assist with manuscript drafting and editing, literature search, and reference checking. These tools also provided feedback on experimental design and the interpretation of results. The authors reviewed the AI-assisted material and take responsibility for the final content.

REFERENCES

Yihang Chen, Qianyi Wu, Weiyao Lin, Mehrtash Harandi, and Jianfei Cai. Hac: Hash-grid assisted context for 3d gaussian splatting compression. In *European Conference on Computer Vision*, pp. 422–438, 2024.

Woong Oh Cho, In Cho, Seoha Kim, Jeongmin Bae, Youngjung Uh, and Seon Joo Kim. 4d scaffold gaussian splatting with dynamic-aware anchor growing for efficient and high-fidelity dynamic scene reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 3363–3371, 2026.

Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, Dejia Xu, and Zhangyang Wang. Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps. In *Advances in Neural Information Processing Systems*, volume 37, pp. 140138–140158, 2024.

Allen Gersho and Robert M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, 1992.

Sharath Girish, Kamal Gupta, and Abhinav Shrivastava. Eagles: Efficient accelerated 3d gaussians with lightweight encodings. In *European Conference on Computer Vision*, pp. 54–71, 2024a.

Sharath Girish, Tianye Li, Amrita Mazumdar, Abhinav Shrivastava, David Luebke, and Shalini De Mello. QUEEN: Quantized efficient encoding of dynamic gaussians for streaming free-viewpoint videos. In *Advances in Neural Information Processing Systems*, volume 37, pp. 43435–43467, 2024b.

Alex Hanson, Allen Tu, Vasu Singla, Mayuka Jayawardhana, Matthias Zwicker, and Tom Goldstein. Pup 3d-gs: Principled uncertainty pruning for 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5949–5958, 2025.

Babak Hassibi and David G. Stork. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in Neural Information Processing Systems*, volume 5, 1993.

Cheng-Yuan Ho, He-Bi Yang, Jui-Chiu Chiang, Yu-Lun Liu, and Wen-Hsiao Peng. TED-4DGS: Temporally activated and embedding-based deformation for 4dgs compression. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6371–6380, 2026.

Qiang Hu, Zihan Zheng, Houqiang Zhong, Sihua Fu, Li Song, Xiaoyun Zhang, Guangtao Zhai, and Yanfeng Wang. 4DGC: Rate-aware 4D gaussian compression for efficient streamable free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 875–885, 2025.

He Huang, Qi Yang, Mufan Liu, Yiling Xu, and Zhu Li. Adc-gs: Anchor-driven deformable and compressed gaussian splatting for dynamic scene reconstruction. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 1179–1187, 2025.

Michael F. Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics—Simulation and Computation*, 19(2):433–450, 1990.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

Wen Jiang, Boshu Lei, and Kostas Daniilidis. Fisherrf: Active view selection and mapping with radiance fields using fisher information. In *European Conference on Computer Vision*, pp. 422–440, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023.

Yann LeCun, John S. Denker, and Sara A. Solla. Optimal brain damage. In *Advances in Neural Information Processing Systems*, volume 2, 1990.

Hyeongmin Lee and Kyungjune Baek. Temporal smoothness-aware rate-distortion optimized 4d gaussian splatting. In *Advances in Neural Information Processing Systems*, volume 38, pp. 132174–132204, 2025.

Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. Compact 3d gaussian representation for radiance field. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21719–21728, 2024a.

Joo Chan Lee, Jong Hwan Ko, and Eunbyung Park. Optimized minimal 3d gaussian splatting. In *Advances in Neural Information Processing Systems*, volume 38, pp. 150615–150639, 2025a.

Junoh Lee, Changyeon Won, Hyunjun Jung, Inhwan Bae, and Hae-Gon Jeon. Fully explicit dynamic gaussian splatting. In *Advances in Neural Information Processing Systems*, volume 37, pp. 5384–5409, 2024b.

Lucas Yunkyu Lee, Soonho Kim, Youngwook Kim, Sangmin Kim, and Jaesik Park. FreeTimeGS++: Secrets of dynamic gaussian splatting and their principles. *arXiv preprint arXiv:2605.03337*, 2026.

Minseo Lee, Byeonghyeon Lee, Lucas Yunkyu Lee, Eunsoo Lee, Sangmin Kim, Seunghyeon Song, Joo Chan Lee, Jong Hwan Ko, Jaesik Park, and Eunbyung Park. Optimized minimal 4d gaussian splatting. *arXiv preprint arXiv:2510.03857*, 2025b.

Hao Li, Sicheng Li, Xiang Gao, Abudouaihati Batuer, Lu Yu, and Yiyi Liao. Gifstream: 4d gaussian-based immersive video with feature stream. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21761–21770, 2025.

Tianye Li, Mira Slavcheva, Michael Zollhöfer, 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*, pp. 5511–5521, 2022.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8508–8520, 2024.

Mufan Liu, Qi Yang, He Huang, Wenjie Huang, Zhenlong Yuan, Zhu Li, Yiling Xu, and Yunfeng Guan. Light4gs: Lightweight compact 4d gaussian splatting generation via context model. *IEEE Transactions on Circuits and Systems for Video Technology*, 36(6):7992–8005, 2026.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Guy M. Morton. A computer oriented geodetic data base and a new technique in file sequencing. Technical report, IBM Ltd., Ottawa, Canada, 1966.

Seungjoo Shin, Jaesik Park, and Sunghyun Cho. Locality-aware gaussian compression for fast and high-quality rendering. In *International Conference on Learning Representations*, 2025.

Henan Wang, Hanxin Zhu, Tianyu He, Runsen Feng, Jiajun Deng, Jiang Bian, and Zhibo Chen. End-to-end rate-distortion optimized 3d gaussian representation. In *European Conference on Computer Vision*, pp. 76–92, 2024.

Yifan Wang, Peishan Yang, Zhen Xu, Jiaming Sun, Zhanhua Zhang, Yong Chen, Hujun Bao, Sida Peng, and Xiaowei Zhou. Freetimegs: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21750–21760, 2025.

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.

Jiahao Wu, Rui Peng, Zhiyan Wang, Lu Xiao, Luyang Tang, Jinbo Yan, Kaiqiang Xiong, and Ronggang Wang. Swift4d: Adaptive divide-and-conquer gaussian splatting for compact and efficient reconstruction of dynamic scene. In *International Conference on Learning Representations*, 2025.

Hanjie Xu, Yuanxing Duan, Qiyu Dai, Ge Li, Baoquan Chen, and He Wang. Layered 4d-rotor gaussian splatting: A compressed representation for long dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18958–18967, 2026.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations*, 2024.

Yuheng Yuan, Qiuhong Shen, Xingyi Yang, and Xinchao Wang. 1000+ fps 4d gaussian splatting for dynamic scene rendering. In *Advances in Neural Information Processing Systems*, volume 38, pp. 83135–83153, 2025.

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.

Wenkang Zhang, Yan Zhao, Qiang Wang, Zhixin Xu, Li Song, and Zhengxue Cheng. D-FCGS: Feedforward compression of dynamic gaussian splatting for free-viewpoint videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 16361–16369, 2026.

Xinjie Zhang, Zhening Liu, Yifan Zhang, Xingtong Ge, Dailan He, Tongda Xu, Yan Wang, Zehong Lin, Shuicheng Yan, and Jun Zhang. Mega: Memory-efficient 4d gaussian splatting for dynamic scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27828–27838, 2025.

Zihan Zheng, Zhenlong Wu, Houqiang Zhong, Yuan Tian, Ning Cao, Lan Xu, Jiangchao Yao, Xiaoyun Zhang, Qiang Hu, and Wenjun Zhang. 4DGCPro: Efficient hierarchical 4D gaussian compression for progressive volumetric video streaming. In *Advances in Neural Information Processing Systems*, volume 38, pp. 100860–100885, 2025.

APPENDIX

This appendix provides the formal definition of persistent co-location, dataset and evaluation protocols, implementation details, and additional experimental results.

A PERSISTENT CO-LOCATION DIAGNOSIS

A.1 PURPOSE AND SCOPE

Persistent co-location, denoted by R_{col}, characterizes whether primitives have neighbors that remain spatially close in projection while active at the same times. We compute this scene-level diagnostic on the frozen, uncompressed teacher without target images or reconstruction residuals.

A.2 DEFINITION

We evaluate the full teacher on a fixed set $\mathcal{Q}_{\mathrm{col}}$ of 48 camera–time queries $q = (c, t)$, where c is a training camera and t is a sampled time. We stride the training cameras by $\max(1, \lfloor N_c/8 \rfloor)$ and the frames by $\max(1, \lfloor N_f/6 \rfloor)$, retaining the first eight cameras and six times, where N_c and N_f are the training camera and frame counts. For each primitive i, we compute a candidate set $\mathcal{N}_i$ containing its eight nearest other primitives in canonical 3D space. The candidate set is computed once and remains fixed across all queries. All teacher primitives participate in this computation; there is no primitive subsampling or random seed.

For this diagnosis, we express the projected mean $\mathbf{u}_{i,q}$ and covariance $\mathbf{\Sigma}'_{i,q} \in \mathbb{R}^{2\times 2}$ of primitive i at query q in normalized image coordinates rather than pixel coordinates, using the elliptical weighted average (EWA) projection used by 3DGS (Kerbl et al., 2023). We define its query weight as

$$\begin{aligned} w_{i,q} &= \alpha_i(t)^2 \sqrt{\det \mathbf{\Sigma}'_{i,q}}\, V_{i,q}, \\ V_{i,q} &= \mathbf{1}\{z_{i,q} > 10^{-2}\}, \end{aligned} \tag{S1}$$

where $\alpha_i(t)$ is the time-gated opacity, $z_{i,q}$ is camera-space depth, and $V_{i,q}$ is a positive-depth indicator. Thus, $w_{i,q}$ combines the squared opacity and projected footprint area of an active primitive without using a reconstruction residual.

For a candidate pair (i, j), let $\Delta\mathbf{u}_{ij,q} = \mathbf{u}_{i,q} - \mathbf{u}_{j,q}$. We measure their projected proximity using the covariance-aware kernel

$$\kappa_{ij,q} = \exp\left[-\frac{1}{2}\Delta\mathbf{u}_{ij,q}^{\mathsf{T}}(\mathbf{\Sigma}'_{i,q} + \mathbf{\Sigma}'_{j,q})^{-1}\Delta\mathbf{u}_{ij,q}\right]. \tag{S2}$$

The summed covariance scales the displacement between projected centers according to the extent and orientation of their footprints.

Using $\epsilon = 10^{-9}$ for numerical stability, the directed co-location score of i with candidate j is

$$\begin{aligned} P_{i\to j} &= \frac{\sum_{q\in\mathcal{Q}_{\mathrm{col}}} \min(w_{i,q}, w_{j,q})\, \kappa_{ij,q}}{\sum_{q\in\mathcal{Q}_{\mathrm{col}}} w_{i,q} + \epsilon}, \\ P_i &= \max_{j\in\mathcal{N}_i} P_{i\to j}. \end{aligned} \tag{S3}$$

The scene-level diagnostic is the contribution-weighted mean

$$R_{\mathrm{col}} = \frac{\sum_i \bar{w}_i P_i}{\max(\sum_i \bar{w}_i, \epsilon)}, \qquad \bar{w}_i = \sum_{q\in\mathcal{Q}_{\mathrm{col}}} w_{i,q}. \tag{S4}$$

Because $0 \le \kappa_{ij,q} \le 1$ and $\min(w_{i,q}, w_{j,q}) \le w_{i,q}$, both P_i and R_{col} lie in $[0, 1]$. P_i summarizes weighted co-location with the best fixed candidate neighbor. R_{col} aggregates these scores, emphasizing primitives with greater accumulated projected footprint energy.

A.3 INTERPRETATION

Equation (S3) combines three aspects of persistent co-location. The kernel $\kappa_{ij,q}$ favors nearby projected centers relative to their footprint extents. The shared weight $\min(w_{i,q}, w_{j,q})$ emphasizes queries where both primitives are active. Taking the maximum over j after summing over queries favors one consistent neighbor across views and time. Together, these terms provide a bounded proxy for the availability of persistent substitutes.

During decimation, trajectory proximity serves as a geometric proxy for candidate compatibility: it informs survivor selection and subsequent matching within the retained set. The scene-level R_{col} diagnosis motivates the reduction regimes; the no-fine-tuning probe selects the branch for each model.

A.4 EMPIRICAL ANALYSIS

Table S1 reports the deterministic full-model diagnosis for all twelve scenes under the fixed 48-query protocol.

The *dance1* and *dance2* scenes have the highest co-location scores and show the largest gains from the delete–merge branch over prune-only reduction (section 5.3). This agreement supports the use of persistent co-location to motivate the two reduction regimes.

B DATASET AND EVALUATION PROTOCOLS

B.1 N3DV

Neural 3D Video (N3DV) contains synchronized multi-view recordings of six dynamic indoor scenes (Li et al., 2022): *coffee martini*, *cook spinach*, *cut roasted beef*, *flame salmon 1*, *flame steak*, and *sear steak*. We abbreviate *flame salmon 1* as *flame salmon* below. We use source frames $[0, 300)$ at 30 fps and the 18-view protocol used by our FTGS-family teachers: camera 0 is held out for evaluation and cameras 1–17 are used for training. Native 2704×2028 images are downsampled by a factor of two to 1352×1014 for both training and evaluation. We use the same held-out camera, temporal interval, and resolution for every method in a comparison.

The primary N3DV experiments use the FTGS configuration reproduced in the FreeTimeGS++ codebase (Lee et al., 2026), matching the representation used by the directly comparable OMG4 checkpoints. The teacher is trained for 30K iterations and contains 500K primitives. Cross-representation experiments use the same N3DV views and frames with the corresponding Ex4DGS teacher.

B.2 SELFCAP

We follow the six-clip SelfCap evaluation subset used by FreeTimeGS++ (Lee et al., 2026): *corgi1*, *corgi2*, *dance1*, *dance2*, *bike1*, and *bike2*. The labels *dance1* and *dance2* refer to two intervals from the released *hair* capture. Each clip contains 60 consecutive source frames at 60 fps. The corgi and hair captures contain 24 synchronized cameras, of which 23 are used for training; the bike capture contains 22 cameras, of which 21 are used for training. The exact temporal intervals, held-out cameras, and evaluation resolutions are listed in table S2.

All methods are evaluated on the decoded model over the full held-out sequence. No held-out camera enters observability estimation, branch selection, position-precision calibration, checkpoint selection, or recovery. Interpolated-view distillation uses pairs of training cameras.

B.3 REPEATED RUNS AND REPORTED STATISTICS

Unless explicitly marked as a value quoted from a published paper or an official released checkpoint, our primary N3DV and SelfCap re-evaluations and matched controlled comparisons report the arithmetic mean over seeds $\{0, 1, 2\}$. The seeds control stochastic optimization, view sampling, and K-means initialization. The full-model co-location diagnosis in section A is deterministic and is therefore evaluated once. When a supplementary ablation uses a different repetition count, its table caption states that count explicitly.

Table S1: Persistent co-location on the full 500K teachers for the SelfCap (Wang et al., 2025) and N3DV (Li et al., 2022) scenes, sorted by R_{col}. The diagnosis uses all teacher primitives over a fixed camera–time grid, without primitive subsampling or randomness.

Scene	Dataset	R_{col} (%)
dance1	SelfCap	95.2
dance2	SelfCap	41.2
cut roasted beef	N3DV	23.2
bike2	SelfCap	21.9
corgi1	SelfCap	21.4
bike1	SelfCap	19.7
corgi2	SelfCap	15.6
flame steak	N3DV	12.6
sear steak	N3DV	12.6
cook spinach	N3DV	10.3
flame salmon	N3DV	3.0
coffee martini	N3DV	0.5

Table S2: SelfCap evaluation protocol. Frame intervals are half-open source frame ranges, and camera indices follow the processed dataset convention. All non-held-out cameras are used for training.

Clip	Capture	Frames	Train cams	Eval. cam	Resolution
corgi1	corgi	$[200, 260)$	23	7	1920×1080
corgi2	corgi	$[2950, 3010)$	23	7	1920×1080
dance1	hair	$[4120, 4180)$	23	15	1920×1080
dance2	hair	$[5530, 5590)$	23	15	1920×1080
bike1	bike	$[8900, 8960)$	21	9	1080×1080
bike2	bike	$[30020, 30080)$	21	9	1080×1080

C Implementation Details

C.1 Environment, Teachers, and Metrics

Each run uses one NVIDIA RTX 4090 (24 GB), PyTorch 2.9.1, and CUDA 12.8. The FTGS-family teachers contain 500K primitives with degree-3 SH appearance and are trained for 30K iterations. N3DV and SelfCap teachers use the same image resolution during training and evaluation, as specified in section B. We use the author-released FreeTimeGS++ codebase as a reproducible FTGS-family implementation (Lee et al., 2026). N3DV follows its reproduced FTGS configuration, whereas SelfCap uses the released FreeTimeGS++ configuration with gated marginalization and motion-guided initialization. The underlying primitive motion remains linear in time. We preserve the released per-scene teacher configurations rather than retuning them for compression: N3DV uses unbounded duration, while SelfCap uses a duration cap of one except for *corgi2*, which uses four.

We compute PSNR from RGB mean-squared error after clamping renders to $[0, 1]$. SSIM (Wang et al., 2004) and LPIPS (Zhang et al., 2018) are evaluated per frame and then averaged over the held-out sequence. All LPIPS values use the AlexNet backbone. For our N3DV evaluations, we follow the input convention of the released OMG4 code: RGB images in $[0, 1]$ are passed to LPIPS with `normalize=False`. Storage is reported in decimal MB (10^6 bytes) for the complete compressed model container, including quantized streams, codebooks, entropy-coding tables, MLP weights, latent codes, and quantization metadata.

C.2 Query Sampling and Observability Estimation

Structural reduction uses deterministic camera–time grids with stage-specific temporal sampling. For N_c training cameras, we stride the cameras by $\max(1, \lfloor N_c/4 \rfloor)$. Loss-gradient pruning samples every 15th frame, yielding 20 temporal samples on N3DV and four on SelfCap. Delete–

merge scoring and trajectory matching instead stride the N_f frames by $\max(1, \lfloor N_f/6 \rfloor)$, producing 24–30 camera–time queries in our datasets. For intrinsic observability, we draw one image-space Rademacher probe per query. Vector–Jacobian products compute the corresponding parameter gradients without materializing the render Jacobian. Only training cameras enter these estimates. The 48-query co-location grid is a separate, denser diagnostic protocol described in section A.

Cross-group normalization and velocity tails. For the group-level observability comparison, we multiply each probed gradient coordinate by the empirical interquartile range of the corresponding parameter coordinate across primitives before accumulating squared group energy. This normalization is used only to compare heterogeneous parameter groups. Velocity parameters are strongly heavy-tailed in both datasets, so range- or standard-deviation scaling can be dominated by a small number of extreme primitives. The IQR characterizes the scale of the distribution body and preserves the group-level observability pattern across the FTGS and FreeTimeGS++ teachers. The FreeTimeGS++ teacher configurations exhibit fewer extreme velocity values than the corresponding FTGS reproductions, while the heavy-tailed structure and the resulting observability conclusions remain unchanged.

C.3 Structural Reduction

Prune-only. We apply prune-only reduction progressively. Each round recomputes the primitive-wise loss-gradient energy on the deterministic query grid, retains 60% of the current primitives unless the target count is reached first, and then recovers the reduced model before the next round. The score is computed in the model's native parameterization without the IQR rescaling used only for the group-level observability diagnosis.

Delete–merge decimation. Screen coordinates are normalized by image width and height and clamped to $[-0.5, 1.5]^2$. Nearest trajectories are found by an exact, blockwise Euclidean search. The deletion and merge costs in eq. (8) are each normalized by their scene-wise mean before taking their minimum. Algorithm S1 details survivor selection and subsequent merge/delete execution, with transfer threshold $\tau_{\mathrm{m}} = 1$. For a merge cluster $\mathcal{G}$ containing a survivor and all primitives assigned to it, let o_i be primitive i's stored base-opacity logit and let $\alpha_i = \mathrm{sigmoid}(o_i)$ be its base opacity, clamped to $[10^{-6}, 1 - 10^{-6}]$, and let $\rho_i = \alpha_i / \sum_{j \in \mathcal{G}} \alpha_j$. Cluster attributes are updated as summarized in table S3, where $\mathbf{q}_i$ denotes the stored rotation quaternion.

Branch probe. The no-fine-tuning probe reduces the teacher with both branches to the same primitive count and selects the result with higher PSNR on a fixed subset of training-camera queries. It uses no held-out cameras and takes about 40 seconds per scene on an RTX 4090.

C.4 Recovery

The selected reduced model is fine-tuned for 16K iterations using Adam ($\mathrm{lr} = 10^{-3}$, $\epsilon = 10^{-15}$), with cosine decay to 5% of the initial rate and the loss

$$\mathcal{L}_{\mathrm{rec}} = 0.8\,\mathcal{L}_1 + 0.2(1 - \mathrm{SSIM}). \tag{S5}$$

Here, $\mathcal{L}_1$ is the mean absolute RGB reconstruction error. Training-view and interpolated-view queries alternate in a 1:1 ratio. A training view uses $\frac{1}{2}(\mathbf{I}^{\mathrm{gt}} + \mathbf{I}^{\mathrm{teacher}})$ as its target. For an interpolated view, we interpolate the poses of two training cameras and use the frozen teacher render as the target. Recovery changes neither the primitive count nor the branch decision.

C.5 Attribute Compression

Positions and temporal centers. Let the reduced model contain K primitives, and let $\mu_{i,k}$ denote coordinate $k \in \{1, 2, 3\}$ of primitive $i \in \{1, \ldots, K\}$. For query q, let $\mathbf{J}_q = \partial \mathbf{I}_q / \partial \boldsymbol{\Theta}$ be the Jacobian of the rendered image $\mathbf{I}_q$ with respect to all model parameters $\boldsymbol{\Theta}$, and define the render-space metric $\mathbf{F} = \mathbb{E}_q[\mathbf{J}_q^{\mathsf{T}} \mathbf{J}_q]$ and its position block $\mathbf{F}_{\mu\mu}$. For each axis, define its range and aggregate observability as

$$A_{\mu,k} = \max_i \mu_{i,k} - \min_i \mu_{i,k}, \qquad O_{\mu,k} = \sum_i [\mathbf{F}]_{\mu_{i,k}, \mu_{i,k}}. \tag{S6}$$

Table S3: Attribute aggregation for a merge cluster $\mathcal{G}$. Opacity is composed as an alpha union; all remaining averages use base-opacity weights ρ_i.

Attribute	Cluster update
Opacity	$\alpha_{\mathcal{G}} = 1 - \prod_{i\in\mathcal{G}}(1-\alpha_i)$, stored in logit form
Rotation	$\mathbf{q}_{\mathcal{G}} = \mathrm{normalize}(\sum_{i\in\mathcal{G}} \rho_i \mathbf{q}_i)$
Scale	Weighted mean of the stored log-scales
Others	Weighted arithmetic mean of position, time center, duration, velocity, marginal gate, and all SH coefficients

Algorithm S1 Unified delete–merge decimation.

Require: Teacher primitives $\mathcal{P} = \{\boldsymbol{\theta}_i\}_{i=1}^{N}$, query grid $\mathcal{Q}$, target count K, transfer threshold τ_{m}
Ensure: Reduced primitive set $\mathcal{P}_{\mathrm{red}}$

for each primitive i **do**
  Construct $\boldsymbol{\Phi}_i = [\mathbf{U}_{i,q}]_{q\in\mathcal{Q}}$
  $j^*(i) \leftarrow \arg\min_{j\neq i} \|\boldsymbol{\Phi}_i - \boldsymbol{\Phi}_j\|_2$
  Compute $\delta_i^{\mathrm{delete}}$ and $\delta_i^{\mathrm{merge}}$ as in the main paper
end for
$\hat{\delta}_i^a \leftarrow \delta_i^a / \operatorname{mean}_{\ell=1}^{N} \delta_\ell^a$, $a \in \{\mathrm{delete}, \mathrm{merge}\}$
$\mathrm{cost}_i \leftarrow \min(\hat{\delta}_i^{\mathrm{delete}}, \hat{\delta}_i^{\mathrm{merge}})$
$\mathcal{S} \leftarrow$ indices of the K largest cost_i; $\mathcal{T} \leftarrow \{1,\dots,N\} \setminus \mathcal{S}$
for each removed primitive $i \in \mathcal{T}$ **do**
  $j^\dagger(i) \leftarrow \arg\min_{j\in\mathcal{S}} \|\boldsymbol{\Phi}_i - \boldsymbol{\Phi}_j\|_2$
  $r_i \leftarrow \|\boldsymbol{\Phi}_i - \boldsymbol{\Phi}_{j^\dagger(i)}\|_2$
end for
$r_{\mathrm{med}} \leftarrow \mathrm{median}_{i\in\mathcal{T}} r_i$
for each $i \in \mathcal{T}$ **do**
  if $r_i < \tau_{\mathrm{m}} r_{\mathrm{med}}$ **then**
    Assign i to the merge cluster of survivor $j^\dagger(i)$
  else
    Delete i
  end if
end for
Aggregate each survivor cluster using table S3
return the updated survivors $\mathcal{P}_{\mathrm{red}}$

We uniformly quantize all three axes with a scene-wise common integer precision b_μ. The step on axis k is $\Delta_k = A_{\mu,k}/(2^{b_\mu}-1) \simeq A_{\mu,k}2^{-b_\mu}$. Under the standard high-rate model (Gersho & Gray, 1992), the coordinate error $\delta_{i,k} = \widehat{\mu}_{i,k} - \mu_{i,k}$, where $\widehat{\mu}_{i,k}$ is the decoded coordinate, is zero-mean with variance $\Delta_k^2/12$, and errors in different coordinates are uncorrelated. The expected position-induced render distortion is therefore

$$\mathbb{E}[\mathcal{D}_\mu] \simeq \mathbb{E}[\boldsymbol{\delta}_\mu^{\mathsf{T}} \mathbf{F}_{\mu\mu} \boldsymbol{\delta}_\mu] \simeq \frac{2^{-2b_\mu}}{12} \sum_{k=1}^{3} O_{\mu,k} A_{\mu,k}^2, \tag{S7}$$

where $\boldsymbol{\delta}_\mu$ stacks all position-coordinate errors and $\mathcal{D}_\mu$ uses the same query and image normalization as $\mathbf{F}$. Requiring $\mathbb{E}[\mathcal{D}_\mu] \le \varepsilon_\mu$, where ε_μ is the tolerated position-induced squared render distortion, gives

$$b_\mu^{(0)} = \left\lceil \frac{1}{2} \log_2 \frac{\sum_{k=1}^{3} O_{\mu,k} A_{\mu,k}^2}{12\varepsilon_\mu} \right\rceil. \tag{S8}$$

This is the closed-form precision prior used in the main paper. For codec calibration, let m_0 be the training-view RGB mean-squared error immediately before position quantization, after geometry/motion vector quantization (VQ), appearance fitting, joint refinement, and the FP16 codebook round-trip. During calibration, only the positions are quantized; temporal centers remain unchanged. Position observability is estimated on this same pre-position-quantization model at six time points, with one randomly sampled training camera and one image-space Rademacher probe per time point (six camera–time queries in total). For an allowed position-induced PSNR decrease η, the corresponding incremental MSE budget is $m_0(10^{\eta/10}-1)$. Under the local linear render model and

zero-mean quantization errors, the cross term between the existing reconstruction residual and the quantization-induced image change vanishes in expectation. We therefore express this budget under the normalization of $\mathcal{D}_\mu$ to obtain ε_μ and use $\eta = 0.05$ dB.

Morton ordering and subsequent entropy coding preserve the quantized coordinates exactly, so they do not alter eq. (S7); they determine the realized byte rate. The analytic estimate $b_\mu^{(0)}$ provides a position-precision prior. We evaluate candidate bit depths on training views and select the lowest tested precision b_μ that satisfies the position-only PSNR tolerance. We measure the quantization distortion of each candidate precision by rendering three interior training cameras at every sixth frame, giving 150 camera–time queries on N3DV and 30 on SelfCap. Table S4 compares the analytic prior $b_\mu^{(0)}$ with the codec-calibrated precision b_μ. Across both datasets, the prior never exceeds the calibrated precision and is within two bits in every scene.

Table S4: Analytic position-precision prior $b_\mu^{(0)}$ from eq. (S8) and codec-calibrated precision b_μ at 250K primitives. The $b_\mu^{(0)}$ column reports the modal integer over three seeds, while b_μ is identical across all three seeds of every scene. On N3DV, $b_\mu^{(0)}$ varies by one bit for *coffee martini* and *flame steak*, whose continuous estimates lie near integer boundaries; it is identical across seeds for all other scenes, and on all six SelfCap scenes.

N3DV scene	$b_\mu^{(0)}$	b_μ	SelfCap scene	$b_\mu^{(0)}$	b_μ
coffee martini	18	18	corgi1	18	18
cook spinach	15	16	corgi2	18	18
cut roasted beef	14	16	dance1	19	20
flame salmon	17	18	dance2	19	20
flame steak	14	16	bike1	16	16
sear steak	14	16	bike2	15	16

The selected quantized positions determine a Morton order (Morton, 1966), which is shared by all attribute rows and requires no stored permutation. Temporal centers use 8-bit uniform quantization and the same row order.

Geometry and motion. Scale, rotation, duration, and velocity use sub-vector quantization (Jégou et al., 2011). Their respective sub-vector dimensions are $1, 2, 1, 1$. SelfCap uses codebook sizes $512, 16384, 512, 512$, whereas N3DV uses $256, 8192, 512, 512$. K-means uses five initializations. Codebooks are stored in FP16; their index streams are Huffman-coded and included in the final container. Evaluation uses the codebooks after the same FP16 round-trip applied during serialization. For velocity, the learned codebooks adapt to the heavy-tailed distribution, avoiding a uniform step size determined by a few extreme values.

Appearance and opacity. We replace per-primitive SH coefficients and opacity by a 6-D latent code and three shared decoding heads for DC color, higher-order SH coefficients, and opacity. Each decoder uses frequency-encoded canonical positions, two hidden layers of width 64, and Leaky-ReLU activations. The 6-D latent is split into 2-D sub-vectors and quantized with 1024-entry codebooks. Both appearance-compression stages optimize $0.8\mathcal{L}_1 + 0.2(1 - \mathrm{SSIM})$ using Adam with $\epsilon = 10^{-8}$, cosine decay to 5% of the initial learning rate, and gradient clipping at a global ℓ_2 norm of 1.0. Appearance fitting uses an initial learning rate of 3×10^{-3} for 12K iterations. After VQ assignments are fixed, all codebooks and continuous decoder parameters are jointly refined at an initial learning rate of 10^{-3} for 6K iterations. Optimization uses frozen-teacher supervision. For both N3DV and SelfCap, checkpoint selection uses PSNR on three interior training cameras.

Serialization. After quantization, every attribute stream is placed in the common Morton row order. The complete representation—including position and time streams, FP16 codebooks, Huffman data, appearance networks, latents, and metadata—is serialized into a single LZMA-compressed container. Reported sizes include all of these components.

The complete 250K compression pipeline, including reduction, recovery, and attribute compression, takes roughly 20 minutes per scene on one RTX 4090.

C.6 Rendering Speed and Appearance Baking

Our appearance decoders are evaluated once when the compressed model is loaded, and their outputs are baked into explicit Gaussian SH coefficients and opacities. Timed inference therefore uses the standard Gaussian renderer without per-frame MLP decoding. By contrast, the released OMG4 inference path used in the N3DV comparison decodes neural appearance during rendering; this difference, together with the reduced primitive count, contributes to the higher measured FPS of ORDO-GS. We measure batch-one rendering on the held-out camera over the complete sequence after ten warm-up frames, with CUDA synchronization around each timed frame. Timing covers rendering only.

C.7 Controlled OMG4 Comparison

Because no official OMG4 SelfCap checkpoint is available, we construct a strong, controlled 250K adaptation denoted by *OMG4+interp.* Starting from the same FTGS++ teacher as ORDO-GS, it applies the SD-Score of OMG4 (Lee et al., 2025b) in a budget-controlled progressive reduction and uses an attribute codec matched to the released FTGS checkpoints. It then uses the same 16K-step interpolated-view recovery protocol and random seeds as ORDO-GS; this additional view supervision improves the baseline over training-view recovery alone. We also evaluated adaptations including the subsequent Gaussian Pruning and Merging stages at their closest attainable primitive counts. Across the SelfCap and FTGS settings, the reported SD-Score configuration gives the highest PSNR among the tested stage compositions at comparable budgets. The comparison therefore gives the OMG4-based baseline its strongest tested operating point while matching the teacher, recovery supervision, and primitive budget. The official OMG4-released N3DV checkpoints are evaluated with their shipped representation and our common held-out-view metric protocol. We use its FTGS–L (284K) and FTGS–S (90K) checkpoints for all six scenes, corresponding to the 5.87 MB / 32.62 dB and 2.01 MB / 32.22 dB anchors, respectively.

C.8 Cross-Representation Transfer to Ex4DGS

We operate directly on the native static and dynamic primitive blocks of Ex4DGS (Lee et al., 2024b). The branch probe selects prune-only for these forward-facing models: loss-gradient ranking is applied to both blocks at a common retention ratio, followed by recovery with the native Ex4DGS renderer. Positions and motion trajectories use uniform quantization with shared Morton ordering, while the remaining geometry and explicit SH appearance use sub-vector quantization before final serialization. For the controlled comparison in the main paper, both ORDO-GS and OMG4 use this explicit-appearance coding protocol.

D Rate–Distortion Curves

To evaluate the complete rate–distortion behavior, we vary the structural target K while keeping the recovery, attribute-compression, and decoded-evaluation protocols fixed. Every operating point runs the complete pipeline, is measured from the serialized container, and is evaluated on the held-out camera. The resulting curves are shown in fig. S1.

On N3DV, the curve saturates beyond 250K primitives. Before attribute compression, the 250K and 400K models have virtually identical PSNR (32.847 and 32.848 dB); the small final variation is introduced during the fixed-capacity appearance-compression stage rather than structural reduction, and the final error bars overlap. At the low-rate end, the 50K model reaches 32.263 dB in 1.074 MB, exceeding the 90K OMG4 operating point while using substantially less storage. At the two headline-scale operating points, the three-seed reruns improve on OMG4 by 0.299 dB with 12.6% less storage at 90K, and by 0.084 dB with 25.7% less storage at 250K versus the 284K OMG4 checkpoint.

Position-codec rate–distortion. At 250K primitives, we sweep b_μ with all non-position codec settings fixed and serialize the complete container at each precision. As shown in fig. S2, the analytic prior lies near the high-fidelity transition. Calibration uses only training views; the locally nonmonotonic held-out response of *flame steak* therefore does not affect precision selection.

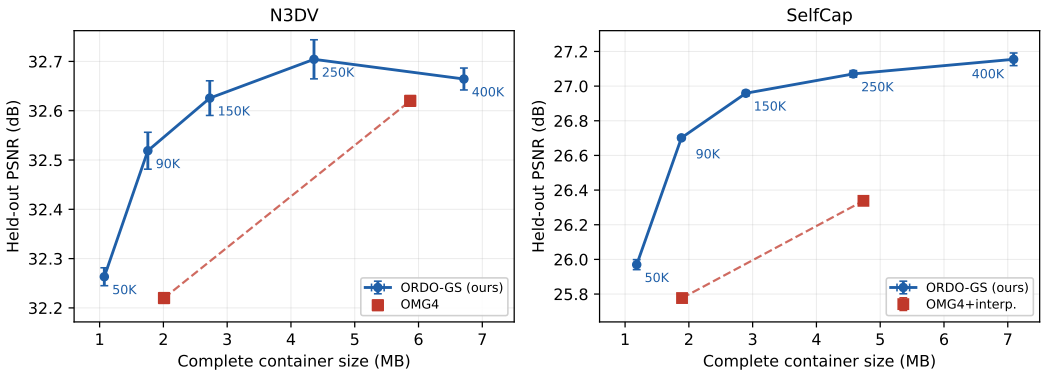


Figure S1: Structural rate–distortion curves on N3DV and SelfCap. For each target primitive count K, we run the complete reduction, recovery, and attribute-compression pipeline. ORDO-GS points are three-seed means, with error bars denoting one standard deviation in held-out PSNR. The N3DV baseline markers are the official 90K and 284K OMG4 checkpoints; the SelfCap markers are three-seed means from the matched 90K and 250K *OMG4+interp.* runs. Dashed segments connect the two measured baseline points only as a visual guide and do not represent evaluated intermediate configurations.

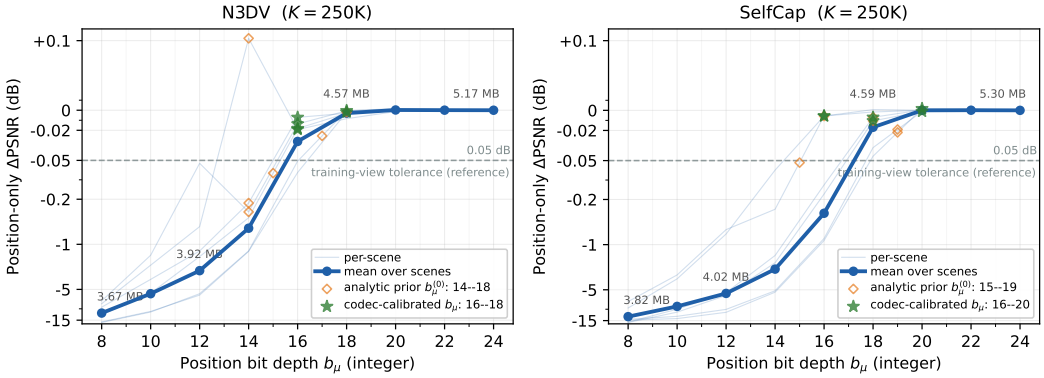


Figure S2: Position-codec rate–distortion behavior at 250K primitives. Thin curves show individual scenes and thick curves their mean. The ordinate is the held-out PSNR change relative to the 24-bit position reference, after removing the bit-independent temporal-quantization offset, and uses a symmetric-log scale to resolve the 0.05 dB operating region. Orange diamonds mark the analytic prior $b_\mu^{(0)}$ and green stars the codec-calibrated precision b_μ selected using training views. The dashed line shows the training-view calibration tolerance only as a visual reference, not as a held-out selection boundary; labeled sizes are mean complete-container sizes in decimal MB.

Position-distortion prediction. We test the local distortion model in eq. (S7) on all twelve scenes at 250K primitives and seed 0. Only positions are quantized; all other attributes remain fixed. The measured quantity is the squared render difference between the quantized and unquantized models, matching the prediction target. At $b_\mu \in \{12, 14, 16, 18, 20, 22\}$, the cross-scene Spearman correlations between predicted and measured distortion are 0.986, 0.986, 0.993, 0.993, 0.993, and 0.944, respectively. At each scene's selected precision, the measured-to-predicted distortion ratio ranges from 1.28 to 3.32, with a median of 1.62. At 16 bits, the full predictor $\sum_k O_{\mu,k} A^2_{\mu,k}$ achieves a Spearman correlation of 0.993 across the combined set of 12 scenes, compared with 0.769 for the range-only predictor $\sum_k A^2_{\mu,k}$. These results support the analytic model as a precision prior, with codec calibration accounting for deviations in the predicted distortion.

E Additional Reduction Analysis

Branch-probe results. At 250K primitives, we repeat the no-fine-tuning probe on a fixed subset of three interior training cameras. The probe is deterministic for this fixed query subset and retains decisive preferences for delete–merge on *dance1* and *dance2*, with pre-recovery margins of 2.910 and 2.564 dB.

Merge features. We compare screen-space trajectories, world-space Fisher features, and handcrafted world attributes at 250K primitives on *corgi1/2* and *dance1/2*. Results in table S5 use three-seed mean PSNR after recovery and attribute compression.

Table S5: Merge-feature comparison across four SelfCap scenes. The maximum per-scene PSNR gap to the best tested feature is reported; lower is better.

	Screen trajectories	World-space Fisher	Handcrafted attributes
Maximum gap (dB)	**0.28**	1.62	1.93

Temporal sampling density for delete–merge reduction. We vary only the number of temporal samples used to compute the residual-weighted deletion scores and screen-space trajectory features in the delete–merge branch, using nested grids of 6, 12, and 24 time points with the same training cameras. Prune-only scoring and probe evaluation queries remain fixed, as do the sampling protocols for observability estimation, recovery training, and position calibration. All variants use 250K primitives, seed 0, 16K recovery iterations, and 12K appearance-fitting plus 6K joint-refinement iterations. Table S6 reports the decoded delete–merge models after this complete schedule. The no-fine-tuning probe selects the same branch at all three densities for each scene, while denser sampling does not consistently improve final reconstruction quality. We retain six time points as the default.

Table S6: Temporal sampling density for delete–merge scoring and trajectory construction. Entries report final PSNR (dB) / complete container size (MB) for the delete–merge branch, irrespective of the probe's branch choice.

Scene	6 time points	12 time points	24 time points
dance1	27.106 / 4.690	27.166 / 4.698	27.114 / 4.703
corgi1	28.927 / 4.501	28.525 / 4.497	28.749 / 4.524
flame steak	34.021 / 4.431	34.160 / 4.381	34.085 / 4.512

Transfer threshold. In a seed-0 sensitivity test at 250K primitives, we sweep $\tau_{\mathrm{m}} \in \{0.5, 1, 2\}$ on *dance1*, *dance2*, and *corgi1*. The corresponding mean merge-assignment rates are 25.9%, 50.0%, and 77.5%, while mean PSNR across the three scenes spans only 0.051 dB. We therefore use the common setting $\tau_{\mathrm{m}} = 1$, which gives a balanced transfer rule within this stable range; the survivor set remains governed by the unified decimation cost.

Residual-weighted deletion. Removing the local residual from the deletion energy isolates the role of its error weighting. In a seed-0 test at 250K primitives across the six SelfCap scenes, residual

weighting improves the PSNR of the unrecovered reduced model in five scenes and by 0.119 dB on average. The common recovery stage subsequently narrows the difference, consistent with the residual term's role in providing a better initial removal ordering.

Contribution transfer with fixed survivors. At 250K primitives and seed 0, we fix the survivors selected by delete–merge and compare direct deletion of the remaining primitives with contribution transfer. After the same 16K-step recovery, transfer changes held-out PSNR by $+0.024$, -0.028, $+0.193$, and $+0.011$ dB on *dance1*, *dance2*, *corgi1*, and *flame steak*, respectively. The complete branch comparison additionally captures differences in scoring and survivor selection.

F Per-Scene Results

To complement the dataset averages in the main paper, we report per-scene PSNR and complete serialized size over three seeds using the same held-out-view protocol as the main comparison. All operating points use the common implementation detailed in section C to ensure consistent comparisons across budgets. Table S7 reports the N3DV results.

Table S7: Per-scene N3DV results independently regenerated under the unified three-seed implementation used for the supplementary sweep. Compressed entries are mean ± standard deviation; in the mean row, the standard deviation is computed across the seed-wise six-scene averages. The larger 90K size variation on *sear steak* arises because different seeds select adjacent position-precision levels under the same 0.05 dB codec-calibration criterion. Sizes are complete containers in decimal MB.

	Teacher (500K)	ORDO-GS–L (250K)		ORDO-GS–S (90K)	
Scene	PSNR	PSNR	Size	PSNR	Size
coffee martini	29.594	29.460 ± 0.117	4.185 ± 0.008	29.382 ± 0.084	1.668 ± 0.001
cook spinach	33.727	33.636 ± 0.032	4.360 ± 0.002	33.380 ± 0.023	1.776 ± 0.002
cut roasted beef	34.196	34.252 ± 0.048	4.436 ± 0.006	33.826 ± 0.076	1.773 ± 0.001
flame salmon	30.663	30.465 ± 0.029	4.237 ± 0.005	30.392 ± 0.034	1.688 ± 0.003
flame steak	34.368	34.235 ± 0.072	4.499 ± 0.002	34.118 ± 0.087	1.786 ± 0.003
sear steak	34.073	34.177 ± 0.116	4.449 ± 0.010	34.014 ± 0.114	1.846 ± 0.067
Mean	32.770	32.704 ± 0.040	4.361 ± 0.004	32.519 ± 0.037	1.756 ± 0.012

Table S8: Per-scene SelfCap results under the unified three-seed protocol at matched primitive counts. PSNR entries and ORDO-GS sizes are mean ± standard deviation across three seeds; *OMG4+interp.* sizes report the corresponding three-seed means. In the mean row, standard deviations are computed across the seed-wise six-scene averages. Sizes are in decimal MB.

	Teacher	ORDO-GS–L (250K)		OMG4+interp. (250K)		ORDO-GS–S (90K)		OMG4+interp. (90K)	
Scene	PSNR	PSNR	Size	PSNR	Size	PSNR	Size	PSNR	Size
corgi1	29.232	28.731 ± 0.032	4.495 ± 0.004	28.874 ± 0.035	4.893	28.066 ± 0.011	1.873 ± 0.002	28.511 ± 0.077	1.950
corgi2	27.242	27.823 ± 0.029	4.532 ± 0.003	27.693 ± 0.057	4.807	27.305 ± 0.077	1.826 ± 0.002	27.598 ± 0.097	1.908
dance1	27.194	27.147 ± 0.037	4.697 ± 0.003	24.605 ± 0.055	4.726	26.748 ± 0.005	1.932 ± 0.006	23.538 ± 0.037	1.884
dance2	26.227	26.098 ± 0.028	4.715 ± 0.002	24.118 ± 0.015	4.723	25.751 ± 0.024	1.934 ± 0.003	22.698 ± 0.021	1.878
bike1	27.347	27.267 ± 0.011	4.453 ± 0.007	27.279 ± 0.036	4.608	27.041 ± 0.040	1.842 ± 0.002	27.153 ± 0.063	1.869
bike2	25.597	25.356 ± 0.055	4.579 ± 0.006	25.459 ± 0.004	4.667	25.297 ± 0.015	1.894 ± 0.014	25.161 ± 0.035	1.867
Mean	27.140	27.070 ± 0.019	4.579 ± 0.002	26.338 ± 0.025	4.737	26.701 ± 0.010	1.884 ± 0.002	25.777 ± 0.015	1.893

On N3DV, at 250K and 90K primitives, ORDO-GS remains within 0.066 and 0.251 dB of the 500K teacher while reducing the complete serialized representation to 4.361 and 1.756 MB, respectively.

Table S8 reports the corresponding SelfCap results and matched-count comparisons with *OMG4+interp.*

At matched 250K and 90K budgets, ORDO-GS improves over *OMG4+interp.* by 0.732 and 0.925 dB while using 3.4% and 0.5% smaller complete containers, respectively. The largest gains occur on *dance1* and *dance2*, consistent with the advantage of the delete–merge branch in scenes with high persistent co-location.

G TEMPORAL QUALITY

We evaluate all six SelfCap scenes at 250K primitives, using all 60 consecutive frames from the same held-out camera for both methods in each scene. Evaluations use restored model states, including the dynamic residuals in *OMG4+interp.* For each scene, we compute the mean and 10th percentile of framewise PSNR against ground truth, and the minimum mean PSNR over all contiguous six-frame windows. Each statistic is then averaged equally across scenes. To measure temporal changes in compression error, we define $\mathbf{e}_t = \mathbf{I}_t^{\text{compressed}} - \mathbf{I}_t^{\text{teacher}}$ and average $|\mathbf{e}_t - \mathbf{e}_{t-1}|$ over pixels, color channels, and all 59 adjacent frame pairs, with RGB values in $[0, 1]$. This measures temporal error variation without motion compensation. Table S9 shows higher mean, lower-tail, and worst-window PSNR for ORDO-GS, alongside slightly higher temporal error variation.

Table S9: Temporal quality over complete SelfCap sequences. Single-run results, averaged over all six scenes.

Metric	ORDO-GS	OMG4+interp.
Mean PSNR (dB) ↑	27.077	26.337
10th-percentile PSNR (dB) ↑	26.670	26.046
Worst six-frame mean PSNR (dB) ↑	26.466	25.918
Temporal error variation ↓	0.002837	0.002695

H MULTIMEDIA APPENDIX

The accompanying multimedia archive contains synchronized held-out-view comparisons for four representative scenes. Each video uses a 2×2 layout with method labels embedded in the frames. The N3DV *coffee martini* video compares ground truth (GT), the FTGS teacher, the official OMG4 checkpoint, and ORDO-GS at the default 30 fps. The SelfCap videos cover *corgi1*, *dance1*, and *dance2*, comparing GT, the FTGS++ teacher, *OMG4+interp.*, and ORDO-GS. They are played at 20 fps and shown twice consecutively to facilitate inspection of motion. In each filename, `cN` denotes held-out camera index N.