GEOMETRY SHOULD NOT FOLLOW THE CAMERA: MOTION–PROJECTION FACTORIZATION FOR 4D GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Dynamic novel-view synthesis reconstructs time-varying scenes from videos and renders novel views at unobserved viewpoints. Existing Gaussian optimization entangles geometry with training-view information, rendering them non-identifiable from 2D projection errors alone. In this paper, we investigate a fundamental question: can image-space errors be attributed to 3D Gaussian geometry in an identifiable and viewpoint-invariant manner? We first prove that position and shape respond differently to viewpoint changes, inducing view-dependent projection bias that can be compensated by normal-direction shrinkage. Subsequently, we propose MoProGS (Motion-Projection Factorized Gaussian Splatting), which factorizes dynamic deformation from view-dependent projection. Specifically, we separately model the geometric responses of Gaussian position and covariance, and derive a geometry-view coupling factor to boundedly adjust the projected support before rasterization. This balances the relative contributions of geometric parameters to forward projection and backward gradients, yielding viewpoint-consistent geometric error attribution. Experiments on D-NeRF, HyperNeRF, and Neu3D show that MoProGS consistently improves novel-view synthesis quality. Controlled studies further verify the distinct projection sensitivities of position and shape parameters, and clarify the roles of camera conditioning and individual components.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) represents a scene with explicit, rasterizable primitives and renders photorealistic novel views in real time. Deformation-based 4D Gaussian Splatting (4DGS) extends 3DGS to dynamic scenes by deforming a canonical set of Gaussians over time (Yang et al., 2024b; Wu et al., 2024; Xu et al., 2024). This extension enables real-time novel-view rendering of time-varying scenes, supporting applications such as free-viewpoint video, immersive telepresence, and AR/VR content creation. The reconstructed dynamic scenes can also serve as simulation environments for robotics and embodied agents. Among capture settings, monocular video is the most accessible, requiring only a single handheld camera rather than a synchronized multi-camera rig (Park et al., 2021b; Stearns et al., 2024; Wang et al., 2025). However, a monocular camera observes each time step from only one viewpoint, whereas these applications need to render the same time step from many viewpoints. Therefore, **the geometry reconstructed at each time step must remain consistent under unobserved viewpoints, rather than merely fitting the single observed view.**

The main solutons have focused on deformation modeling. Coordinate networks, factorized spatiotemporal features, and hash encodings improve representation capacity (Yang et al., 2024b; Wu et al., 2024; Cao & Johnson, 2023; Fridovich-Keil et al., 2023; Xu et al., 2024). Sparse controls, temporal aggregation, and multi-step deformation impose additional structure on motion (Huang et al., 2024b; Lu et al., 2024; Hu et al., 2026). Per-Gaussian embeddings and scale modulation adapt the response to individual primitives (Bae et al., 2024; Hu et al., 2026), while frequency-aware representations address the tension between fine-scale detail and motion (Miao et al., 2026). These advances improve deformation expressivity and learning reliability, but the deformation field is still optimized only through image-space reconstruction losses, which cannot distinguish different geometric explanations

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.

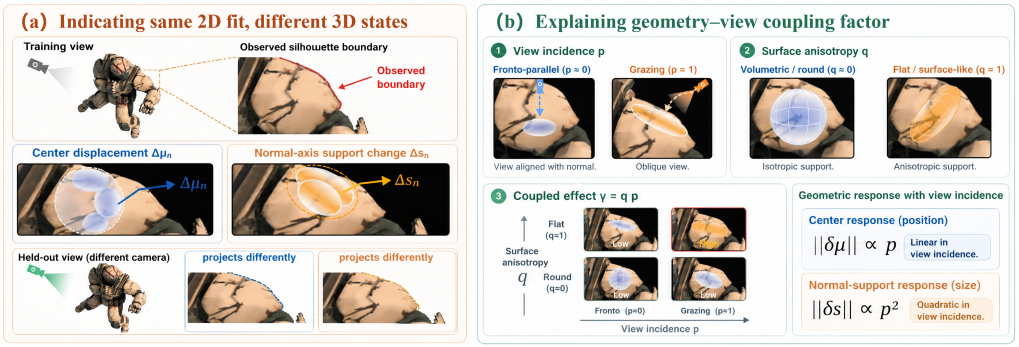

Figure 1: **Non-unique projection and view-dependent sensitivity.** (a) Translating a surface-like Gaussian along its thinnest direction ($\Delta\mu_n$) or changing its thickness (Δs_n) fits the training view equally well but projects differently in a held-out view. (b) Incidence p (①) and anisotropy q (②) form the coupling $\gamma = qp$ (③); center displacement affects the projected image at first order in p, whereas thickness change affects it at second order.

that render the same observations. Increasing deformation capacity alone therefore does not remove this ambiguity, since distinct 3D configurations can attain the same image-space loss.

In above solutions, the 2D projection of a Gaussian is determined by its center, scale, rotation, and temporal deformation. For locally surface-like Gaussians, shifting the center or changing the thickness along the thinnest axis can yield nearly identical 2D projections while corresponding to fundamentally different 3D states (Fig. 1a). Both configurations fit the training view, yet they describe different 3D geometry and look different from other viewpoints, creating an inherent ambiguity between 2D observations and 3D geometry. The ambiguity is further amplified by the viewpoint-dependent bias of different Gaussian parameters on the 2D projection. Fig. 1b illustrates this phenomenon. Specifically, translation and thickness variation along the thinnest axis exhibit first- and second-order viewpoint sensitivities in the projected center and covariance, respectively. Such sensitivities introduce an implicit dependence of the optimized Gaussian parameters on the training cameras. This camera dependence is further compounded in monocular dynamic scenes, where rendered observations are jointly affected by camera pose, time, and scene state, leading to inconsistencies under novel views.

We propose MoProGS (Motion–Projection Factorized Gaussian Splatting). First, we decouple the geometric responses of position and covariance at the output of the deformation field, so that their relative magnitudes are no longer constrained by a view-independent constant. This introduces an adjustable degree of freedom to account for the inherent ambiguity between 2D projections and 3D geometry. Second, we design a geometry–view coupling factor, $\gamma = qp$, based on each deformed Gaussian and the current camera, which modulates normal-direction thickness. As illustrated in Fig. 1, the view factor p measures the projected visibility of the Gaussian's thinnest axis. It approaches 0 when the axis aligns with the viewing direction and approaches 1 under near-tangential viewing (Fig. 1b, ①). The geometry factor q measures the Gaussian's anisotropy and indicates whether its minimum-scale axis provides a reliable surface-normal estimate. The factor q approaches 0 when the three scales are similar (Fig. 1b, ②). Consequently, γ becomes large only when the Gaussian is sufficiently flat and its thinnest direction is well exposed under the current view (Fig. 1b, ③). Based on the designed geometry–view coupling factor, we apply a bounded contraction to the Gaussian thickness along this direction before rasterization. The factor γ is computed from the deformed Gaussian and the current camera, and is detached from the computational graph. It thus acts as a parameter-free projection factor, introducing view information at rasterization, downstream of a camera-agnostic deformation field.

Our main contributions are threefold:

- We theoretically show that, in prevailing 4DGS models, the relative contributions of position and shape parameters are camera-dependent rather than determined by the scene state, revealing an additional limitation of monocular 4D Gaussian Splatting.
- We propose MoProGS, a motion–projection factorization for 4D Gaussian Splatting that decouples the geometric responses of position and shape. By modulating their relative contributions with a

geometry–view coupling factor before rasterization, MoProGS reduces the view-dependent bias in geometric attribution.

- We validate MoProGS on D-NeRF, HyperNeRF, and synchronized Neu3D through controlled projection studies and ablations, with consistent gains across monocular and multi-view settings.

2 Related Work

Dynamic scene representations. Dynamic radiance fields (Mildenhall et al., 2020) deform a canonical field (Pumarola et al., 2021; Park et al., 2021a;b; Tretschk et al., 2021), condition on time (Xian et al., 2021; Li et al., 2022), predict scene flow (Li et al., 2021; Gao et al., 2021), or aggregate nearby views (Li et al., 2023), and voxel, plane, and hybrid factorizations accelerate them (Fang et al., 2022; Cao & Johnson, 2023; Fridovich-Keil et al., 2023; Shao et al., 2023; Wang et al., 2023; Song et al., 2023; Attal et al., 2023). Dynamic extensions of 3DGS (Kerbl et al., 2023) learn time-varying 4D primitives (Yang et al., 2024a; Li et al., 2024; Duan et al., 2024), track per-frame Gaussians under multi-view capture (Luiten et al., 2024; Sun et al., 2024), or deform a canonical Gaussian set with coordinate MLPs (Yang et al., 2024b), polynomial and Fourier trajectories (Lin et al., 2024), factorized space–time features (Wu et al., 2024; Cao & Johnson, 2023; Fridovich-Keil et al., 2023), or hash encodings (Xu et al., 2024). Subsequent methods introduce sparse controls (Huang et al., 2024b), temporal aggregation (Lu et al., 2024), static–dynamic separation (Wu et al., 2025), multi-step updates and scale modulation (Hu et al., 2026), per-Gaussian embeddings (Bae et al., 2024), and frequency-aware designs (Miao et al., 2026) to improve motion modeling and temporal coherence. Most of these methods decode position and covariance updates from a shared temporal feature, and some use separate heads for the two (Wu et al., 2024). MoProGS instead operates at the output of the deformation field. It decouples the two branch responses and uses the deformed primitive and current camera to calibrate their relative image response.

Monocular dynamic reconstruction. Monocular dynamic reconstruction is underconstrained because each scene state is observed from a single view, coupling geometry, motion, appearance, and visibility (Park et al., 2021a;b), and reported quality depends strongly on how much the camera moves between frames (Gao et al., 2022). Recent methods add depth, optical-flow, and tracking priors (Li et al., 2021; Gao et al., 2021; Liu et al., 2023; Wang et al., 2025; Lei et al., 2025) or reduce per-primitive freedom with isotropic Gaussians (Stearns et al., 2024). MoProGS keeps anisotropic Gaussians. We further identify an effect of acquisition itself. When each time is paired with one camera, time becomes correlated with the training view. This correlation is a shortcut in the sense of Geirhos et al. (2020) and allows a camera-blind deformation field to absorb view-specific fitting into temporal geometry. We isolate this coupling from projection dependence using synchronized capture.

Geometry-aware splatting and view-dependent appearance. SuGaR (Guédon & Lepetit, 2024) encourages surface alignment, 2DGS (Huang et al., 2024a) and Gaussian surfels (Dai et al., 2024) use oriented planar primitives, Gaussian Opacity Fields (Yu et al., 2024b) extract surfaces from Gaussian opacity along rays, DRK (Huang et al., 2025) generalizes the kernel shape, and Mip-Splatting (Yu et al., 2024a) addresses sampling artifacts. Per-image embeddings (Martin-Brualla et al., 2021) and reflection-aware color (Verbin et al., 2022) instead let appearance depend on the observer. MoProGS keeps volumetric Gaussians and uses the thin axis as a local normal cue. Current incidence modulates projected normal support and calibrates how image-space losses update position and covariance.

3 Method

3.1 Dynamic Gaussian deformation

A 3D Gaussian primitive i has center $\boldsymbol{\mu}_i$, rotation $\mathbf{R}_i$, scales $\mathbf{s}_i$, opacity, and spherical-harmonic appearance, with covariance $\boldsymbol{\Sigma}_i = \mathbf{R}_i \operatorname{diag}(\mathbf{s}_i^2)\mathbf{R}_i^\top$. For a camera with local projection Jacobian $\mathbf{J}_i$, its screen-space covariance is $\boldsymbol{\Sigma}_i^{2D} = \mathbf{J}_i \boldsymbol{\Sigma}_i \mathbf{J}_i^\top$, the local affine approximation of EWA splatting (Zwicker et al., 2001). A deformation model extracts a spatiotemporal feature $\mathbf{h}_i^t = \Phi_\psi(\boldsymbol{\mu}_i, t)$ and predicts center and covariance-parameter updates,

$$\Delta\boldsymbol{\mu}_i^t = F_\mu(\mathbf{h}_i^t), \qquad (\Delta\mathbf{r}_i^t, \Delta\mathbf{s}_i^t) = F_\Sigma(\mathbf{h}_i^t). \tag{1}$$

MoProGS retains this camera-blind temporal encoder. We next analyze how the outputs of the two geometry branches affect the rendered image through projection.

3.2 VIEW-DEPENDENT PROJECTION SENSITIVITY

The two geometry branches in Eq. (1) are supervised only through the rendered image, so the update each branch receives is governed by how its output changes the projection. We analyze this projection step for a locally surface-like Gaussian and show that the relative image-space response of position and normal thickness depends on the observing camera, even when the scene geometry is fixed. Specifically, for geometry branch $b \in \{\mu, \Sigma\}$ with output parameters θ_b and response parameters ϕ_b, the image-space loss reaches the response parameters through

$$\nabla_{\phi_b}\mathcal{L} = \left(\frac{\partial \theta_b}{\partial \phi_b}\right)^\top \left(\frac{\partial \widehat{I}}{\partial \theta_b}\right)^\top \nabla_{\widehat{I}}\mathcal{L}. \tag{2}$$

The first term maps response parameters to geometric changes. The second maps those changes to the image and depends on the current camera. We analyze this projection term for a locally surface-like Gaussian. Let $\mathbf{v}_i$ be the unit viewing direction from Gaussian i to the camera and $\mathbf{P}_i = \mathbf{I} - \mathbf{v}_i\mathbf{v}_i^\top$. The smallest-scale principal axis $\mathbf{n}_i$ approximates the local surface normal. We define

$$a_i = |\mathbf{n}_i^\top \mathbf{v}_i|, \qquad p_i = \|\mathbf{P}_i\mathbf{n}_i\| = \sqrt{1 - a_i^2}, \tag{3}$$

where p_i measures how much of the thin axis remains after projection. It is zero for a fronto-parallel surface and one at grazing incidence.

A displacement $\delta\mathbf{n}_i$ along the thin axis changes the projected center as $\|\mathbf{J}_i\delta\mathbf{n}_i\| = \Theta(|\delta|p_i)$. The same axis contributes $s_{i,n}^2\mathbf{J}_i\mathbf{n}_i\mathbf{n}_i^\top\mathbf{J}_i^\top$ to the projected covariance, whose trace scales as $\Theta(s_{i,n}^2 p_i^2)$. Center displacement is therefore first order in incidence, while normal support is second order. For normal displacement d and scale s_n, define $E_\mu = \|\partial\boldsymbol{\mu}^{2D}/\partial d\|_2$ and $E_s = \|\partial\boldsymbol{\Sigma}^{2D}/\partial s_n\|_F$ (singular-value bounds in Supplement Sec. A). Holding the remaining local projection factors fixed gives

$$E_\mu/E_s \propto 1/p_i. \tag{4}$$

The relative response of position and normal support therefore changes with the observer even when the scene geometry is fixed. Secs. 4.4 and 4.5 verify that this incidence dependence persists through full rasterization and in trained models. The dependence is intrinsic to projection and arises under any capture setup. Monocular capture further removes the cross-view observations that could resolve it, allowing camera-correlated attribution to be absorbed into temporal geometry.

3.3 MOPROGS: MOTION–PROJECTION FACTORIZATION

Based on this analysis, MoProGS separates camera-blind motion modeling from view-conditioned projection calibration (Fig. 2). We refer to the deformed Gaussian as the *scene state*, which is determined by the canonical Gaussian and t. The covariance passed to the rasterizer is the *projected support*, which may additionally depend on the observing camera.

Camera-blind branch responses. Existing deformation models decode the position and covariance updates from a shared temporal feature, which couples the two updates even though their relative image-space response varies with the observing camera (Sec. 3.2). To give the two branches independent responses, we insert identity-initialized adapters before the two geometry heads, $\mathbf{h}_{\mu,i}^t = \mathcal{A}_\mu(\mathbf{h}_i^t)$ and $\mathbf{h}_{\Sigma,i}^t = \mathcal{A}_\Sigma(\mathbf{h}_i^t)$, and predict $\Delta\boldsymbol{\mu}_i^t = F_\mu(\mathbf{h}_{\mu,i}^t)$ and $(\Delta\mathbf{r}_i^t, \Delta\mathbf{s}_i^t) = F_\Sigma(\mathbf{h}_{\Sigma,i}^t)$. Both adapters share the upstream temporal encoder, and each receives gradients only through its own geometric output. The two branches obtain independent responses.

Geometry–view coupling. After deformation, let $j^* = \arg\min_j s_{i,j}^t$ and $\mathbf{n}_i^t = \mathbf{R}_i^t\mathbf{e}_{j^*}$. We compute the anisotropy factor q, the incidence factor p, and their product, the *geometry–view coupling*,

$$q_i^t = \mathrm{clip}\left(1 - \frac{s_{i,j^*}^t}{s_{i,\mathrm{mid}}^t + \varepsilon}, 0, 1\right), \qquad p_i^t = \sqrt{1 - |\mathbf{n}_i^{t\top}\mathbf{v}_i^t|^2}, \qquad \gamma_i^t = q_i^t\, p_i^t \in [0,1]. \tag{5}$$

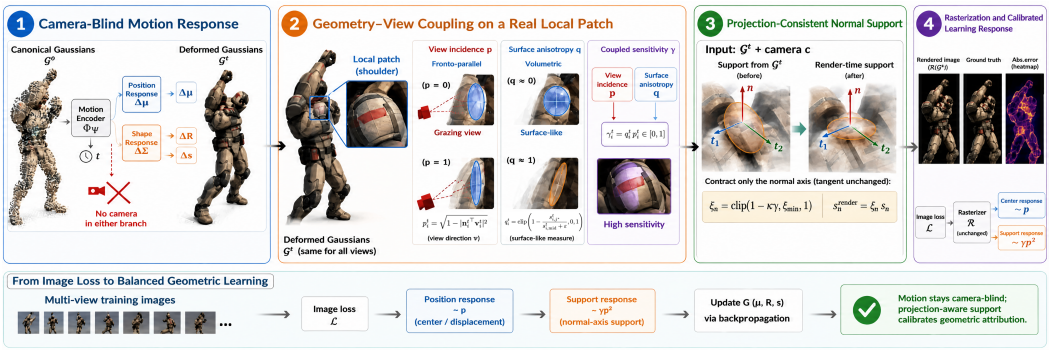

Figure 2: **Overview of MoProGS.** MoProGS keeps motion modeling camera-blind and introduces the current camera only at projection. 1 **Camera-Blind Motion Response:** independent position and shape response adapters act on a shared, camera-blind temporal feature. 2 **Geometry–View Coupling on a Real Local Patch:** the coupling $\gamma = qp$ is computed from the deformed primitive and current camera and is detached during backpropagation. 3 **Projection-Consistent Normal Support:** only the normal-axis support is contracted before rasterization; tangential scales are unchanged. 4 **Rasterization and Calibrated Learning Response:** the unchanged rasterizer renders the calibrated Gaussians, and the image loss updates both geometry branches through the calibrated center and support responses.

The factor q_i^t measures whether the Gaussian is anisotropic enough for its thin axis to be a reliable normal, and p_i^t how strongly this axis is exposed to the current camera. Their product activates calibration only for surface-like Gaussians whose normal support is visible, and both are computed from the deformed state used for rendering.

Projection-conditioned support calibration. The thin-axis scale $s_{i,n}$ defines the Gaussian's normal thickness. Before rasterization, we apply the per-axis factor

$$\xi_{i,n}^t = \mathrm{clip}\big(1 - \kappa\,\widetilde{\gamma}_i^t, \xi_{\min}, 1\big)\,, \qquad \xi_{i,t_1}^t = \xi_{i,t_2}^t = 1, \tag{6}$$

where κ is a learned global strength and $\widetilde{\gamma}$ denotes the detached coupling. We pass $\widetilde{\mathbf{\Sigma}}_i^t = \mathbf{R}_i^t\,\mathrm{diag}[(\boldsymbol{\xi}_i^t \odot \mathbf{s}_i^t)^2]\mathbf{R}_i^{t\top}$ to splatting. Only the normal scale is modified. The center and tangential scales remain unchanged. To measure the effect on relative response, hold the routing variables, thin-axis direction, and local $\mathbf{J}$ fixed. With $\lambda_n = s_n^2$ and $C_{\tan}$ the tangential projected covariance,

$$\widetilde{C} = C_{\tan} + \xi_n^2\lambda_n(\mathbf{Jn})(\mathbf{Jn})^\top, \qquad \frac{\partial\widetilde{C}}{\partial\lambda_n} = \xi_n^2(\mathbf{Jn})(\mathbf{Jn})^\top. \tag{7}$$

The calibrated scale sensitivity is $\widetilde{E}_s = 2s_n\xi_n^2\|\mathbf{Jn}\|_2^2$, while the projected-center sensitivity remains $E_\mu = \|\mathbf{Jn}\|_2$. For $p > 0$,

$$\widetilde{E}_s/E_\mu = 2s_n\xi_n^2\|\mathbf{Jn}\|_2 \;\propto\; p\,\xi_n^2. \tag{8}$$

Without calibration, the support-to-position response ratio grows with p. Within the active range, ξ_n^2 decreases with p and counteracts this trend. The operator changes the relative gradient response of position and normal support, and Sec. 4.4 measures this response after full rasterization.

Detached routing. We evaluate the coupling as $\widetilde{\gamma} = \mathrm{sg}(\gamma)$, stopping gradients through p and q so that optimization cannot reduce the loss by changing the routing cue itself. Rendering gradients still update position, rotation, scale, and κ, and the coupling is recomputed at every forward pass.

4 Experiments

4.1 Setup

Datasets and metrics. We evaluate on the seven synthetic monocular sequences of D-NeRF (Pumarola et al., 2021), four real handheld *vrig* sequences of HyperNeRF (Park et al., 2021b),

Table 1: **Novel-view synthesis on D-NeRF** (PSNR / SSIM / LPIPS. Best , second best .)

Method	Bouncing Balls			Hell Warrior			Hook			Jumping Jacks		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Tensor-4D (Shao et al., 2023)	24.47	.962	.044	31.26	.925	.074	28.63	.943	.064	24.20	.925	.067
HexPlane (Cao & Johnson, 2023)	40.36	.992	.031	24.30	.944	.073	28.26	.955	.052	31.74	.974	.036
TiNeuVox-B (Fang et al., 2022)	40.28	.991	.039	28.30	.966	.064	31.86	.972	.044	34.77	.987	.033
4D-GS (Wu et al., 2024)	40.77	.994	.016	28.61	.973	.036	32.90	.977	.027	35.37	.985	.020
DeformGS (Yang et al., 2024b)	41.67	.995	.008	41.09	.986	.027	37.42	.986	.014	37.06	.989	.016
SC-GS (Huang et al., 2024b)	41.85	**.996**	.022	42.10	.989	.028	38.81	.991	.014	39.80	.992	.013
Grid4D (Xu et al., 2024)	42.36	.995	.007	42.85	.990	.016	38.89	.990	.009	39.36	.993	.008
FAGS (Miao et al., 2026)	43.32	.995	.007	43.13	**.991**	.014	40.18	.992	.008	40.52	**.994**	.007
MoProGS (ours)	**43.60**	**.996**	**.004**	**43.27**	**.991**	**.012**	**40.48**	**.993**	**.005**	**41.38**	**.994**	**.006**

Method	Mutant			Standup			Trex			Average		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Tensor-4D (Shao et al., 2023)	29.11	.945	.060	30.56	.958	.036	31.25	.964	.048	28.50	.946	.056
HexPlane (Cao & Johnson, 2023)	33.66	.982	.028	34.12	.983	.019	31.01	.976	.028	31.92	.972	.038
TiNeuVox-B (Fang et al., 2022)	33.99	.977	.030	36.10	.986	.020	32.87	.978	.031	34.02	.979	.037
4D-GS (Wu et al., 2024)	37.71	.988	.016	37.88	.990	.014	34.14	.985	.022	35.34	.985	.022
DeformGS (Yang et al., 2024b)	42.13	.994	.007	43.70	.994	.009	37.78	.993	.010	40.12	.991	.013
SC-GS (Huang et al., 2024b)	42.55	.995	.011	46.16	**.997**	.008	39.53	.994	.008	41.54	.994	.014
Grid4D (Xu et al., 2024)	43.94	.996	.004	46.51	**.997**	.004	40.01	.994	.008	41.99	.993	.008
FAGS (Miao et al., 2026)	44.09	**.997**	.004	47.30	**.997**	.003	40.69	.995	.008	42.75	**.995**	.007
MoProGS (ours)	**44.16**	**.997**	**.002**	**47.57**	**.997**	**.002**	**41.46**	**.996**	**.004**	**43.13**	**.995**	**.005**

Table 2: **Novel-view synthesis on Neu3D** (synchronized multi-view, held-out cam00). Format as in Tab. 1.

Method	Coffee Martini			Cook Spinach			Cut Roasted Beef			Flame Steak			Sear Steak			Average		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
DeformGS (Yang et al., 2024b)	26.71	.890	–	31.35	.943	–	31.89	.943	–	30.96	.953	–	31.59	.951	–	30.50	.936	–
4D-GS (Wu et al., 2024)	27.00	.885	.189	31.80	.942	.159	32.06	.939	.167	30.23	.945	.147	32.18	.958	.144	30.65	.934	.161
Grid4D (Xu et al., 2024)	27.90	.893	.179	32.38	.947	.144	33.20	.947	.157	32.07	.942	.136	32.61	.955	.132	31.63	.937	.149
FAGS (Miao et al., 2026)	28.37	.899	.179	32.79	**.967**	.145	33.20	.950	.148	**32.95**	.947	.132	33.58	**.968**	.125	32.18	.946	.146
MoProGS (ours)	**28.41**	**.910**	**.115**	**32.90**	.952	**.088**	**33.49**	**.952**	**.088**	32.71	**.958**	**.073**	**33.85**	.962	**.084**	**32.27**	**.947**	**.089**

Table 3: **HyperNeRF** (four *vrig* sequences).

Method	PSNR↑	SSIM↑
TiNeuVox-B	24.21	–
DeformGS	23.27	.646
4D-GS	24.64	.665
Grid4D	25.24	.685
FAGS	25.63	.719
MoProGS (ours)	**25.97**	**.738**

Table 4: **Component analysis on D-NeRF** (mean over seven scenes).

Variant	PSNR↑	SSIM↑	LPIPS↓	ΔPSNR
MoProGS (full)	43.13	.995	.005	–
w/o response decomposition	42.93	.994	.007	−0.20
w/o flatness confidence q	43.01	.994	.007	−0.12
w/o detached routing	42.96	.994	.007	−0.17
routing from canonical state	43.03	.994	.007	−0.10

and the common five-scene protocol of synchronized multi-view Neu3D (Li et al., 2022), and report PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) whenever the comparison source provides them. Protocols and implementation details are summarized in the Reproducibility Statement.

4.2 Comparison with prior methods

Quantitative results. As shown in Tabs. 1–3, MoProGS improves the strongest published D-NeRF average by 0.38 dB and achieves the best PSNR on all seven scenes, with the largest gains on the fast articulated limbs of Jumping Jacks (+0.86 dB) and the thin structures of T-Rex (+0.77 dB). On synchronized Neu3D, where multiple cameras observe the same scene state, MoProGS achieves the best average PSNR, SSIM, and LPIPS, with the best LPIPS in every scene and a 39% lower average LPIPS than the strongest baseline. The gain therefore persists when monocular camera–time coupling is removed. On HyperNeRF, MoProGS reaches 25.97 dB and 0.738 SSIM, 0.34 dB and 0.019 above FAGS.

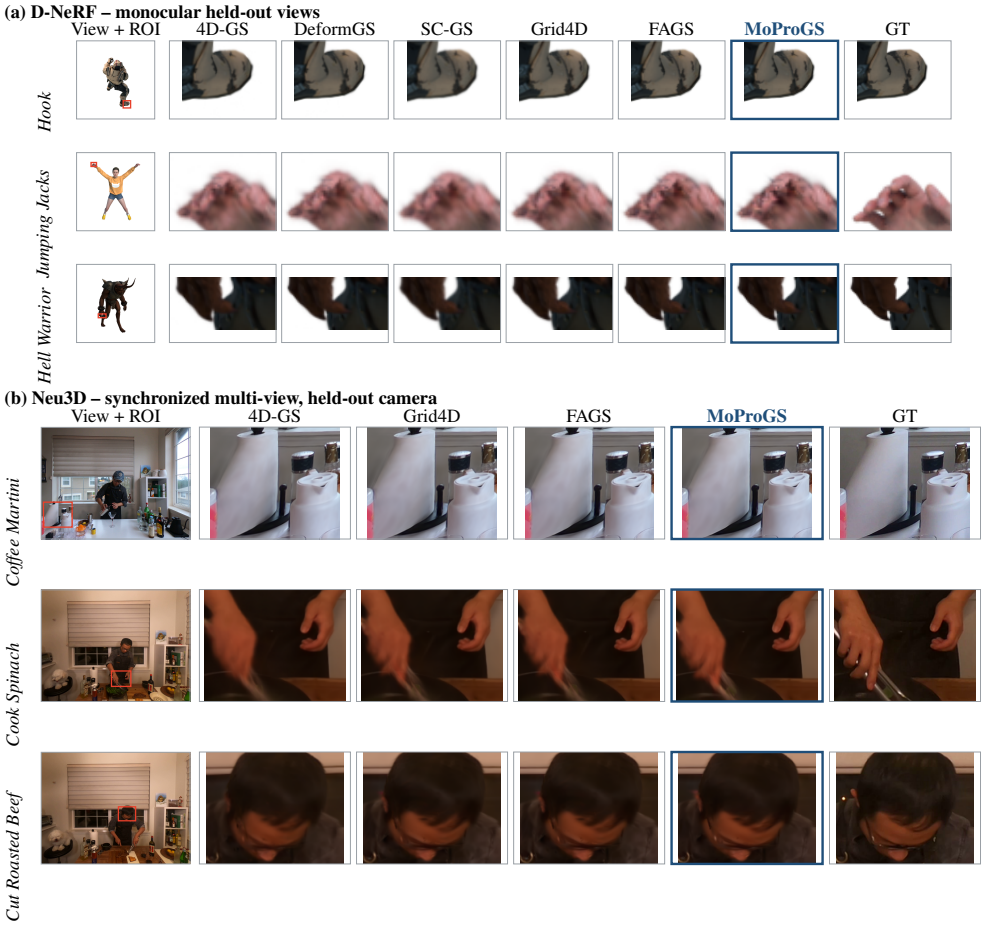

Figure 3: **Qualitative comparison on held-out views.** (a) D-NeRF (monocular) and (b) Neu3D (synchronized multi-view). The red box in the first image of each row marks the crop shared by all methods, and the MoProGS column is outlined in blue. Additional sequences are in Supplement Sec. F.

Qualitative results. Fig. 3 focuses on thin structures and object boundaries, where center displacement and normal support compete to explain the same image edge. On D-NeRF, the worst 5% of pixels carry 55% of the total residual (Supplement Sec. F).

4.3 Component Analysis

Response decomposition and support design. Sharing the temporal response reduces PSNR by 0.20 dB, the largest drop. Without q, near-isotropic Gaussians are contracted along an axis that is not a reliable normal (-0.12 dB); without detachment, the loss can rotate the thin axis to change the routing cue itself (-0.17 dB); and canonical-state routing evaluates incidence on geometry that is not rendered (-0.10 dB).

Late support under synchronized capture. With a shared response on synchronized *Cook Spinach*, enabling only late support raises held-out cam00 PSNR from 31.569 to 31.774 dB and lowers L1 by 3.1% (Supplement Sec. E), isolating the effect of late support from both response splitting and monocular camera–time coupling.

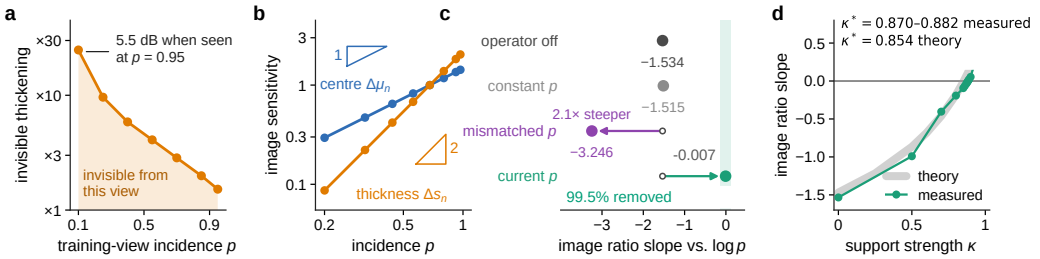

Figure 4: **Projection sensitivity on a controlled Gaussian sheet** (training rasterizer, no time variable). (a) Largest normal thickening that stays invisible (> 40 dB) from a view of incidence p. (b) Image sensitivity to a centre shift and to a thickness change (normalized at $p = 0.68$); triangles mark slopes 1 and 2. (c) Slope of the image-level centre/support ratio for each support variant at $\kappa = 0.88$; zero (shaded) is view independent. (d) The same slope versus support strength, measured and predicted by the edge model.

4.4 Projection orders and observer correspondence

We test Sec. 3.2 on a 7×7 sheet of anisotropic Gaussians with a known thin axis, sweeping $p \in \{0.20, 0.32, \ldots, 0.97\}$ through the training rasterizer with routing held fixed and no time variable (Supplement Sec. B).

A single view cannot separate the two explanations (Fig. 4a). At $p = 0.10$, a $24.8\times$ increase in normal thickness remains within 40 dB of the unperturbed render. The same change falls to 5.5 dB when observed at $p = 0.95$ (Supplement Sec. B). The two perturbations also follow the predicted projection orders. Their log–log slopes are 1.000 for E_μ, 1.999 for E_s, and -0.999 for the ratio (Fig. 4b). With the full kernel and image gradients, the image-level ratio scales as $p^{-1.53}$.

The correction must match the current observer (Fig. 4c). Current-camera incidence changes the image-level ratio slope from -1.534 to -0.007 at the balance point $\kappa = 0.88$ (-0.066 at $\kappa = 0.86$). A global constant leaves the slope at -1.515, while incidence from the wrong view makes it -3.246. A parameter-free edge model (Fig. 4d) predicts the baseline exponent within 4.2%, tracks six sheet geometries with a mean absolute error of 0.080, and predicts $\kappa^\star = 0.854$ against the measured 0.870–0.882 (Supplement Sec. B).

4.5 Error attribution in trained models

Synchronized Neu3D fixes the scene state while varying the observer: at frame 150, all 20 training cameras observe the same camera-blind geometry (Fig. 5a). For the five full-system checkpoints, we estimate $J_\mu = \partial I / \partial(\delta / s_n)$ and $J_s = \partial I / \partial \log s_n$ with 32 Rademacher probes (Hutchinson, 1989) and fit the within-Gaussian slope of $\log(\|J_s\|/\|J_\mu\|)$ on $\log p$. This comparison removes primitive-specific offsets and measures only the cross-view change (Supplement Sec. C).

With the operator off, $\log R$ increases with incidence for every scene, matching the observer dependence predicted by Eq. 4. The fraction of the L1 gradient attributed to thickness rises from below 1% at $p < 0.1$ to 63–79% at $p > 0.9$, and the within-Gaussian slope ranges from 0.97 to 1.33 (Fig. 5b,c).

Applying support with the current observer removes 48% of this slope on average and up to 65%, whereas frame-mean and shuffled incidence do not reduce it. The correction therefore depends on matching the projection factor to the current camera, rather than on contraction alone; one learned κ shared by primitives with varying q makes it partial (Supplement Sec. A.6). A held-out reconstruction control gives the same trend. On *Cook Spinach*, current p yields lower MAE than frame-mean p on 96.3% of the 300 cam00 frames (Supplement Sec. C).

4.6 Camera–time leakage and camera placement

We next test whether monocular acquisition writes view-specific corrections into temporal geometry. In a frozen-scene intervention, 56 unseen camera–time recombinations lose 2.903 dB even though the physical scene is unchanged. When multiple cameras instead share each time under a matched ray budget, full-grid PSNR improves by 3.253 dB (Supplement Sec. D). A random one-to-one label

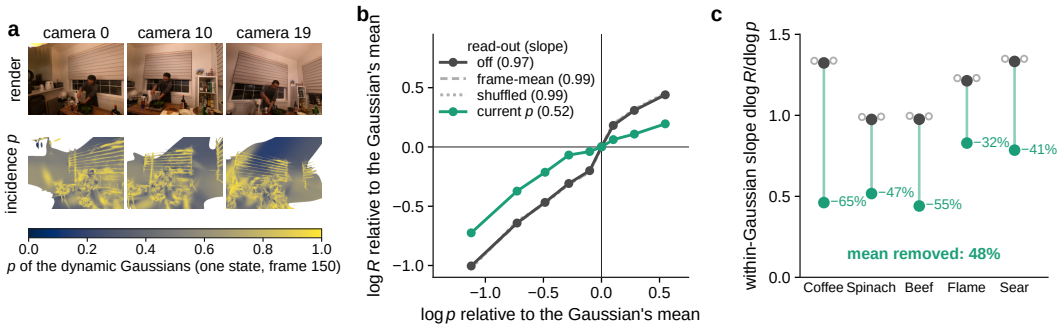

Figure 5: **Cross-view error attribution in trained Neu3D models.** (a) One camera-blind state (Cook Spinach, frame 150) seen by three of the 20 training cameras, with the incidence p of its dynamic Gaussians. (b) Within-Gaussian relation between incidence and the thickness/position response ratio, pooled over all cameras. (c) Slope per scene with the operator off (dark), current p (green), and frame-mean or shuffled p (open); percentages give the removed fraction.

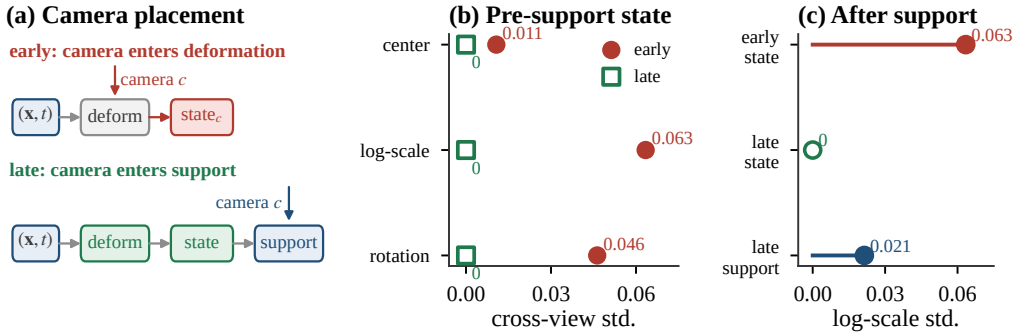

Figure 6: **Where camera information enters.** (a) Early conditioning feeds the camera into deformation, whereas MoProGS introduces it at support. (b) Pre-support quantities under late placement are view-invariant. (c) View variation is confined to the rendered support.

assignment yields at least the same gap as a smooth ordering (4.62 vs. 4.32 dB), so a unique time–camera pairing suffices to store camera-correlated fitting in time. MoProGS keeps the deformation field camera-agnostic and introduces view information at projection, where it acts on the rendered support of each frame and leaves the scene state shared across views.

Fig. 6 compares this late placement with early camera conditioning. Early conditioning produces cross-view standard deviations of 0.011 in center, 0.063 in log-scale, and 0.046 in rotation. The shape parameters carry most of this variation, matching Sec. 3.2, where normal support shows the stronger incidence dependence. Under late placement, all pre-support quantities are view-invariant, and view variation appears in the rendered log-scale with a standard deviation of 0.021, one third of the early-conditioning value (Supplement Sec. D).

5 Conclusion

We introduced MoProGS, a motion–projection factorization for dynamic Gaussian splatting. Our analysis shows that center displacement and normal thickness enter the image at first and second order in incidence, and their relative attribution varies with the observing camera. MoProGS keeps temporal deformation camera-blind, decouples the geometric responses of position and shape, and calibrates normal support with a geometry–view coupling factor before rasterization. By introducing view information at projection rather than into the scene representation, MoProGS attributes image-space errors to geometry more consistently across viewpoints and improves novel-view synthesis in both monocular and multi-view settings.

REPRODUCIBILITY STATEMENT

Sec. 3 specifies the camera-blind response adapters, the geometry–view coupling, and the projection-conditioned support operator, and Supplement Sec. G lists the remaining implementation details. The MoProGS module is identical across datasets, and controlled studies change only the stated factor while keeping the data, initialization, ray budget, optimizer, and backbone fixed. The adapters are initialized as identities, the support strength $\kappa = \kappa_{\max} \operatorname{sigmoid}(\eta)$ with $\kappa_{\max} = 4$ starts near zero, and $\xi_{\min} = 0.1$. D-NeRF models are trained for 80k iterations on one RTX 3090, and the other benchmarks follow the released schedules of their backbones. Evaluation follows the standard protocols: D-NeRF at 800×800 with 20 held-out frames per sequence, the HyperNeRF *vrig* split, and Neu3D with cam00 held out for all 300 frames at 1352×1014. Supplement Sec. A states the assumptions and gives complete derivations of the projection orders, the edge model, and the calibrated response ratio. Supplement Secs. B, C, D, and E describe the protocols of the controlled sheet experiments, the attribution estimator, the camera–time interventions, and the component controls, and Supplement Sec. F documents the qualitative frame selection and crop protocol.

AI USE STATEMENT

We used large language model assistants to aid and polish the writing of this paper, including rephrasing, improving clarity, and checking the consistency of the text. The research ideas, methods, experiments, results, and conclusions are those of the authors. The authors reviewed and verified all AI-assisted text and take full responsibility for the content of the paper.

REFERENCES

Benjamin Attal, Jia-Bin Huang, Christian Richardt, Michael Zollhöfer, Johannes Kopf, Matthew O'Toole, and Changil Kim. HyperReel: High-fidelity 6-DoF video with ray-conditioned sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16610–16620, 2023.

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.

Ang Cao and Justin Johnson. HexPlane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 130–141, 2023.

Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using Gaussian surfels. In *ACM SIGGRAPH Conference Papers*, pp. 22:1–22:11, 2024.

Yuanxing Duan, Fangyin Wei, Qiyu Dai, Yuhang He, Wenzheng Chen, and Baoquan Chen. 4D-Rotor Gaussian splatting: Towards efficient novel view synthesis for dynamic scenes. In *ACM SIGGRAPH Conference Papers*, pp. 87:1–87:11, 2024.

Jiemin Fang, Taoran Yi, Xinggang Wang, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Matthias Nießner, and Qi Tian. Fast dynamic radiance fields with time-aware neural voxels. In *SIGGRAPH Asia Conference Papers*, pp. 11:1–11:9, 2022.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12479–12488, 2023.

Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5712–5721, 2021.

Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Monocular dynamic view synthesis: A reality check. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 33768–33780, 2022.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11): 665–673, 2020.

Antoine Guédon and Vincent Lepetit. SuGaR: Surface-aligned Gaussian splatting for efficient 3D mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5354–5363, 2024.

Jiaheng Hu, Zhizhong Zhang, Jingyu Gong, Lizhuang Ma, Xin Tan, and Yuan Xie. Multi-step deformable Gaussian splatting for dynamic scene rendering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(6):4834–4842, 2026.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2D Gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH Conference Papers*, pp. 32:1–32:11, 2024a.

Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. SC-GS: Sparse-controlled Gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4220–4230, 2024b.

Yi-Hua Huang, Ming-Xian Lin, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Deformable radial kernel splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21513–21523, 2025.

Michael F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics – Simulation and Computation*, 18(3):1059–1076, 1989.

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.

Jiahui Lei, Yijia Weng, Adam W. Harley, Leonidas Guibas, and Kostas Daniilidis. MoSca: Dynamic Gaussian fusion from casual videos via 4D motion scaffolds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6165–6177, 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 (CVPR)*, pp. 5521–5531, 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 (CVPR)*, pp. 8508–8520, 2024.

Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6498–6508, 2021.

Zhengqi Li, Qianqian Wang, Forrester Cole, Richard Tucker, and Noah Snavely. DynIBaR: Neural dynamic image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4273–4284, 2023.

Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-Flow: 4D reconstruction with dynamic 3D Gaussian particle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21136–21145, 2024.

Yu-Lun Liu, Chen Gao, Andréas Meuleman, Hung-Yu Tseng, Ayush Saraf, Changil Kim, Yung-Yu Chuang, Johannes Kopf, and Jia-Bin Huang. Robust dynamic radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13–23, 2023.

Jiahao Lu, Jiacheng Deng, Ruijie Zhu, Yanzhe Liang, Wenfei Yang, Tianzhu Zhang, and Xu Zhou. DN-4DGS: Denoised deformable network with temporal-spatial aggregation for dynamic scene rendering. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 84114–84138, 2024.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3D Gaussians: Tracking by persistent dynamic view synthesis. In *International Conference on 3D Vision (3DV)*, pp. 800–809, 2024.

Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7210–7219, 2021.

Qiaowei Miao, JinSheng Quan, Kehan Li, Yichao Xu, Yi Yang, and Yawei Luo. Frequency-aware dynamic Gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2026.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision (ECCV)*, pp. 405–421, 2020.

Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B. Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5865–5874, 2021a.

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):238:1–238:12, 2021b.

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 (CVPR)*, pp. 10318–10327, 2021.

Ruizhi Shao, Zerong Zheng, Hanzhang Tu, Boning Liu, Hongwen Zhang, and Yebin Liu. Tensor4D: Efficient neural 4D decomposition for high-fidelity dynamic reconstruction and rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16632–16642, 2023.

Liangchen Song, Anpei Chen, Zhong Li, Zhang Chen, Lele Chen, Junsong Yuan, Yi Xu, and Andreas Geiger. NeRFPlayer: A streamable dynamic scene representation with decomposed neural radiance fields. *IEEE Transactions on Visualization and Computer Graphics*, 29(5):2732–2742, 2023.

Colton Stearns, Adam W. Harley, Mikaela Uy, Florian Dubost, Federico Tombari, Gordon Wetzstein, and Leonidas Guibas. Dynamic Gaussian marbles for novel view synthesis of casual monocular videos. In *SIGGRAPH Asia Conference Papers*, pp. 30:1–30:11, 2024.

Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 3DGStream: On-the-fly training of 3D Gaussians for efficient streaming of photo-realistic free-viewpoint videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20675–20685, 2024.

Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Christoph Lassner, and Christian Theobalt. Non-rigid neural radiance fields: Reconstruction and novel view synthesis of a dynamic scene from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12959–12970, 2021.

Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5491–5500, 2022.

Feng Wang, Sinan Tan, Xinghang Li, Zeyue Tian, Yafei Song, and Huaping Liu. Mixed neural voxels for fast multi-view video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19706–19716, 2023.

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4D reconstruction from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9660–9672, 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 (CVPR)*, 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 (ICLR)*, 2025.

Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9421–9431, 2021.

Jiawei Xu, Zexin Fan, Jian Yang, and Jin Xie. Grid4D: 4D decomposed hash encoding for high-fidelity dynamic Gaussian splatting. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 123787–123811, 2024.

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 (ICLR)*, 2024a.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3D Gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20331–20341, 2024b.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3D Gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19447–19456, 2024a.

Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 43(6):271:1–271:13, 2024b.

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 Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

Matthias Zwicker, Hanspeter Pfister, Jeroen van Baar, and Markus Gross. EWA volume splatting. In *Proceedings of IEEE Visualization*, pp. 29–36, 2001.

Supplementary Material

This supplement presents theoretical derivations and controlled empirical analyses that support the main paper. Section A derives the projection responses of position and normal support. It connects their incidence orders to observer dependent geometric attribution. Section B verifies these predictions on known geometry with the training rasterizer. Section C measures the same attribution behavior in trained 4DGS models. Section D analyzes how camera and time correlation transfers observer dependent fitting into temporal geometry. It motivates camera conditioning at projection. Section E evaluates the projection stage design choices of MoProGS. Section F reports extended reconstruction evidence. Section G gives the implementation details and experimental protocols used for reproduction. Section H discusses limitations.

A Projection ambiguity and attribution across views

MoProGS is motivated by a fundamental asymmetry in perspective projection. For a Gaussian representing a local surface, displacement along the thin axis changes the projected center at first order in incidence, while normal support changes the projected covariance at second order. The relative image response of these two geometric directions therefore varies with the observer even when the 3D Gaussian is fixed. We derive both response orders, connect them to gradient attribution, and analyze how late projection support modifies the leading incidence dependence. Section B verifies each prediction with the training rasterizer.

A.1 Local projection geometry and assumptions

Consider a Gaussian with center $\boldsymbol{\mu} \in \mathbb{R}^3$, covariance $\boldsymbol{\Sigma}$, and camera projection π. Define

$$\mathbf{J} = \left.\frac{\partial \pi(\mathbf{x})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\boldsymbol{\mu}} \in \mathbb{R}^{2\times 3} \tag{9}$$

as the local projection Jacobian. Let $\mathbf{v}$ denote the unit direction from the Gaussian to the camera, and let $\mathbf{P} = \mathbf{I} - \mathbf{v}\mathbf{v}^\top$ project onto $\mathbf{v}^\perp$. For a pinhole camera expressed in camera coordinates,

$$\pi(X, Y, Z) = \left(f_x X/Z + c_x,\ f_y Y/Z + c_y\right), \qquad \mathbf{J} = \begin{bmatrix} f_x/Z & 0 & -f_x X/Z^2 \\ 0 & f_y/Z & -f_y Y/Z^2 \end{bmatrix}. \tag{10}$$

Multiplication by the camera ray vector $(X, Y, Z)^\top$ gives zero. The viewing ray therefore spans the null space of the projection Jacobian, so $\mathbf{J}\mathbf{v} = 0$ for the corresponding unit direction. Let $\mathbf{n}$ denote the smallest scale principal axis of a Gaussian representing a local surface, and define

$$p = \|\mathbf{P}\mathbf{n}\| = \sqrt{1 - |\mathbf{n}^\top \mathbf{v}|^2} \in [0, 1]. \tag{11}$$

The value $p = 0$ corresponds to a thin axis parallel to the viewing ray and hence to front facing incidence. The limit $p \to 1$ corresponds to grazing incidence.

Let $\sigma_{\min}$ and $\sigma_{\max}$ denote the two singular values of $\mathbf{J}$ restricted to $\mathbf{v}^\perp$. They remain positive for a finite and nondegenerate camera configuration. The asymptotic analysis treats incidence p as the variable of interest. Focal length, depth, Gaussian scale, and the remaining local projection factors are fixed or bounded away from degeneracy. The controlled experiments in Section B evaluate the full rendered image response. They include kernel, compositing, visibility, and image gradient effects.

A.2 Proposition 1: thin axis center motion is first order in p

Consider a scalar displacement d along the thin axis, with $\boldsymbol{\mu}(d) = \boldsymbol{\mu} + d\mathbf{n}$. Decompose

$$\mathbf{n} = \mathbf{P}\mathbf{n} + (\mathbf{n}^\top \mathbf{v})\mathbf{v}. \tag{12}$$

Since $\mathbf{J}\mathbf{v} = 0$, only the component of $\mathbf{n}$ orthogonal to the viewing ray contributes to the projected displacement, and $\mathbf{J}\mathbf{n} = \mathbf{J}\mathbf{P}\mathbf{n}$. The vector $\mathbf{P}\mathbf{n}$ lies in $\mathbf{v}^\perp$ and has norm p. The restricted singular value bounds give

$$\sigma_{\min} p \le \|\mathbf{J}\mathbf{n}\|_2 \le \sigma_{\max} p. \tag{13}$$

The projected center sensitivity is therefore

$$E_\mu \equiv \left\| \frac{\partial \boldsymbol{\mu}^{2D}}{\partial d} \right\|_2 = \|\mathbf{J}\mathbf{n}\|_2 = \Theta(p). \tag{14}$$

Equivalently, an infinitesimal displacement satisfies $\|\mathbf{J}(d\mathbf{n})\|_2^2 = \Theta(d^2p^2)$. The first order result concerns the amplitude of the projected displacement. Squaring that amplitude produces the corresponding second order dependence in the squared norm.

A.3 Proposition 2: thin axis support is second order in p

Write the 3D covariance in principal axes,

$$\boldsymbol{\Sigma} = \sum_j s_j^2 \mathbf{r}_j \mathbf{r}_j^\top, \tag{15}$$

and identify the thin axis with $\mathbf{n} = \mathbf{r}_n$ and scale s_n. Its contribution is

$$C_n = s_n^2 (\mathbf{J}\mathbf{n})(\mathbf{J}\mathbf{n})^\top \tag{16}$$

to the screen space covariance $\boldsymbol{\Sigma}^{2D} = \mathbf{J}\boldsymbol{\Sigma}\mathbf{J}^\top$. Let $\mathbf{u} = \mathbf{J}\mathbf{n}$. Since $\|\mathbf{u}\mathbf{u}^\top\|_F = \|\mathbf{u}\|_2^2$,

$$\frac{\partial C_n}{\partial s_n} = 2 s_n \mathbf{u}\mathbf{u}^\top, \qquad E_s \equiv \left\| \frac{\partial C_n}{\partial s_n} \right\|_F = 2 s_n \|\mathbf{J}\mathbf{n}\|_2^2. \tag{17}$$

Combining Eqs. 13 and 17,

$$2 s_n \sigma_{\min}^2 p^2 \le E_s \le 2 s_n \sigma_{\max}^2 p^2, \tag{18}$$

which gives $E_s = \Theta(p^2)$. The same incidence order holds for the log scale parameterization used in the trained model probes. With $\rho_n = \log s_n$,

$$\left\| \frac{\partial C_n}{\partial \rho_n} \right\|_F = 2 s_n^2 \|\mathbf{J}\mathbf{n}\|_2^2 = \Theta(p^2). \tag{19}$$

Therefore the second order incidence dependence is invariant to whether the normal scale is parameterized by s_n or by $\log s_n$.

A.4 Relative geometric response varies with the observer

For the scalarized displacement and normal scale directions above, Eqs. 14 and 17 give

$$\frac{E_\mu}{E_s} = \frac{1}{2 s_n \|\mathbf{J}\mathbf{n}\|_2} = \Theta\left(\frac{1}{p}\right). \tag{20}$$

The multiplicative constant depends on parameter normalization and local camera scale, while the incidence exponent remains unchanged. If, in addition, the direction of $\mathbf{P}\mathbf{n}$ within $\mathbf{v}^\perp$ and the local projection are held fixed, as in main-text Eq. 4, then $\|\mathbf{J}\mathbf{n}\|_2$ is exactly proportional to p and the Θ relation becomes the proportionality $E_\mu / E_s \propto 1/p$. Consequently, two cameras observing the same 3D Gaussian can induce different relative image responses for position and normal support before any temporal model is introduced.

The full rendered image Jacobian additionally depends on the Gaussian kernel, alpha compositing, visibility, image gradients, and the current residual. A local blurred edge model makes the resulting correction to the geometric $1/p$ law explicit. Assume that the thin axis and one tangent axis lie in the plane spanned by the viewing ray and the surface normal. Let f denote the focal length in pixels, z the camera distance, s_t the tangent scale in this plane, s_n the normal scale, ξ the multiplicative normal support factor, and $\sigma_{\text{lp}}^2 = 0.3\,\text{px}^2$ the rasterizer low pass variance. The projected variance along the edge normal is

$$\sigma^2(p) = \left(\frac{f}{z}\right)^2 \left[(\xi s_n p)^2 + s_t^2 (1 - p^2)\right] + \sigma_{\text{lp}}^2. \tag{21}$$

For a blurred step edge $I(x) = A\,\Phi((x_e - x)/\sigma)$, with Φ the standard normal CDF and $\phi = \Phi'$, direct integration gives

$$\left\| \frac{\partial I}{\partial x_e} \right\|_2 = |A| \|\phi\|_2 \, \sigma^{-1/2}, \qquad \left\| \frac{\partial I}{\partial \sigma} \right\|_2 = |A| \|u\phi(u)\|_2 \, \sigma^{-1/2}. \tag{22}$$

For a thin axis center displacement d, $\partial x_e/\partial d = (f/z)p$, while

$$\frac{\partial \sigma}{\partial s_n} = \frac{(f/z)^2 \xi^2 s_n p^2}{\sigma}. \tag{23}$$

Since $\|\phi\|_2/\|u\phi(u)\|_2 = \sqrt{2}$, the continuous edge model yields

$$\frac{E_\mu}{E_s} = \sqrt{2}\,\frac{z}{f}\,\frac{\sigma(p)}{\xi^2 s_n p}. \tag{24}$$

Equation 24 predicts the additional incidence dependence through the projected kernel width $\sigma(p)$. Section B compares this prediction with the training rasterizer across multiple sheet geometries and support strengths.

A.5 Gradient attribution from a single residual across views

The difference in projection order affects optimization when two parameter directions induce overlapping image changes. Consider the squared image loss $\mathcal{L} = \frac{1}{2}\|\mathbf{r}\|_2^2$ and a first order linearization of the renderer around the current state. Let scalar direction T generate the residual, so $\mathbf{r} = J_T \Delta_T$ to first order. For another scalarized direction O, the initial gradients satisfy

$$|g_O| = |J_O^\top J_T \Delta_T|, \qquad |g_T| = \|J_T\|_2^2 |\Delta_T|. \tag{25}$$

Their magnitude ratio is therefore

$$\chi_{T\to O} \equiv \frac{|g_O|}{|g_T|} = |\cos(J_O, J_T)|\,\frac{\|J_O\|_2}{\|J_T\|_2}. \tag{26}$$

If we normalize only these two competing directions, the fraction of their combined initial gradient magnitude assigned to direction O is

$$\mathrm{share}_{\mathrm{wrong}} = \frac{\chi_{T\to O}}{1 + \chi_{T\to O}}. \tag{27}$$

Equation 26 factors attribution into the overlap between the two image Jacobians and the ratio of their response magnitudes. The overlap term is $|\cos(J_O, J_T)|$, while the response term is $\|J_O\|/\|J_T\|$. The leading $1/p$ dependence from Eq. 20 enters through the response term even when the overlap varies little across observers. A residual with a fixed physical cause can therefore induce different optimization preferences under different views. Section B verifies this factorization with residuals of known cause, and Section C measures the same effect in trained checkpoints.

A.6 Effect of late projection support on incidence dependence

MoProGS applies a normal axis support factor after the camera blind deformation state has been formed,

$$\xi_n = \mathrm{clip}(1 - \kappa\ \mathrm{sg}(q)\,\mathrm{sg}(p), \xi_{\min}, 1), \qquad \xi_{t_1} = \xi_{t_2} = 1. \tag{28}$$

The stop gradient determines how this factor enters optimization. During each backward pass, q and p serve only as routing variables, so ξ_n is constant with respect to the optimized geometry parameters. Changing the observer triggers a new forward evaluation of p and therefore a new value of ξ_n. The operator can consequently reshape the response across views without creating a gradient path through the routing cue.

When the thin axis direction and local projection are fixed during differentiation, the calibrated projected covariance becomes

$$\widetilde{C} = C_{\tan} + \xi_n^2 s_n^2 (\mathbf{Jn})(\mathbf{Jn})^\top. \tag{29}$$

Its normal scale sensitivity is $\widetilde{E}_s = 2 s_n \xi_n^2 \|\mathbf{Jn}\|_2^2$, while the projected center sensitivity remains $E_\mu = \|\mathbf{Jn}\|_2$. Therefore

$$\frac{\widetilde{E}_s}{E_\mu} = 2 s_n \xi_n^2 \|\mathbf{Jn}\|_2 \propto p\,\xi_n^2. \tag{30}$$

Without calibration, the leading support response relative to position grows linearly with p. Within the unclipped active interval, define $\alpha = \kappa q$, which gives $\xi_n = 1 - \alpha p$. The logarithmic incidence

slope of this isolated term is

$$\frac{\mathrm{d}\log[p(1-\alpha p)^2]}{\mathrm{d}\log p} = 1 - \frac{2\alpha p}{1-\alpha p}. \tag{31}$$

The uncalibrated slope equals 1. The second term has the opposite sign and therefore reduces the leading observer dependence. The isolated slope vanishes locally when $\alpha p = 1/3$. This condition characterizes the leading incidence term. The full image Jacobian also includes kernel width, visibility, compositing, image gradients, the distribution of q, and clipping. When the support floor becomes active, ξ_n is locally constant and the isolated slope returns to 1. Section B evaluates the balanced operating point with the full rasterizer.

A.7 Camera dependence after temporal deformation

The analysis above identifies projection as the stage that requires camera dependence. The temporal geometric state itself need not depend on the camera. Let the geometric state before projection support be

$$\theta^t = D_\psi(\theta^0, t), \tag{32}$$

with no camera input. For two cameras c_1 and c_2 observing the same time,

$$\theta^{t,c_1} = \theta^{t,c_2} = \theta^t, \tag{33}$$

by construction. Camera dependence enters through the support used for rendering, $\widetilde{\boldsymbol{\Sigma}}^{t,c}$. The current observer sets $p = p(\theta^t, c)$ after deformation. This factorization keeps the scene state shared across cameras and retains view dependence in projected support. As with the screen-space low-pass filters of EWA splatting and Mip-Splatting, this view dependence is confined to rendering; we regard the camera-blind pre-support state θ^t as the reconstructed geometry.

Monocular capture introduces an additional camera and time ambiguity under this factorization. With one observer at each time, a camera blind temporal field can use t as an indirect key for the training camera. Section D isolates this mechanism with a static physical scene and breaks the correspondence using simultaneous observations from multiple cameras.

B Controlled verification with known geometry

The derivation in Section A yields three testable predictions. Center motion and normal support should exhibit first and second order incidence responses. Their relative response should vary with the observer and should alter gradient attribution. Late projection support should reduce this variation when the incidence cue is evaluated from the current observer. We test all three predictions on a fixed Gaussian sheet with known geometry. These experiments contain neither a time variable nor a learned deformation field.

B.1 Controlled sheet and measurement protocol

The controlled scene is a 7×7 sheet of overlapping anisotropic Gaussians. The thin axis is known by construction. The tangent scale is 0.02, the normal scale is 0.002, opacity is 0.8, and Gaussian spacing is 0.025. We render 512×512 images with a 60° field of view. Unless stated otherwise, the camera distance is 6 and the mean flatness confidence is $q \simeq 0.90$.

The derivative and edge model experiments use the predefined incidence values

$$p \in \{0.20, 0.32, 0.44, 0.56, 0.68, 0.80, 0.92, 0.97\}. \tag{34}$$

The observability experiment uses

$$p \in \{0.10, 0.25, 0.40, 0.55, 0.70, 0.85, 0.95\}. \tag{35}$$

Image derivatives are computed with central finite differences using a base step of 2.5×10^{-4}. When projection support is active, the routing variables are held fixed during differentiation, consistent with the detached routing in Eq. 28. Each control varies only the declared observer, geometric parameter, or support setting.

B.2 DIRECT VERIFICATION OF THE p AND p^2 ORDERS

We first measure the geometric response before introducing the image kernel. For a center displacement along the normal direction, the response is $E_\mu = \|\mathbf{Jn}\|$. For a perturbation of the thin axis scale, the response is the corresponding derivative of projected covariance. Fitting the incidence sweep in log space gives slopes

$$1.00006 \quad \text{for center,} \qquad 1.99934 \quad \text{for support,} \qquad -0.99928 \quad \text{for } E_\mu/E_s. \tag{36}$$

The measured exponents recover the first and second order predictions of Eqs. 14 and 18 to within 0.1%. This measurement precedes the image kernel and residual and is independent of the support variant. It therefore verifies the incidence order of the projection terms directly.

B.3 SINGLE VIEW AMBIGUITY OF THE 3D EXPLANATION

We next construct two controlled geometric alternatives. State A displaces every Gaussian center by $2s_n$ along the thin axis. State $B(m)$ keeps the centers fixed and multiplies the normal scale by m. Photometric quantities are evaluated over the union of the rendered footprints, which prevents the empty background from dominating PSNR.

Table 5 quantifies the geometric ambiguity permitted by one training view. At $p = 0.10$, the normal thickness can increase by $24.80\times$ while the rendered image remains within 40 dB of the unperturbed sheet. At $p = 0.95$, the largest admissible multiplier decreases to $1.52\times$. The fixed two sigma center displacement follows the opposite trend and becomes increasingly visible as incidence grows. The ambiguity is specific to the observer. The largest thickness accepted at $p = 0.10$ produces only 5.5 dB when rendered from $p = 0.95$. The lower multiplier reaches the search limit of $0.02\times$ for every view, which reflects the weak visibility of further thinning when the normal axis is already below pixel scale.

Table 5: **Geometric ambiguity under a single view.** We report normal thickness multipliers whose render remains within 40 dB of the unperturbed sheet over its own footprint, together with the visibility of a normal displacement equal to two thin axis standard deviations. The final columns evaluate the largest admissible thickening under the most grazing observer.

training view p	admissible thickness range multiplier	displacement PSNR dB	cross view evaluation worst view p	PSNR dB
0.10	$\times 0.02$–$\times 24.80$	47.9	0.95	5.5
0.25	$\times 0.02$–$\times 9.62$	39.9	0.95	10.5
0.40	$\times 0.02$–$\times 5.86$	35.6	0.95	15.4
0.55	$\times 0.02$–$\times 4.09$	32.1	0.95	21.0
0.70	$\times 0.02$–$\times 2.85$	28.9	0.95	26.5
0.85	$\times 0.02$–$\times 1.99$	25.0	0.95	33.7
0.95	$\times 0.02$–$\times 1.52$	20.8	0.95	40.5

A single image therefore does not uniquely determine the underlying 3D explanation. The optimization preference among compatible explanations additionally depends on their relative image responses and on the overlap between their image Jacobians. We measure both quantities next.

B.4 RASTERIZED RESPONSE PREDICTED BY THE EDGE MODEL

Equation 24 extends the geometric $1/p$ relation through the projected kernel width. We evaluate it directly from known geometry and camera parameters. Table 6 varies the normal scale, tangent scale, and camera distance over six sheet configurations with projection support disabled. For each configuration, we compute the predicted slope and measure the corresponding slope with the training rasterizer.

Table 6: **Edge model compared with the training rasterizer.** We report the log space slope of the center to support image response ratio over $p \in [0.20, 0.97]$ for six sheet geometries with projection support disabled. The analytic model contains no fitted incidence exponent and predicts the change in slope induced by anisotropy and camera distance.

geometry	s_t	s_n	z	predicted	measured full / boundary
default	0.020	0.002	6	-1.469	-1.544 / -1.534
thin normal	0.020	0.001	6	-1.477	-1.587 / -1.535
thick normal	0.020	0.004	6	-1.438	-1.506 / -1.519
near camera	0.020	0.002	3	-1.604	-1.606 / -1.693
far camera	0.020	0.002	12	-1.266	-1.364 / -1.364
wide tangent	0.040	0.004	6	-1.603	-1.731 / -1.727
mean |predicted − measured|					0.080 / 0.086
balanced strength $\kappa^\star$				0.854	0.870 / 0.882

The mean absolute difference between prediction and measurement is 0.080 over the full image and 0.086 in the boundary band. The same analytic expression also predicts how the response exponent changes with anisotropy and camera distance. For the default sheet, the predicted slope is -1.469, compared with -1.544 over the full image and -1.534 in the boundary band. The relative difference is 4.2% with respect to the boundary band and 4.9% with respect to the full image. The remaining discrepancy reflects rendering effects beyond the local edge approximation.

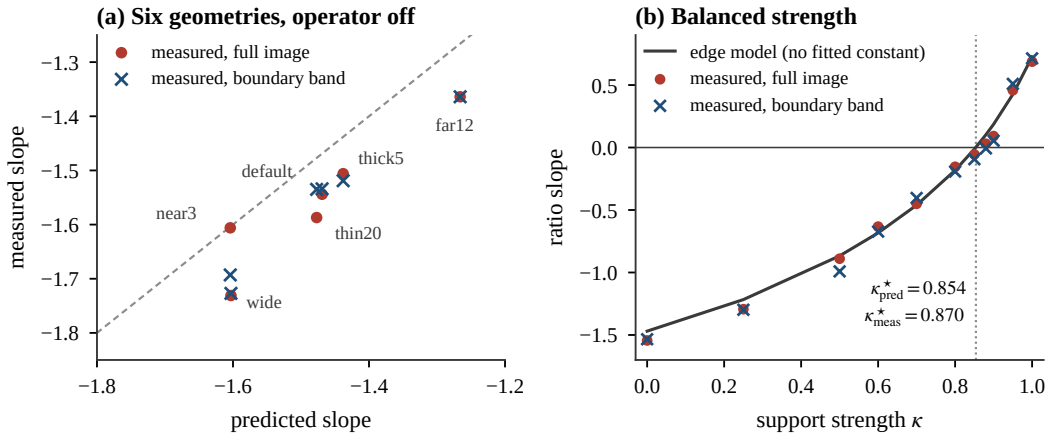

Figure 7: **Analytic prediction of the image response orders.** The left panel compares predicted and measured slopes of the center to support response ratio for six sheet geometries with projection support disabled. The right panel compares the same analytic model with the rasterizer while varying the support strength. The model predicts both the baseline exponent and the balanced support strength.

B.5 Current observer incidence calibrates the response

For the default sheet, the boundary image ratio between center and support response has slope -1.5340 after accounting for kernel width and image gradients. We keep the support law fixed and vary only the source of the incidence value. Table 7 reports the resulting image sensitivity slopes for $\kappa = 0.88$.

Table 7: **Correspondence between observer and incidence.** The table reports slopes in log space fitted to boundary image sensitivity across the incidence sweep. Current incidence is evaluated from the present observer. The mismatched condition evaluates the support cue using a different observer.

Variant	center slope	support slope	ratio slope
projection support disabled	1.6096	3.1435	-1.5340
global incidence constant	1.6158	3.1306	-1.5148
current observer incidence	1.6146	1.6217	-0.0071
mismatched observer incidence	1.6098	4.8556	-3.2459

Current observer incidence reduces the ratio slope from -1.5340 to -0.0071. A global incidence constant leaves the slope at -1.5148. Mismatched observer incidence increases its magnitude to -3.2459. These results identify observer correspondence as the factor that balances center and support response.

B.6 Prediction of the balanced support strength

The measured ratio slope changes continuously from -1.5340 at $\kappa = 0$ to $+0.7132$ at $\kappa = 1$. A finer sweep gives $-0.0659, -0.0369, -0.0226, -0.0071, +0.0087, +0.0233, +0.0546$ at $\kappa = 0.86, 0.87, 0.875, 0.88, 0.885, 0.89, 0.90$. The zero crossing occurs at $\kappa^\star = 0.870$ for the full image and 0.882 for the boundary band. The edge model predicts 0.854 in Fig. 7. We use $\kappa = 0.88$ as the synthetic operating point for the observer correspondence experiments.

B.7 Known cause residuals validate the attribution factorization

We next test Eq. 26 using residuals whose generating geometric direction is known. For each residual, we compare the initial gradient magnitude on the generating direction with the magnitude on the competing direction. The analytic prediction uses the measured Jacobian overlap and response ratio. Across the four support variants in Table 8, predicted and measured attribution shares agree with Pearson correlation 0.96 to 0.98 and mean absolute error 0.03 to 0.05. The column $\mathrm{d}\log R/\mathrm{d}\log p$ in Table 8 uses the support to center ratio $R = \|J_s\|/\|J_\mu\|$ of Section C.

At $p = 0.10$, a residual generated by normal thickness produces $\chi_{s\to\mu} \approx 26$. The initial scalarized gradient magnitude on position is therefore about $26\times$ the magnitude on the generating thickness direction. At high incidence, the ratio decreases to approximately 0.1 to 0.3. The overlap term changes much less. On the synthetic sheet, full image overlap is approximately 0.12 to 0.13, while the one sided boundary band gives 0.52 to 0.59 across support variants. The response ratio therefore accounts for the main observer dependence.

Table 8: **Jacobian overlap and response ratio predict initial attribution.** We keep the physical sheet fixed, vary only the observer, and retain detached routing. The analytic quantity is $\chi = |\cos(J_\mu, J_s)|\, \|J_{\text{other}}\|/\|J_{\text{true}}\|$. It predicts the attribution share $\chi/(1+\chi)$. We compare this prediction with the initial gradient induced by a residual whose geometric cause is known.

support setting	$\mathrm{d}\log R/\mathrm{d}\log p$	$\lvert\cos\rvert$ full	$\lvert\cos\rvert$ band	prediction r	mean $\lvert$error$\rvert$
projection support disabled	1.301	0.125	0.581	0.971	0.035
mismatched observer incidence	2.533	0.118	0.582	0.983	0.028
global incidence constant	1.386	0.126	0.585	0.967	0.051
current observer incidence	0.349	0.116	0.524	0.963	0.051

We estimate the same overlap term in trained models using products across probe responses. The overlap estimate uses twelve probes, fewer than the 32 used for the response norms in Section C, and its noise floor is 0.236. Assuming that probe noise is independent of the true overlap, the squared estimate is approximately the squared overlap plus the squared floor. Removing the floor in quadrature under this assumption yields an overlap of 0.52 to 0.58 that remains nearly constant across incidence. The synthetic and trained measurements therefore show the same pattern. Jacobian overlap is substantial, while the response ratio contains the dominant incidence dependence.

Figure 8 summarizes the two parts of the mechanism. The left panel shows the range of 3D thicknesses that remain compatible with a single observer. The right panel shows how a residual generated by thickness is redistributed toward position as the observer changes.

Summary of controlled verification. The experiments with known geometry validate the sequence of predictions in Section A. Projection produces first and second order incidence responses for center motion and normal support. The edge model predicts the image level exponent after rasterization. A single observer admits substantially different 3D explanations. Their gradient attribution follows Jacobian overlap and relative response. Late support reduces observer dependence when incidence is recomputed from the current camera. These controls isolate the projection mechanism without a temporal field. Sections C and D next examine how the same mechanism appears in trained dynamic models.

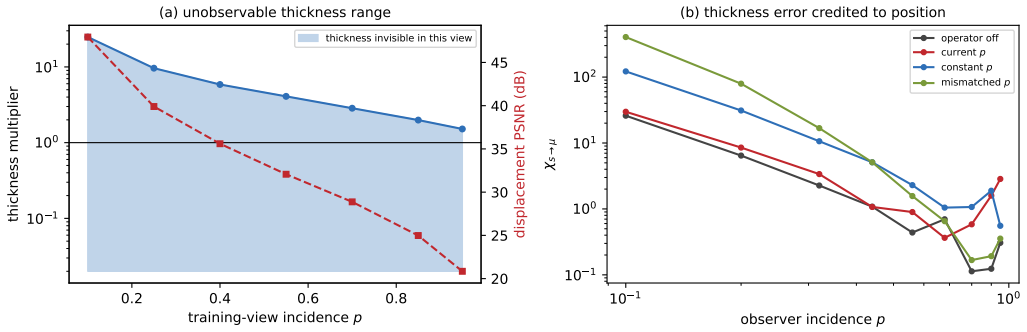

Figure 8: **Single view ambiguity and attribution across observers.** The left panel shows that the admissible normal thickness range contracts as incidence increases, while a fixed two sigma normal displacement becomes more visible. The right panel considers a residual generated by thickness and reports $\chi_{s\to\mu}$, which measures the initial scalarized gradient on position relative to the generating thickness direction. Current observer incidence reduces the dependence on viewing incidence, while constant or mismatched incidence does not.

C ATTRIBUTION ACROSS VIEWS IN TRAINED MODELS

The controlled experiments in Section B isolate the projection mechanism under known geometry. We next test whether the same observer dependence persists in fully trained 4DGS models, where visibility, alpha compositing, texture, kernel overlap, and photometric residuals all contribute to the image Jacobian. We first freeze a trained geometric state and vary only the observer. We then train current incidence and frame mean incidence independently. Finally, we determine whether a persistent primitive statistic can represent the incidence signal required by the projection stage.

C.1 FIXED TRAINED STATE ACROSS MULTIPLE OBSERVERS

We use the five full MoProGS Neu3D checkpoints from the main paper. At physical frame 150, all 20 training cameras observe the same deformed state. We freeze this state and vary only the rasterization camera. Temporal motion is therefore absent from the comparison, and any systematic change within a primitive is caused by the observer at fixed geometry.

Rademacher estimation of image response. For a scalar geometric coordinate a with image Jacobian $j_a = \partial I/\partial a \in \mathbb{R}^{3HW}$, let $w \in \{-1, +1\}^{3HW}$ contain independent Rademacher entries. A backward pass of $\langle w, I\rangle$ returns $w^\top j_a$, and

$$\mathbb{E}_w\left[(w^\top j_a)^2\right] = j_a^\top \mathbb{E}[ww^\top] j_a = \|j_a\|_2^2. \tag{37}$$

Repeated probes therefore provide an unbiased estimate of the squared image response without explicitly constructing the full image Jacobian. We use 32 probes per camera. The two coordinates are displacement along the thin axis normalized by the normal scale, $a_\mu = \delta/s_n$, and normal log

thickness, $a_s = \log s_n$. We denote their response norms by $\|J_\mu\|$ and $\|J_s\|$ and analyze

$$R_{iv} = \frac{\|J_{s,iv}\|}{\|J_{\mu,iv}\|} \tag{38}$$

for primitive i under camera v. Section A shows that the second order incidence dependence is unchanged under the log scale parameterization. We also evaluate gradients of the actual L1 loss with respect to the same two coordinates and report the thickness contribution $g_s^2/(g_s^2 + g_\mu^2)$.

Regression within each Gaussian. Absolute response magnitude varies with scale, opacity, color, visibility, and local image structure. We remove primitive specific offsets by comparing each Gaussian with itself across cameras. Define

$$x_{iv} = \log p_{iv} - \overline{\log p_i}, \qquad y_{iv} = \log R_{iv} - \overline{\log R_i}, \tag{39}$$

and fit the pooled within Gaussian slope

$$\widehat{\beta} = \frac{\sum_{i,v} x_{iv} y_{iv}}{\sum_{i,v} x_{iv}^2}. \tag{40}$$

Subtracting the primitive means removes persistent offsets associated with Gaussian identity. The fitted slope measures cross observer change within each primitive. A primitive contributes to the estimate when it is rasterized by at least three cameras with $p > 10^{-4}$. A stricter filter requiring a spread greater than 0.15 in $\log p$ changes the support disabled slopes by 0.019 to 0.032. The removed fraction changes by at most 0.01.

Attribution consistency. We call the geometric error attribution of a primitive viewpoint-consistent if R_{iv} does not vary systematically with p_{iv} across cameras observing the same pre-support state, that is, if the within Gaussian elasticity $\beta = \mathrm{d}\log R/\mathrm{d}\log p$ vanishes. Equation 40 estimates this quantity, and $\beta = 0$ is the target of projection support. The property concerns the relative sensitivity of the renderer rather than recovery of the true geometry: a single view still admits many 3D explanations (Table 5).

Readout controls. All counterfactual support operators are evaluated on the same trained state. The current incidence condition uses p_{iv} from the evaluating camera. The frame mean condition replaces it with the mean across views for that primitive. The shuffled condition preserves the empirical incidence distribution while breaking the correspondence between primitive and observer. The support law, bounds, flatness term, and learned κ remain unchanged. Explicit state rasterization matches the model renderer exactly, with maximum absolute pixel difference equal to zero. The geometric state before projection support is also identical across cameras. Bootstrap intervals use 200 resamples over Gaussians.

Observed attribution drift. Table 9 reports results for all five scenes. With projection support disabled, $\widehat{\beta}$ is positive in every scene and ranges from 0.975 to 1.333. As a primitive becomes more grazing, its relative image response shifts toward thickness even though the 3D state remains fixed. The actual L1 loss exhibits the same trend. Across the five scenes, the thickness contribution is below 1% for $p < 0.1$ and increases to 63 to 79% for $p > 0.9$. This behavior is the trained model counterpart of the first and second order imbalance derived in Section A.

Using current incidence reduces the within Gaussian slope in all five scenes. The removed fraction ranges from 32% to 65% and averages 48%. Frame mean incidence and shuffled incidence leave the slope close to the support disabled value. This contrast identifies observer correspondence as the source of the calibration effect.

C.2 Reoptimization with current incidence

The preceding analysis compares alternative support rules on a frozen trained state. We next train the current incidence and frame mean incidence variants independently. The comparison uses all 300 held out cam00 frames of *Cook Spinach*. Spatial strata use fixed masks derived from support disabled reference geometry. Dynamic coverage comes from the backbone with support disabled. Ground truth Sobel magnitude defines boundary pixels. Flatness confidence selects the reliable half of these pixels. Reference geometry partitions them into incidence quartiles for each frame. The frame mean

Table 9: **Within Gaussian attribution slope on five Neu3D checkpoints.** All measurements use frame 150 and the same 20 training cameras. We fit the slope of $\log R = \log(\|J_s\|/\|J_\mu\|)$ with respect to $\log p$ and report bootstrap 95% intervals over Gaussians. Every support readout uses the same frozen trained state. The removed fraction measures how much of the support disabled slope is eliminated by current observer incidence.

		support readout and incidence source					
Scene	κ	disabled	current	frame mean	shuffled	removed fraction	Gaussians
Coffee Martini	0.80	1.324 [1.312, 1.334]	0.461 [0.447, 0.474]	1.336	1.338	0.65	35,545
Cook Spinach	0.85	0.975 [0.957, 0.993]	0.517 [0.501, 0.536]	0.990	0.992	0.47	32,051
Cut Roasted Beef	0.88	0.976 [0.957, 0.992]	0.441 [0.423, 0.454]	0.997	0.994	0.55	38,108
Flame Steak	0.65	1.213 [1.199, 1.232]	0.829 [0.817, 0.844]	1.230	1.230	0.32	31,046
Sear Steak	0.74	1.333 [1.319, 1.347]	0.785 [0.772, 0.797]	1.349	1.349	0.41	27,420

control retains the same late support operator and the same q while removing correspondence to the current observer.

Across the full image, current incidence reduces MAE by 0.623%, improves PSNR by 0.020 dB, and yields lower MAE on 96.3% of frames. The boundary analysis localizes the effect beyond the lowest incidence regime. Q1 changes by 0.006%. Q2 through Q4 jointly reduce MAE by 0.389% and improve 74.7% of frames.

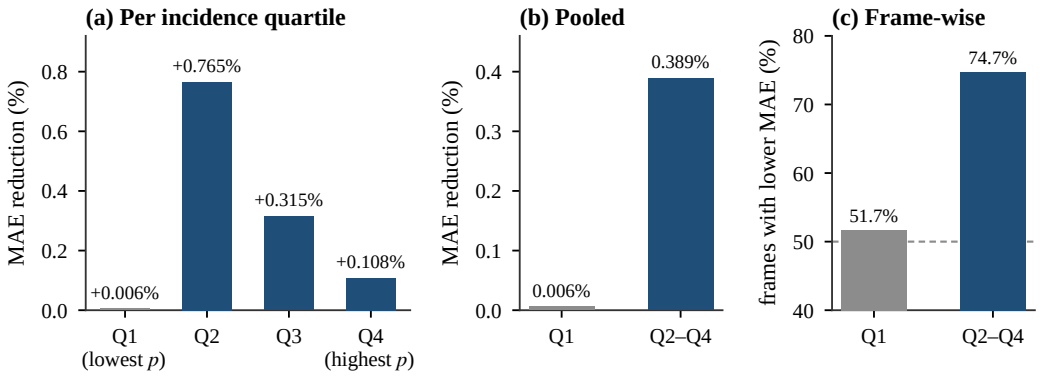

Figure 9: **Reconstruction test of current incidence.** The left panel reports the MAE reduction of current incidence relative to frame mean incidence across four incidence quartiles. The center panel shows that Q1 is approximately neutral while Q2 through Q4 are positive in aggregate. The right panel reports the same comparison across frames. The masks use ground truth edges and support disabled reference geometry.

Table 10: **Current incidence compared with frame mean incidence after independent training.** The evaluation uses *Cook Spinach* and all 300 held out cam00 frames. Incidence quartiles are defined for each frame within reliable high q ground truth boundary pixels using reference geometry from the model with projection support disabled.

Region	MAE reduction ↑	frames with lower MAE ↑
full image	0.623%	96.3%
Q1, lowest p	0.006%	51.7%
Q2	0.765%	81.0%
Q3	0.315%	69.0%
Q4, highest p	0.108%	55.3%
Q2 to Q4 pooled	**0.389%**	**74.7%**

C.3 LIMITS OF PERSISTENT PRIMITIVE INCIDENCE

A natural alternative assigns each Gaussian a persistent scalar that approximates incidence. Least squares analysis shows the intrinsic limitation of this choice. Let p_i^v denote current incidence for primitive i under observation v. Among all constants a_i that do not depend on the observation, the optimal value is the primitive mean,

$$a_i^* = \arg\min_{a_i} \mathbb{E}_v[(p_i^v - a_i)^2] = \bar{p}_i = \mathbb{E}_v[p_i^v], \tag{41}$$

and its irreducible error is

$$\min_{a_i} \mathbb{E}_{i,v}[(p_i^v - a_i)^2] = \mathbb{E}_i[\mathrm{Var}_v(p_i^v)]. \tag{42}$$

Equivalently, the law of total variance gives

$$\mathrm{Var}(p) = \underbrace{\mathrm{Var}_i(\bar{p}_i)}_{\text{persistent component}} + \underbrace{\mathbb{E}_i[\mathrm{Var}_v(p_i^v)]}_{\text{observation dependent component}}. \tag{43}$$

A persistent scalar can represent only the first term.

Fixed state across cameras. Across the five Neu3D scenes and the 20 cameras used for attribution, 13.7% of the absolute variance of p remains within primitives across cameras after the best primitive-wise constant is removed. The median standard deviation across cameras within a primitive is 0.042. The complementary 86.3% reflects stable differences between primitives and is removed by the centering in Eq. 40. The attribution slope is therefore determined by the observer-dependent component that a persistent scalar cannot represent.

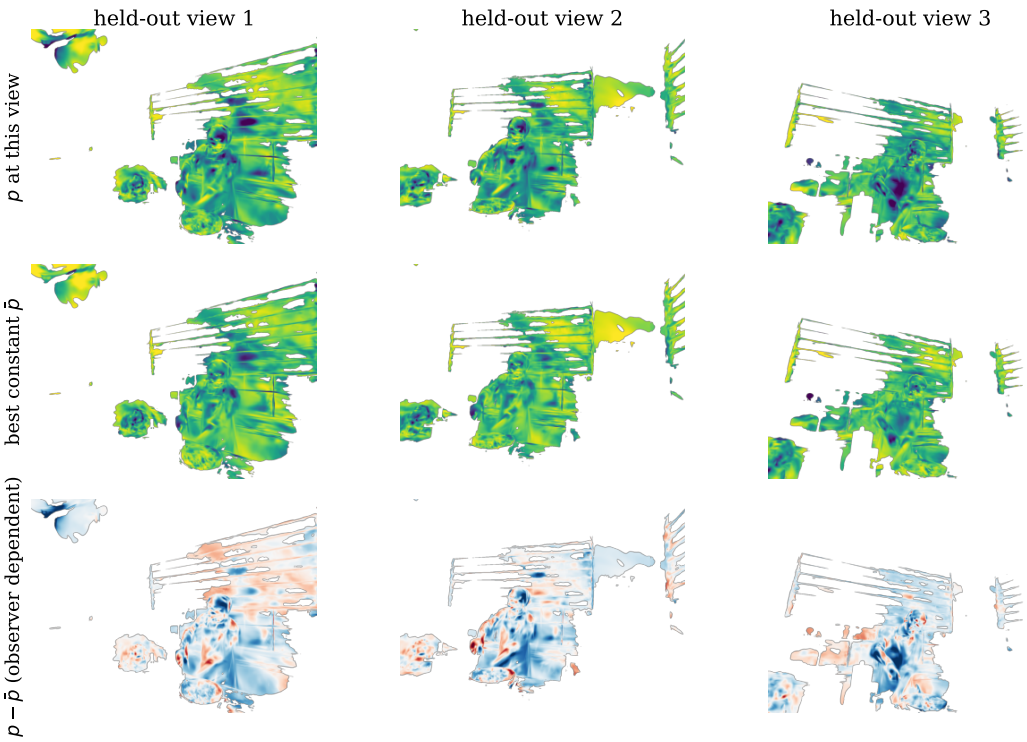

Figure 10: **Observer dependent incidence at a fixed geometric state.** For one *Cook Spinach* state, the rows show current p, the optimal primitive constant $\bar{p}$, and the residual $p - \bar{p}$ under three held out views. In this pixel weighted visualization, 26.2% of the variance remains after the best primitive-wise constant is removed. This residual varies with the observer and cannot be represented by a persistent scalar.

The two percentages use different populations. The 13.7% statistic is primitive weighted over all 20 cameras, whereas the 26.2% visualization is pixel weighted over the three held out views shown in Fig. 10.

Variation along a monocular trajectory. When both the camera and the deformed state vary, the observation dependent component becomes substantially larger. For each D-NeRF sequence, we use the converged and index aligned primitive set, iterate over the training frames, and compute current p_i^v without optimization or gradients. Table 11 shows that the persistent component accounts for only 2.4% to 12.2% of the total variance, with a mean of 5.2% across the seven scenes. This trajectory statistic captures the joint camera and scene variation encountered during monocular training.

The fixed state and trajectory measurements characterize different sources of variation. The fixed state experiment isolates observer variation at constant geometry. The D-NeRF trajectory statistic measures the joint variation encountered as camera and deformed state evolve together.

Summary of trained model evidence. At fixed geometry, the same Gaussian exhibits systematic changes in relative position and thickness response as the observer changes. Current observer incidence reduces this variation, while frame mean and shuffled incidence do not. Independent training shows the same advantage in reconstruction, and the variance decomposition proves that a persistent primitive scalar cannot represent the component that varies across observers. These results connect the controlled projection mechanism to trained 4DGS models.

Table 11: **Incidence variance along D-NeRF trajectories.** $\mathrm{sd}_{\mathrm{prim}}$ measures the spread of primitive means across primitives, and $\mathrm{sd}_{\mathrm{frame}}$ measures the spread across frames within each primitive. The persistent fraction is the first term of Eq. 43 divided by the total variance.

Sequence	$\mathrm{sd}_{\mathrm{prim}}$	$\mathrm{sd}_{\mathrm{frame}}$	persistent fraction
Bouncing Balls	0.041	0.229	3.1%
Hell Warrior	0.079	0.212	12.2%
Hook	0.037	0.219	2.8%
Jumping Jacks	0.036	0.231	2.4%
Mutant	0.044	0.224	3.7%
Standup	0.052	0.218	5.4%
T-Rex	0.061	0.226	6.8%
mean	0.050	0.223	5.2%

D CAMERA AND TIME CORRELATION IN TEMPORAL GEOMETRY

Section C shows that geometric attribution varies with the observer even at a fixed scene state. Monocular capture introduces a second identifiability problem. Training time can determine the training camera. A temporal field can then encode camera dependent fitting through time with no explicit camera input. We formalize this ambiguity with a static scene intervention. We remove the correspondence with simultaneous multiview observations. We also identify the geometric storage channels and evaluate the camera placement used by MoProGS.

D.1 IDENTIFIABILITY UNDER MONOCULAR CAPTURE

For a monocular training sequence, the objective can be written as

$$\mathcal{L}_{\mathrm{mono}}(\psi) = \sum_{t} \ell\big(\mathcal{R}(D_\psi(\theta^0, t), c_t), I_t\big)\,, \tag{44}$$

where D_ψ receives the canonical state and time. If the training data contain one camera for each time, then the observed support satisfies $c_t = h(t)$. Any nuisance correction indexed by the camera, $\Delta(c_t)$, can therefore be represented on that support as

$$\widetilde{\Delta}(t) = \Delta(h(t)). \tag{45}$$

The monocular objective evaluates the observed pairs (t, c_t). On this support, physical time and camera indexed corrections share the same temporal key. The training support therefore creates a camera and time ambiguity in temporal geometry.

With simultaneous observations from multiple cameras, the objective becomes

$$\mathcal{L}_{\mathrm{mv}}(\psi) = \sum_{t} \sum_{m=1}^{M_t} \ell\big(\mathcal{R}(D_\psi(\theta^0, t), c_{t,m}), I_{t,m}\big)\,. \tag{46}$$

The state $D_\psi(\theta^0, t)$ must now explain several cameras at the same time. Time can no longer select an independent state for each camera. The following experiments instantiate this distinction while keeping either the physical scene or the optimization budget fixed.

D.2 STATIC PSEUDO TIME INTERVENTION

We first remove physical motion from the problem. The experiment uses frame 0 of *Cook Spinach* and camera slots $\{0, 3, 5, 8, 11, 14, 16, 19\}$. All eight images therefore depict the same physical 3D state. Every configuration starts from the same high fidelity canonical Gaussian snapshot. Geometry and appearance remain frozen except for bounded temporal updates to center, scale, and rotation on a shared dynamic mask. These updates are optimized for 1200 iterations. The deformation field receives position and time.

The configurations share the data, ray budget, network, parameter count, optimizer, mask, schedule, and initialization. Their temporal labels differ. The shared time configuration assigns $t = 0$ to all cameras. The unique time configuration assigns eight distinct labels between 0 and 0.9967 in camera

order. An output adapter initialized exactly at zero makes every initial geometric update zero while retaining a nonzero gradient at the first optimization step. The coarse and fine initial renders are identical with maximum pixel difference zero, and the maximum initial state deviation is also zero. A photometric backward pass produces an adapter gradient norm of 5.744×10^{-3}. When pseudo time is held fixed, changing only the camera leaves center, raw scale, and raw rotation unchanged before rasterization. Any state variation learned across pseudo times must therefore arise from the training correspondence between temporal label and camera.

Cartesian recombination. Let θ_j denote the state produced by temporal label j. We render every combination in the 8×8 Cartesian product $\mathcal{R}(\theta_j, c_k)$. The eight diagonal combinations are observed during training, while the remaining 56 combine a learned state with a different camera. The physical scene is static. A camera independent temporal state should transfer across all cameras. A performance gap between observed and recombined pairs measures training camera information stored through the temporal label.

Table 12: **Recombination in the static pseudo time intervention.** Recombined PSNR averages the 56 combinations of camera and temporal state that are absent from training. The shared time configuration contains a single state and therefore has no recombined category.

Configuration	initial	assigned	recombined	gap
shared time	33.080	32.980	–	–
unique time	33.080	34.117	31.214	**2.903**

The unique time configuration improves its observed camera and time pairs by 1.037 dB relative to initialization, while performance on unseen recombinations decreases by 1.866 dB. The same physically static scene also develops substantial variation in its temporal geometric state, as reported in Table 13. The recombination failure is therefore accompanied by geometric variation that is correlated with camera identity.

Table 13: **Temporal geometric variation in the static scene intervention.** Center variation is also normalized by the median Gaussian scale. The physical scene has zero temporal variation by construction.

Configuration	center SD	center / median scale	log scale SD	rotation SD
shared time	0	0	0	0
unique time	0.0669	**0.6356**	0.1547	0.0759

Bound utilization remains low across the trained temporal labels. The fraction above 95% of the center bound is 0.58%. The corresponding fractions are 7.30% for log scale and 0.34% for rotation.

Shortcut under random temporal labels. A second experiment starts from a common snapshot with an additional controlled residual of 27.8575 dB. It compares shared labels, smoothly ordered unique labels, and a random permutation of the same one to one labels. The random assignment breaks the smooth relation between label order and camera trajectory while preserving a unique label for each camera. Table 14 shows a 4.624 dB gap for the random assignment and a 4.320 dB gap for the smooth assignment. A unique temporal key correlated with camera identity is sufficient for the shortcut.

Table 14: **Uniqueness control for temporal labels.** A random one to one assignment between cameras and temporal labels retains the separation between observed and recombined pairs.

Configuration	initial	assigned	recombined	gap
shared time	27.858	32.980	–	–
smooth unique time	27.858	33.670	29.350	4.320
random unique time	27.858	**34.001**	**29.376**	**4.624**

D.3 SIMULTANEOUS MULTIVIEW SUPERVISION BREAKS THE CORRESPONDENCE

The static intervention shows that a camera blind temporal field can encode camera dependent corrections through time. We next test the same ambiguity in dynamic training. The camera and time correspondence is the changed factor. Using *Cook Spinach*, we select camera slots 0 and 10 and ten physical times $\{0, 33, 66, 100, 133, 166, 199, 233, 266, 299\}$. Both configurations contain 20 training samples and share the same initialization, architecture, optimizer, time set, camera set, image and ray budget, and total optimization budget.

In the single trajectory configuration, each time is associated with only one of the two cameras, alternating across the selected times. Each sample is duplicated through two mask phases so that the total count remains 20. Camera is therefore a deterministic function of time on the observed training support. In the simultaneous multiview configuration, both cameras are observed at every selected time, and one temporal state must explain both views. These two configurations are the finite sample counterparts of Eqs. 44 and 46.

Table 15: **Matched ray intervention on *Cook Spinach*.** In the single trajectory configuration, the unseen composition is the second camera at each training time. In the simultaneous multiview configuration, both cameras are observed at every selected time.

Configuration	observed pairs	unseen combinations	complete grid over both cameras	held out cam00
single trajectory	**33.479**	26.727	30.103	23.082
simultaneous multiview	33.356	–	**33.356**	23.205

The single trajectory configuration exhibits a composition gap of 6.7516 dB. Providing both cameras at each selected time raises reconstruction over the full two camera grid by 3.2527 dB. PSNR on the originally observed pairs changes by 0.123 dB. The separate held out cam00 evaluation changes by the same amount. The improvement is concentrated on camera and time compositions absent from the monocular training support. The matched sample count, camera set, ray budget, and optimization budget isolate the camera and time correspondence.

D.4 NORMAL-AXIS STORAGE IN THE STATIC INTERVENTION

The pseudo time experiment assigns a unique temporal label to each camera while the physical scene remains fixed. We use the matched pilot residual pair to examine the two geometric channels that correspond directly to the center and support analysis: normal displacement and normal log thickness. Both runs start from the same prepared snapshot and use the same residual regime; they differ only by the MoProGS projection support used during optimization.

Table 16 shows a selective change in the support channel. Median normal displacement remains 4.4 standard deviations along the thin axis, whereas median normal log thickness decreases from 0.0997 to 0.0500. At the primitive level, MoProGS reduces thickness variation for 63.6% of primitives, with a median ratio of 0.63 relative to the baseline. Positional variation remains nearly unchanged, with a median ratio of 1.01.

These measurements localize the effect to the normal support channel predicted by the projection analysis. When a temporal key absorbs observer specific fitting, late projection support reduces the amount stored as normal thickness. Normal displacement remains essentially unchanged.

Table 16: **Normal-axis storage in the matched static intervention.** Both runs use the same prepared snapshot and residual regime. Normal displacement is measured in standard deviations along the thin axis; thickness is reported as a log ratio. The comparison is scoped to the two channels analyzed by the position/support model.

Configuration	normal displacement	normal log thickness
baseline	4.4	0.0997
MoProGS	4.4	**0.0500**

D.5 CAMERA DEPENDENCE IS INTRODUCED AFTER DEFORMATION

The preceding experiments characterize the indirect route created by camera and time correlation. We next compare direct camera conditioning in deformation with camera conditioning in projected support. The placement experiment holds the physical scene state fixed.

We use frame 150, camera slots $\{0, 3, 5, 8, 11, 14, 16, 19\}$, and 42,991 dynamic Gaussians. The early conditioning configuration feeds a camera encoding to the deformation field. The late conditioning configuration first forms a camera blind geometric state. It then applies the current view in projection support. Under early conditioning, changing the camera changes center, log scale, and rotation before support. Under late conditioning, these quantities are invariant across cameras before support. The rendered log scale varies after projection support. Table 17 reports the corresponding standard deviations.

Table 17: **Camera sweep at a fixed scene state.** The table reports the standard deviation across cameras, averaged over dynamic Gaussians. Quantities before support belong to the temporal geometric state. Rendered scale is measured after projection support and is the scale passed to the rasterizer.

Configuration	center before support	log scale before support	rotation before support	rendered log scale
early camera conditioning	0.01072	0.06343	0.04632	0.06343
late projected support	**0**	**0**	**0**	0.02143

Late placement yields zero cross camera variation in the pre support state. Observer variation appears in rendered support. The pseudo time and simultaneous multiview experiments characterize the separate camera and time ambiguity. Together, these results motivate the factorization in MoProGS. Temporal deformation remains camera blind. Observer dependence enters at projection and compensates the incidence dependent response derived in Section A.

E VALIDATION OF MOPROGS DESIGN COMPONENTS

Sections A through D establish the mechanism underlying MoProGS. Projection produces an imbalance between position and normal support. The current deformed primitive and current camera define the routing cue. Camera dependence enters projected support after the scene state is formed. This section evaluates the corresponding implementation choices under matched training conditions. The full model retains separate response adapters for center and covariance. We evaluate projection support and routing with matched controls.

E.1 PROJECTION ROUTING ON D-NERF

The full operator computes $\gamma = qp$ from the current deformed state, uses $\widetilde{\gamma} = \mathrm{sg}(\gamma)$ as a detached routing variable, and applies the resulting normal support factor immediately before rasterization. The D-NeRF component study averages over all seven scenes and changes one routing choice at a time while keeping the remaining model and optimization settings fixed.

Table 18: **Projection routing ablations on D-NeRF.** Each control changes one element of the routing rule while preserving late projection support and all remaining training settings.

Variant	Changed element	PSNR↑	SSIM↑	LPIPS↓	ΔPSNR
full MoProGS	current q, p with detached routing	43.13	.995	.005	–
without flatness confidence q	remove the q factor from $\gamma = qp$	43.01	.994	.007	−0.12
without detached routing	backpropagate through the routing cue	42.96	.994	.007	−0.17
routing from canonical state	form the routing cue from canonical geometry	43.03	.994	.007	−0.10

The three controls isolate distinct assumptions in the routing design. The flatness confidence q determines whether the smallest axis provides a reliable local normal cue. Removing q allows incidence alone to activate support correction on primitives whose smallest axis may not represent a surface normal, and PSNR decreases by 0.12 dB. Detachment prevents p and q from becoming additional optimization paths. Allowing gradients through the routing cue produces the largest decrease in this control set, 0.17 dB. Current state routing evaluates incidence and flatness on the same deformed primitive that is subsequently rasterized. Replacing it with canonical state routing decreases PSNR by 0.10 dB. SSIM and LPIPS change in the same direction in all three comparisons. Together, these matched controls support the routing design and identify the role of each routing choice within the reported configuration.

E.2 Late projection support with synchronized capture

We isolate late projection support with a paired synchronized capture experiment on *Cook Spinach*. Both configurations branch from the same prepared snapshot. They share the data, initialization, optimizer, ray budget, response architecture, and training schedule of $13{,}001$ iterations. Support activation is the changed factor. Synchronized capture also removes monocular camera and time correspondence from this comparison.

Table 19: **Isolation of late projection support under synchronized capture.** The paired runs share the response architecture, initial snapshot, data, optimizer, ray budget, and training schedule. They differ in late projection support.

Evaluation set	Variant	PSNR↑	SSIM↑	L1↓
held out cam00	without late support	31.569	.94824	.01717
held out cam00	with late support	**31.774**	**.94855**	**.01663**
observed views	without late support	34.807	.96197	.01033
observed views	with late support	**34.864**	**.96200**	**.01026**

Late projection support increases PSNR on held out cam00 by 0.20454 dB and reduces L1 from 0.01717 to 0.01663. This is a 3.1% reduction in L1. On the observed views, PSNR increases by 0.05636 dB and L1 decreases from 0.01033 to 0.01026. The larger gain on the held out camera matches the cross view role of the projection support operator.

Figure 11 shows held out frame 217. Enabling late projection support changes PSNR from 31.386 to 31.779 dB on this frame. The gain is 0.393 dB. The visual difference is concentrated at a high contrast silhouette boundary. Projected center displacement and normal support have strongly overlapping image effects in this region.

Implications of the component controls. The D-NeRF routing ablations and the synchronized capture experiment isolate complementary parts of the projection design. The routing experiments establish the benefit of a reliable normal cue, current state evaluation, and detached routing. The synchronized experiment isolates the late support operator under a shared response architecture. The support gain persists under synchronized capture. These controls validate the projection component of the factorization.

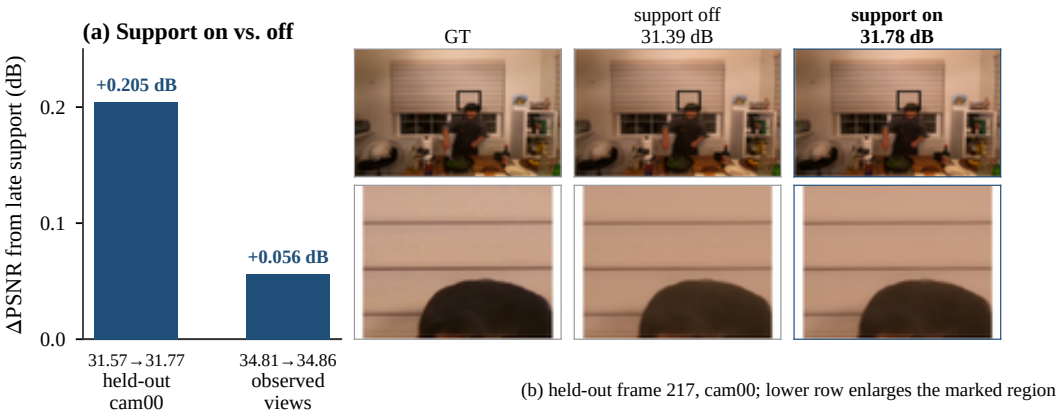

Figure 11: **Visual effect of late projection support under synchronized capture.** The left panel reports PSNR on held out cam00 and the observed cameras with a shared response backbone. The right panel shows held out frame 217 and the marked boundary crop. The lower row enlarges the same crop.

F Extended reconstruction evidence

The preceding sections establish the mechanism through theory, controlled geometry, trained model attribution, and matched component interventions. We now examine its consequences in benchmark reconstruction. We report sequence level coverage, spatial residual localization, and qualitative frame selection. These results connect reconstruction quality to the projection behavior analyzed above.

F.1 Sequence level coverage

The main paper reports every D-NeRF and Neu3D sequence individually, so those tables are not repeated here. On D-NeRF, the PSNR margin over the strongest value reported by a prior method is positive for all seven scenes. The margin ranges from 0.07 dB on Mutant to 0.86 dB on Jumping Jacks, and the mean improvement is 0.38 dB as reported in the main paper. All seven scenes contribute positive PSNR margins. The Neu3D table in the main paper similarly reports all five synchronized scenes.

Table 20 expands the HyperNeRF average into its four sequences. Their arithmetic mean is 25.97 dB and matches the result in the main paper.

Table 20: **Per sequence PSNR of MoProGS on HyperNeRF.** The four sequence values average to 25.97 dB, matching the result reported in the main paper.

Method	3D Printer	Broom	Chicken	Peel Banana	Average
MoProGS	22.72	22.21	29.38	29.57	25.97

F.2 Qualitative coverage of all benchmark scenes

The main paper visualizes Hook, Jumping Jacks, and Hell Warrior from D-NeRF, together with Coffee Martini, Cook Spinach, and Cut Roasted Beef from Neu3D. Figure 12 provides the complementary sequences. It contains Bouncing Balls, Mutant, Standup, and T-Rex for D-NeRF and Flame Steak and Sear Steak for Neu3D. Together, the main and supplementary figures cover every D-NeRF and Neu3D sequence under the same crop protocol.

Together with the per scene table in the main paper, these figures cover the full benchmark scene set under a shared crop protocol. Visible differences are concentrated around local contours, articulated boundaries, and thin structures. We analyze this spatial concentration next.

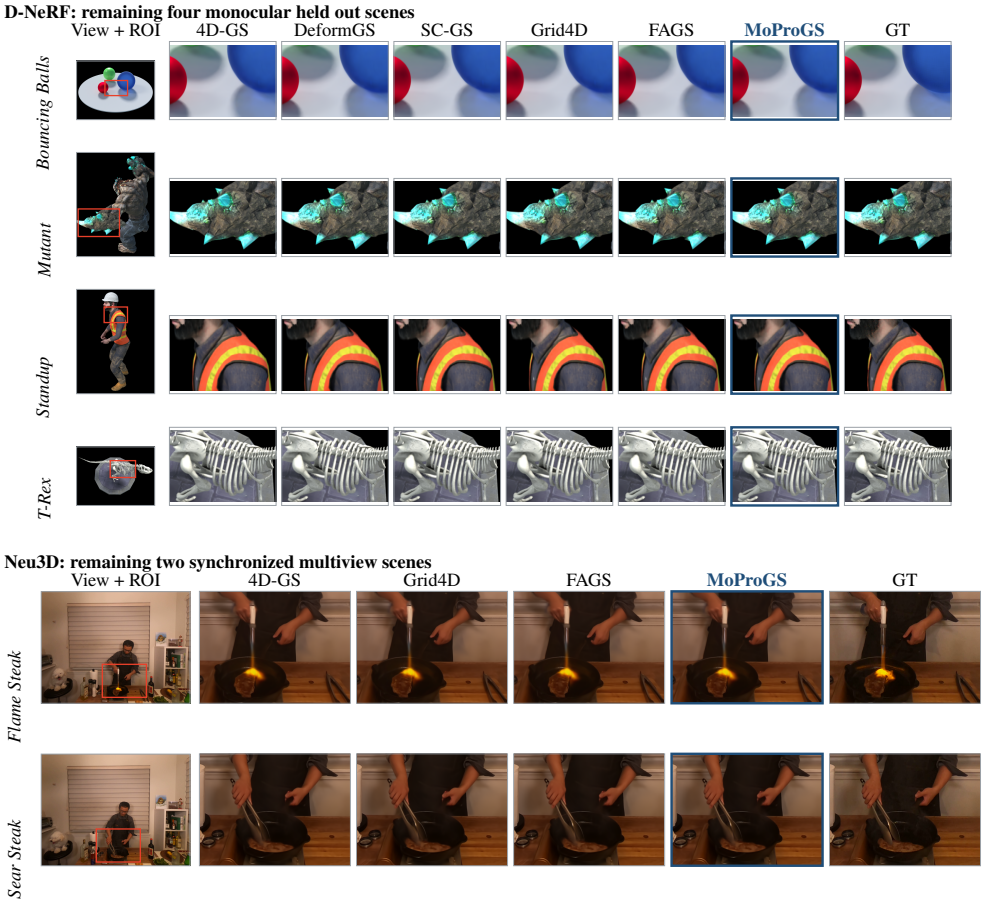

Figure 12: **Qualitative results for the benchmark sequences not visualized in the main paper.** The D-NeRF rows show Bouncing Balls, Mutant, Standup, and T-Rex, which complement Hook, Jumping Jacks, and Hell Warrior in the main paper. The Neu3D rows show Flame Steak and Sear Steak, which complement Coffee Martini, Cook Spinach, and Cut Roasted Beef. The first image in each row locates the predefined crop. Every method uses the same held out frame, camera, and crop.

F.3 Spatial localization of reconstruction residuals

The projection mechanism is most visible near boundaries, where the thin Gaussian axis contributes appreciable image support. Full image metrics do not reveal the spatial concentration of the residual. Across all seven D-NeRF scenes, the worst 5% of pixels account for 55% of the total absolute residual. Figure 13 summarizes this concentration.

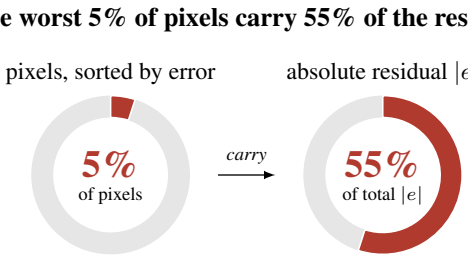

Figure 13: **Spatial concentration of reconstruction residuals on D-NeRF.** The worst 5% of pixels account for 55% of the total absolute residual across all seven scenes.

Figure 14 provides residual maps for the fixed qualitative crop of every D-NeRF sequence. We compute $|\text{render} - \text{GT}|$ directly from each RGB crop pair and use one linear color range across all methods and scenes. The first three rows correspond to the D-NeRF examples in the main paper. The remaining four rows correspond to Figure 12. The residual visualization is aligned with the full qualitative scene set.

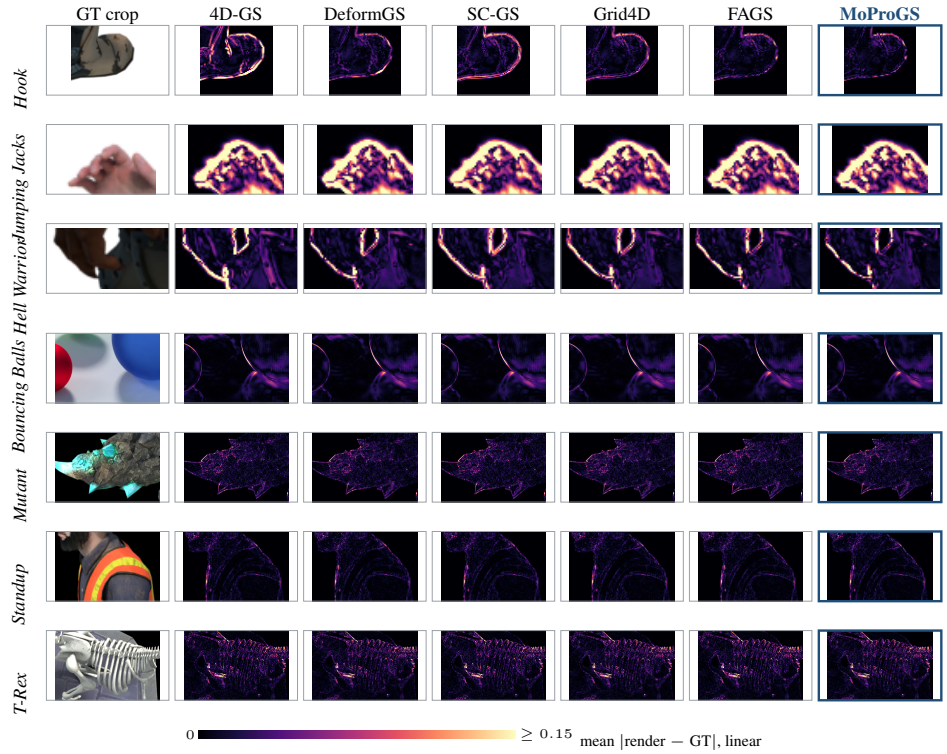

Figure 14: **Residual maps on the fixed qualitative crops for all seven D-NeRF scenes.** The first column shows the ground truth crop, and the remaining columns show mean absolute RGB error on the same region. The first three rows correspond to the D-NeRF examples in the main paper. The remaining four rows correspond to Fig. 12. Every map uses the same linear color scale with maximum value 0.15. Residual energy is concentrated around silhouettes, articulated boundaries, and thin structures, which are the spatial regions emphasized by the position and support analysis.

The residual maps provide spatial evidence that complements the controlled mechanistic experiments. The derivation predicts observer dependent competition between center position and normal support. The trained model measurements identify the corresponding change in attribution. The benchmark residuals concentrate in regions where these projected effects overlap most strongly. Sections B through E isolate the mechanism through controlled experiments.

F.4 Qualitative frame selection protocol

Figure 15 places each selected D-NeRF frame within the distribution of the 20 held out frames from the same scene. The selected ranks are $5, 12, 9, 7, 9, 10, 11$ out of 20 in benchmark scene order. Four selected frames fall below the mean PSNR of their scene. The selections span the held out frame distributions across scenes.

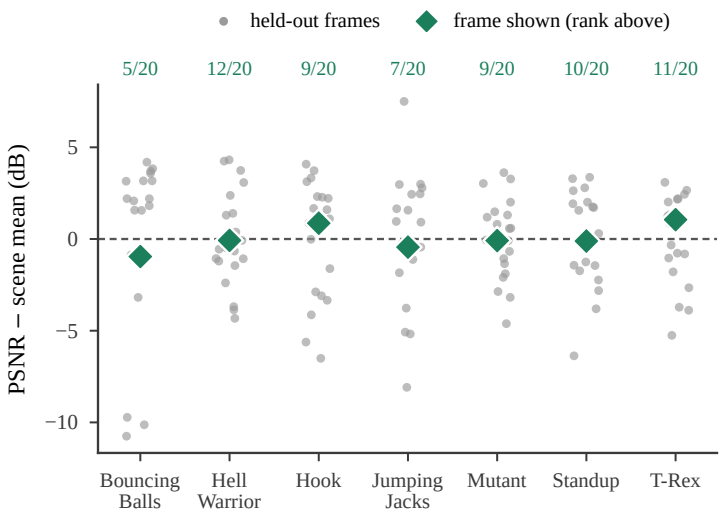

Figure 15: **D-NeRF qualitative frame selection.** Gray points show all 20 held out frames for each scene as PSNR relative to the scene mean. The green diamond indicates the frame used by the qualitative and residual figures, with its rank shown above.

G Implementation and reproducibility

We report the implementation details and experimental protocols required to reproduce the projection support and the controlled comparisons above.

G.1 Projection support

The normal factor is

$$\xi_n = \mathrm{clip}(1 - \kappa\ \mathrm{sg}(q)\,\mathrm{sg}(p), \xi_{\min}, 1), \qquad \xi_{t_1} = \xi_{t_2} = 1. \tag{47}$$

We use $\xi_{\min} = 0.1$ and $\kappa = \kappa_{\max}\sigma(\eta)$ with $\kappa_{\max} = 4$, initialized near zero. The factor is computed from deformed center, rotation, and scale and applied to the activated scale entering rasterization. Stop gradient is applied only to the routing copies of p and q; ordinary rendering remains differentiable with respect to geometry.

The operator is continuous in the deformed state even when the thin axis changes. When the scales evolve continuously, $j^* = \arg\min_j s_j$ can change only between the smallest and the middle axis, at a crossing $s_{j^*} = s_{\mathrm{mid}}$. At such a crossing, $q = 1 - s_{j^*}/(s_{\mathrm{mid}} + \varepsilon) = \varepsilon/(s_{\mathrm{mid}} + \varepsilon) \approx 0$, so $\xi_n \approx 1$ and the covariance modification $(\xi_n^2 - 1)s_n^2\mathbf{n}\mathbf{n}^\top$ vanishes up to $O(\kappa\varepsilon/s_{\mathrm{mid}})$ on both sides of the switch. The clip in Eq. 47 is also continuous. The operator is therefore continuous but not differentiable at axis crossings, and because p and q are detached, no gradient flows through the routing cue at these points. Without q, the factor $1 - \kappa p$ stays away from 1 at a crossing and the contracted axis jumps discontinuously; this corresponds to the variant without flatness confidence in Table 18.

G.2 Reliability of the local normal cue

On reliable D-NeRF primitives, the mean absolute cosine between a primitive's smallest axis and those of its eight nearest neighbors is 0.648. Shuffling the normals over the same point set reduces this value to 0.502, which is close to the random direction expectation of $1/2$. The incidence range is populated: $\mathrm{frac}(p < 0.5) = 0.129$ and $p_{10} = 0.444$.

G.3 CAMERA BLIND RESPONSE ADAPTERS

Independent adapters for center and covariance use the native hidden dimensionality of each deformation configuration. They are initialized as exact identities. Initialization preserves the global random number stream. The first forward pass matches the model before response splitting. Opacity and color bypass these adapters. Both geometric branches are predicted from a shared temporal encoder. Besides decoupling the two branch responses, the adapters add a small number of parameters. Projection support then calibrates their rendered response using current geometry. Section E.2 isolates the contribution of the support operator while keeping the response architecture fixed.

G.4 DATASET CONFIGURATIONS AND OPTIMIZATION

Across all benchmark configurations, MoProGS computes detached incidence after temporal deformation and uses it to condition the normal support applied before rasterization. The full model also retains camera blind response adapters for center and covariance. D-NeRF is optimized for 80k iterations on one RTX 3090. The remaining benchmarks follow their released optimization schedules. The image objective combines L1 and D-SSIM.

G.5 RENDERING SPEED AND MEMORY

We measure inference on Neu3D over 200 frames of the held-out cam00 view at 1352×1014. Timing covers the complete fine-stage rendering pipeline, including temporal deformation, projection support, and rasterization. The model contains about 329.8K Gaussians. Table 21 shows that MoProGS renders at 74.47 FPS with a 95th percentile frame time of 14.64 ms. Peak allocated GPU memory during rendering is 484.2 MiB, of which 374.1 MiB is allocated on top of the memory already resident before rendering.

Table 21: **Rendering speed and memory of MoProGS** over 200 held-out Neu3D frames at 1352×1014 with the complete fine-stage pipeline.

Quantity	Value
Gaussians	329.8K
Rendering speed	74.47 FPS
Frame time, mean / median / P95	13.43 / 13.01 / 14.64 ms
Peak allocated memory	484.2 MiB
Peak reserved memory	624.0 MiB
Additional peak over resident memory	374.1 MiB

G.6 STATIC INTERVENTION IMPLEMENTATION

The static intervention freezes all canonical parameters and enables only temporal updates to geometry. Full output adapters initialized exactly at zero preserve the identity mapping while maintaining nonzero gradients at the first optimization step. Each run checks initialization, gradient propagation, parameter updates, finite values, camera invariance before rasterization, and bound utilization. The high fidelity static intervention and the temporal label uniqueness control use the protocols described in Sec. D.2. The auxiliary control uses the center perturbation described in Sec. D.2 to set the residual regime.

H LIMITATIONS

The projection analysis is local and assumes surface-like anisotropic Gaussians whose thinnest axis approximates a surface normal. Volumetric or semi-transparent content, such as the flame in *Flame Steak*, may violate this assumption; the flatness confidence q suppresses the operator only where such content is represented by near-isotropic primitives. A single global κ removes only part of the incidence dependence in trained models, since the slope reduction in Eq. 31 differs across primitives with different q and p; per-primitive calibration is a natural extension. MoProGS reduces the view

dependence of geometric error attribution but does not make 3D geometry identifiable from a single view (Table 5).