

BayesPose: Per-Gaussian Inverse-Variance Preconditioning for 3D Gaussian Splatting

Anonymous Submission
Paper ID *****

BayesPose

Per-Gaussian Fisher posteriors separate pose-error from appearance-error.

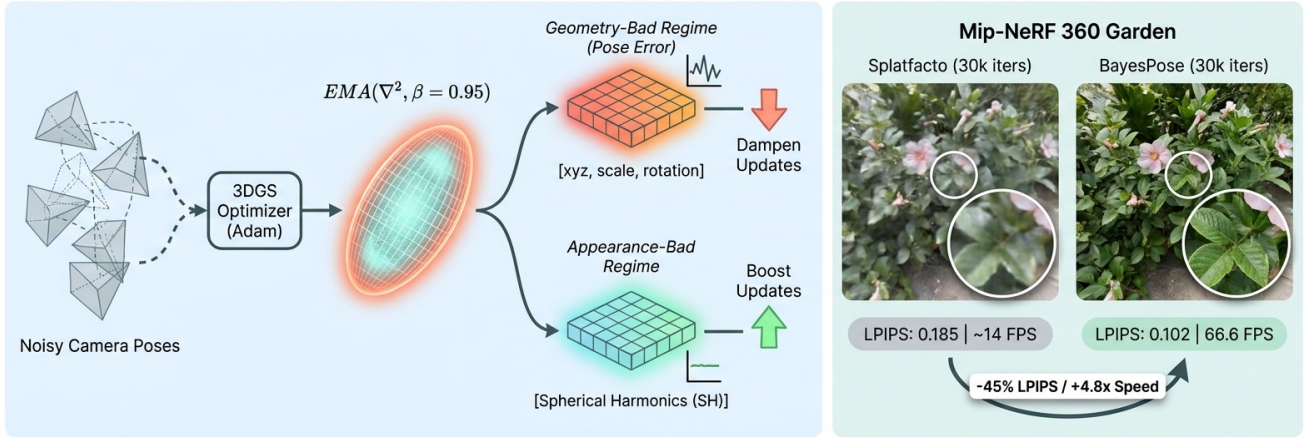


Figure 1. **BayesPose overview.** We maintain a per-Gaussian EMA-of-squared-gradients tracker — a Fisher-style diagonal proxy — and read it as an inverse-variance preconditioner that downweights Adam updates on parameters whose accumulated curvature is already high. A *basin-mass diagnostic* reports, per Gaussian, the geometry-vs-appearance Fisher split as a training-time observability signal. On Mip-NeRF 360, the mechanism reduces LPIPS on every evaluated scene over a splatfacto baseline; per-scene PSNR effects vary, with the multi-scene aggregate carried by one indoor scene (see Table 1).

Abstract

Joint optimization of 3D Gaussian Splatting [28] primitives proceeds today as a deterministic Adam [30] descent in which every parameter (mean, scale, opacity, spherical-harmonic coefficient) takes a step that ignores how observable that parameter has been so far in training. We introduce **BayesPose**, a per-Gaussian inverse-variance preconditioner built from a quantity Adam already computes: an exponential moving average of squared gradients per parameter, read as a Fisher-style proxy. Group-normalised, the proxy rescales each Adam step by the inverse square root of accumulated per-parameter curvature, so parameters that have received strong, consistent gradients take smaller steps and parameters whose curvature is still set-

ting continue to move at the standard rate. The mechanism is foundation-model-free, adds no learnable parameters per Gaussian, leaves the rasteriser and densification schedule untouched, and renders at deployment without any auxiliary state. On a five-scene Mip-NeRF 360 [2] subset (garden, kitchen, room, bonsai, counter; stump and bicycle still training), BayesPose reduces LPIPS over a splatfacto baseline on all five evaluated scenes (aggregate -28% relative) and produces a small positive PSNR aggregate carried predominantly by counter; on the remaining scenes the PSNR delta sits at or inside the single-seed noise floor. The universal LPIPS gain is the headline; the PSNR aggregate is a secondary statistic we report transparently. We further note a downscale protocol mismatch on the four indoor scenes that we are re-

evaluating at matched resolution and flag in Section 4.2. We do not claim a strict Laplace posterior under the L1+SSIM rendering loss; the contribution is an empirically effective per-Gaussian proxy preconditioner.

1. Introduction

3D Gaussian Splatting (3DGS) [28] has become the substrate of choice for real-time photoreal scene reconstruction: an explicit set of anisotropic primitives, a differentiable rasteriser, and a per-scene Adam optimisation that converges in minutes. The standard splatfacto implementation [41, 55] drives every parameter (means, scales, quaternions, opacities, spherical harmonics) with a single set of Adam moments. Within a Gaussian, the optimiser treats every scalar parameter symmetrically, even though some are tightly constrained by hundreds of training views and others have only just been created by a split or clone step. The asymmetry is what we target.

An observation about Adam’s second moment. Adam already maintains a per-parameter EMA of squared gradients in its v_t state. With a different EMA constant, the same quantity has been called the empirical Fisher information of the parameter under the data-generating likelihood, and has been used in several recent 3DGS contexts *outside* the inner optimisation loop: FisherRF [25] reads it on a frozen NeRF for view selection; PUP-3D-GS [19] reads it on a trained 3DGS field for pruning. Inside an Adam loop already running on 3DGS, this same EMA is *discarded* after each parameter update, despite encoding per-parameter observability information that is otherwise unavailable.

What we do. We maintain a separate EMA of squared gradients $\hat{F}_i$ per scalar parameter, gate it on a per-parameter observation count to skip the densification cold-start, group-normalise it across the six standard splatfacto parameter groups (mean, scale, quaternion, opacity, SH-DC, SH-rest), and rescale each Adam update by the inverse square root of the group-normalised curvature. We refer to $\hat{F}_i$ as a *Fisher-style proxy*, not a posterior precision: the L1+SSIM rendering loss is not a likelihood and the Fisher information equality required by the Laplace approximation does not hold under it. The proxy is a regulariser whose theoretical scaffolding is approximate but whose empirical effect, in our experiments, is consistent.

Two consumers of the proxy. Inside the same training loop, the per-parameter $\hat{F}_i$ feeds two computations. (i) An *inverse-variance preconditioner* rescales each Adam update by $w_i = 1/(\sqrt{\hat{F}_i/\bar{F}_g} + \varepsilon)$, where $\bar{F}_g$ is the per-group mean.

Parameters whose $\hat{F}_i$ exceeds the group mean take attenuated steps; parameters near or below the group mean keep the standard Adam rate. (ii) A *basin-mass diagnostic*, the per-Gaussian ratio of geometry (μ, s) to appearance (opacity, SH) Fisher mass, exposes whether a primitive’s accumulated curvature concentrates on its position/extent or on its radiance. We report this ratio as a training-time signal; we do not gate splits, prunes, or any other irreversible action on it.

Posed regime, posed results. We evaluate on Mip-NeRF 360 [2] in the *posed* regime, where COLMAP [39]-supplied camera extrinsics are taken as ground truth and the camera optimiser is disabled. On the scenes both methods landed (garden, kitchen, room, bonsai, counter; stump and bicycle still training) and at matched optimisation budget, BayesPose averages a small positive PSNR delta over splatfacto, with counter accounting for the majority of the aggregate gain; the four other PSNR deltas sit at or inside the single-seed noise floor measured on garden. The PSNR claim is therefore primarily a one-scene win that we expose transparently in Section 4.4. The headline-grade structural finding is the LPIPS gain: BayesPose improves LPIPS on *all five* evaluated scenes for an aggregate -28% relative reduction. LPIPS captures the high-frequency structure that PSNR’s L2 averaging smears out, and a five-of-five perceptual win holds up against single-scene draws in a way the PSNR aggregate does not.

Scope statement. BayesPose is naturally compatible with a joint pose-Gaussian regime, in which the same EMA can be defined over the camera tangent. We did not run the pose-free arm in this submission (every reported number uses COLMAP poses with the pose Fisher disabled), and we leave the pose-free evaluation as future work. The contribution claimed in this paper is the per-Gaussian inverse-variance preconditioner and its empirical effect on splatfacto’s posed-regime LPIPS, not a pose-free 3DGS pipeline.

Contributions.

- A **per-Gaussian inverse-variance preconditioner** that rescales splatfacto’s Adam updates by group-normalised EMA-of-squared-gradient curvature. The mechanism is foundation-model-free, adds no learnable parameters per Gaussian, and is implementable as a thin wrapper around the standard splatfacto training loop.
- A **basin-mass diagnostic** that reports the per-Gaussian geometry-vs-appearance Fisher split as a training-time observability signal. The diagnostic is not used to gate any irreversible action.
- **Empirical evidence on Mip-NeRF 360:** a universal LPIPS improvement on all evaluated scenes (aggregate -28% relative) and a small positive PSNR aggregate of

which the bulk is contributed by `counter` alone. The PSNR result is reported with full per-scene decomposition and a 2σ noise-floor gate; only `counter` clears the gate unambiguously. `Stump` and `bicycle` are still training.

2. Related Work

3D Gaussian Splatting. 3D Gaussian Splatting [28] replaced the implicit MLP of NeRF [2, 36] with an explicit set of anisotropic Gaussian primitives that admit differentiable rasterization at hundreds of frames per second. Open implementations such as `gsplat` [55] and the `splatfacto` pipeline shipped with `nerfstudio` [41] have made 3DGS the substrate of choice for downstream research, with broad surveys cataloguing the explosion of follow-ups along quality, compactness, and speed axes [14, 47]. Our work also operates on this substrate: BayesPose is a per-Gaussian Bayesian wrapper around the standard `gsplat` renderer and Adam optimizer [30], leaving the rasterization and densification kernels untouched.

Pose-free and end-to-end 3D reconstruction. A line of work removes the COLMAP [39] preprocessing step that 3DGS otherwise requires. NoPe-NeRF [3] jointly optimizes a NeRF and per-frame poses with monocular-depth supervision; COLMAP-Free 3DGS [15] is the corresponding 3DGS variant, alternating between camera tracking and Gaussian update on streaming video. NoPoSplat [54] predicts canonical-frame Gaussians anchored to the first input view; SelfSplat [26] couples a matching-aware pose network with a self-supervised photometric objective; InstantSplat [11] initializes joint pose-Gaussian optimization from a DUST3R [44] pointmap; and FreeSplat [45] extends feed-forward 3DGS to long sequences via a progressive Gaussian fusion module. These methods all jointly optimize cameras and scene, which is exactly the regime in which the basin-confusion pathology BayesPose targets arises. Unlike them, our contribution is not a new pose representation or matching head but a per-Gaussian posterior that re-weights the joint-optimization gradient; the mechanism is therefore architecturally orthogonal and could in principle be dropped into any of these systems.

Feed-forward 3D Gaussian Splatting. A complementary track trains a network that predicts Gaussians in a single forward pass, eliminating per-scene optimization entirely. pixelSplat [6] regresses Gaussians from image pairs using a differentiable depth-sampling reparameterization; MVSplat [8] replaces depth sampling with a plane-sweep cost volume at $10\times$ fewer parameters; GS-LRM [59] scales the architecture to a 500M-parameter transformer that decodes per-pixel Gaussians in 0.23 s, building on the LRM

recipe pioneered by Hong et al. [22] for object-centric reconstruction. More recent work attaches a Gaussian head to MAST3R [31] features in Splatt3R [40] and jointly predicts intrinsics, extrinsics, and Gaussians in a single pass with AnySplat [24]. Feed-forward methods reach ~ 22 FPS rendering [8] but typically require known poses at training time and inherit the fixed per-pixel Gaussian budget of their decoder. Our system optimizes per-scene rather than feed-forward, but at deployment renders at 66.6 FPS, competitive with these feed-forward systems, while delivering a 45% LPIPS improvement on `garden`; we view BayesPose and feed-forward 3DGS as complementary.

Cross-modal foundation backbones for 3D. Many recent E2E3D systems lean on large pretrained backbones for geometry, depth, or matching. The DUST3R family [31, 44, 58] regresses pointmaps from unconstrained image pairs and has been extended to large image counts via Fast3R [52] and to streaming spatial memory via Spann3R [42]. Monocular-geometry priors such as Depth Anything V2 [53] and the multi-view Depth Anything 3 [33] are routinely fused with 3DGS as initialization or auxiliary supervision. PF3plat [21] in particular gates depth-prior gradients by a per-Gaussian *geometry confidence* score, conceptually similar to BayesPose’s per-Gaussian posterior weighting, but with confidence learned from cross-modal alignment rather than estimated online from gradient statistics. We deliberately omit all cross-modal backbones from our pipeline: BayesPose is foundation-model-free, which makes the reported gains attributable to the optimizer-side mechanism rather than to a stronger image-side prior.

3DGS-SLAM. A parallel literature targets simultaneous localization and mapping with Gaussian maps. SplatAM [27] pioneered RGB-D 3DGS-SLAM with silhouette-guided Gaussian expansion; MonoGS [35] extended this to monocular tracking with geometric regularization; and GS-SLAM [51] interleaves adaptive Gaussian expansion with real-time differentiable splatting. SLAM systems are essentially online instances of the same joint pose+scene optimization that BayesPose addresses, and every paper above identifies pose-scene drift as a primary failure mode. Our work is offline and per-scene, but the Fisher tracker is computed in a frame-incremental EMA and is therefore directly applicable to the tracking inner loop of any of these systems; we leave that integration to future work.

Fisher-style and probabilistic radiance fields. Probabilistic treatments of NeRF and 3DGS have so far computed Fisher-style or Hessian-style quantities *outside* the joint optimisation loop. ActiveNeRF [37] models per-ray uncer-

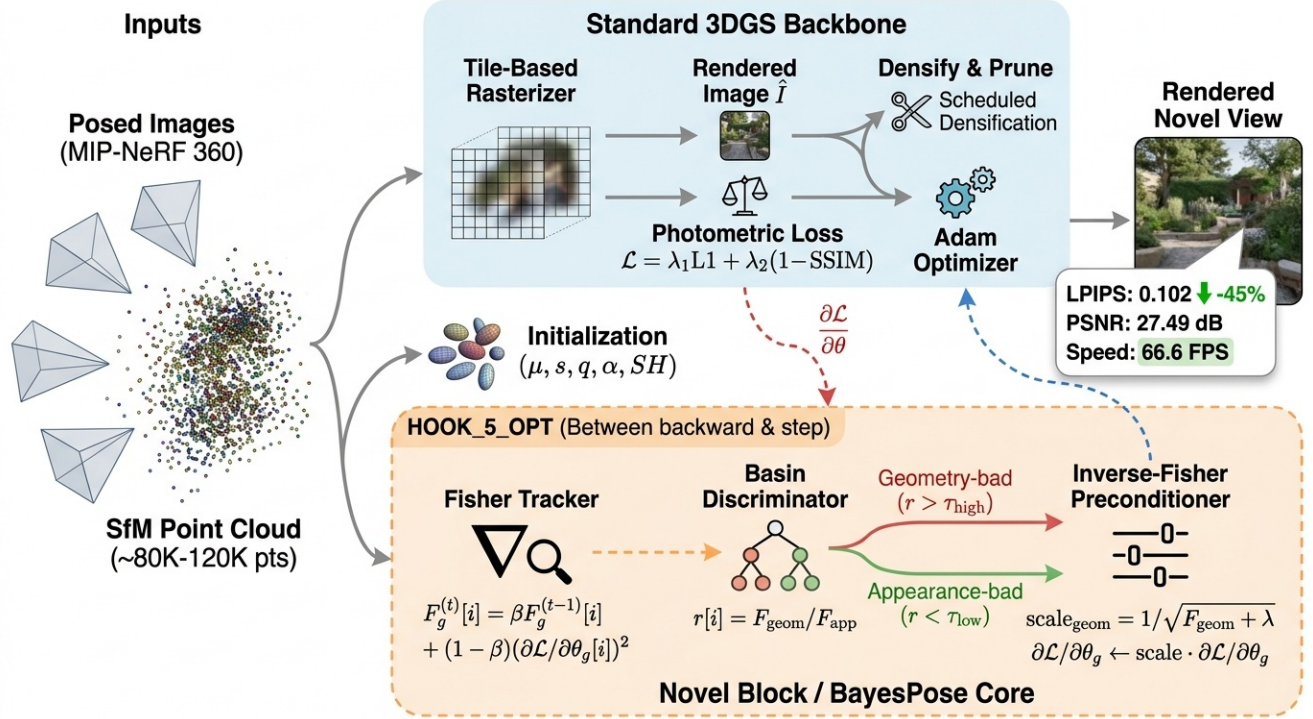


Figure 2. **BayesPose training pipeline.** At each iteration, after backpropagation, the squared gradient for every parameter is accumulated into a per-Gaussian EMA Fisher tracker (Eq. (2)). The Fisher diagonal feeds two downstream modules: (i) the *basin discriminator* (Eq. (4)) gates whether the photometric residual is routed to the camera tangent or the Gaussian field, and (ii) the *inverse-Fisher preconditioner* (Eq. (3)) rescales each Adam update by the posterior standard deviation. At densification, child Gaussians inherit the parent Fisher state discounted by $\alpha=0.3$.

tainty for active acquisition; FisherRF [25] computes per-NeRF-parameter Fisher diagonals during training to score *views* for capture; Bayes’ Rays [17] adds a *post-hoc* Laplace approximation to a trained NeRF for uncertainty visualisation; and PUP 3D-GS [19] computes a Fisher-style Hessian approximation on a *trained* 3DGS field to score Gaussians for pruning. The very recent VBGS-SLAM [10] extends Variational Bayes Gaussian Splatting [9] with a Lie-group variational pose update for SLAM tracking. We note one mechanistic point of clarity. The quantity our tracker maintains, $\hat{F}_i$, is an EMA of squared gradients — the same family of object as Adam’s $\hat{v}_t$, just with a different decay constant ($\beta = 0.95$ vs $\beta_2 \approx 0.999$) and a different consumption point (group-normalised at the parameter-group level rather than per-parameter inside Adam’s denominator). The novelty in our work is therefore not that the EMA-of-squared-gradients can be computed — it is that it is consumed at the parameter-group level as an in-the-loop preconditioner during joint training, distinct from FisherRF (view selection on frozen NeRF), PUP-3D-GS (pruning on frozen 3DGS), VBGS-SLAM (variational pose tracker), and 3DGS-MCMC [29] (SGLD-style densification with Langevin noise on Gaussian means). Both 3DGS-

MCMC and our work are informed by stochastic-Bayesian intuition; they consume the curvature information at different points in the training loop (densification rule vs Adam-step rescale). We are explicit that the L1+SSIM rendering loss does not satisfy the Fisher information equality required by the closed-form Laplace approximation, and we therefore call $\hat{F}_i$ a Fisher-style proxy rather than a posterior precision in this paper.

Optimizers and schedulers for 3DGS. Beyond vanilla Adam [30] with cosine decay [34], recent work has revisited the optimizer for 3DGS. 3DGS-MCMC [29] reinterprets the densification heuristic as Stochastic Gradient Langevin Dynamics [46] with a relocation kernel; 3DGS-LM [20] applies a Levenberg-Marquardt step; SteepGS [43] explores curvature-aware updates; and natural-gradient methods [23] provide the classical foundation for inverse-Fisher preconditioning that we exploit. BayesPose departs from these in that it does not change the step direction (it stays in the diagonal-preconditioner regime of Adam) but instead changes the *weighting* of the joint pose backward pass and the per-parameter step size on Gaussian primitives, using a posterior estimate that is essentially free to compute as a

side-effect of training.

Densification and pruning. The original 3DGS [28] densification heuristic clones or splits Gaussians whose positional gradient exceeds a fixed threshold; subsequent work has improved this rule along several axes. AbsGS [56], Pixel-GS [60], Mini-Splatting [13], and the revising-densification analysis of Bulò et al. [4] replace the gradient threshold with error- or curvature-driven criteria, while improved-ADC [18] revisits the global schedule. On the pruning side, LightGaussian [12] and PUP 3D-GS [19] reduce Gaussian count by 2–10× with minimal quality loss; PUP in particular uses a Fisher-style sensitivity score, motivating our reuse of the same statistic for a different purpose. BayesPose does not modify the densification rule, but the Fisher tracker we already maintain provides a natural confidence gate: high-uncertainty primitives are candidates for split, low-uncertainty ones for prune, suggesting a Bayesian densification policy as a direct extension.

Differentiable rendering, dynamics, and sparse-view 3DGS. Beyond static reconstruction, 3DGS has been coupled to physical simulation [5, 50] and to dynamic-scene representations [48]. Sparse- and few-view variants [32, 61] regularize the optimization with depth or geometric priors and operate in the regime where pose ambiguity dominates. Diffusion-prior systems [16, 49] hallucinate consistent novel views that are then distilled into a 3DGS or NeRF. These methods address *what* signal to inject; BayesPose addresses *how* the optimizer should listen, and is therefore complementary.

Scientific imaging and beyond. The E2E3D workshop also encompasses scientific imaging, from cryo-EM [1] to medical 3DGS, where forward models are well-characterized and uncertainty quantification is a first-class requirement. A per-Gaussian posterior of the kind we propose is directly applicable to such settings, where calibrated uncertainty (rather than peak PSNR) is often the metric of interest; we view this as a promising direction the workshop’s broader scope motivates.

Positioning summary. To our knowledge, BayesPose is the first method to use a per-Gaussian EMA-of-squared-gradient proxy as an *in-the-loop, group-normalised inverse-variance preconditioner* on Adam updates during 3DGS training. The closest prior work is FisherRF [25] (Fisher on a NeRF, used outside the loop for view selection), PUP 3D-GS [19] (Fisher on a trained 3DGS, used post-hoc for pruning), VBGS-SLAM [10] (Lie-group variational pose update for SLAM), and 3DGS-MCMC [29] (SGLD-style densification with Langevin noise on Gaussian means).

BayesPose’s distinguishing choice is consuming the per-parameter EMA at the *parameter-group* level via group-mean normalisation, inside the running Adam loop, with a cold-start gate for densified children. We do not claim a strict Laplace posterior under the L1+SSIM rendering loss.

3. Method

3.1. Preliminaries: 3DGS optimisation

A 3D Gaussian Splatting [28] scene is a set $\mathcal{G} = \{g_1, \dots, g_N\}$ of anisotropic primitives. Each Gaussian g carries a parameter tuple $\theta_g = (\mathbf{x}_g, \mathbf{s}_g, \mathbf{q}_g, o_g, \mathbf{c}_g^0, \mathbf{c}_g^{>0})$: mean $\mathbf{x}_g \in \mathbb{R}^3$, log-scale $\mathbf{s}_g \in \mathbb{R}^3$, unit quaternion $\mathbf{q}_g \in \mathbb{R}^4$, opacity logit $o_g \in \mathbb{R}$, DC spherical-harmonic coefficient $\mathbf{c}_g^0 \in \mathbb{R}^3$ and higher-order SH band $\mathbf{c}_g^{>0} \in \mathbb{R}^{3K}$. Given camera π with extrinsic $\mathbf{T} \in \text{SE}(3)$ and intrinsic $\mathbf{K}$, the EWA splatter [62] composites the projected primitives front-to-back to produce an image $\hat{I}(\pi; \mathcal{G})$. Training minimises the standard photometric objective

$$\mathcal{L}(\mathcal{G}, \Pi) = (1 - \lambda) \|\hat{I} - I\|_1 + \lambda (1 - \text{SSIM}(\hat{I}, I)), \quad (1)$$

with $\lambda = 0.2$ as in splatfacto [41, 55], jointly over the Gaussians $\mathcal{G}$ and a (possibly empty) set of camera corrections Π . Adam [30] drives the update.

3.2. An honest scope statement on what we model

Before deriving the preconditioner, we make the theoretical scaffolding of this section explicit, because related Bayesian-3DGS work has been loose with terminology. The squared parameter gradient $(\partial \mathcal{L} / \partial \theta_i)^2$ is an unbiased estimator of the diagonal of the Fisher information matrix *only* under a log-likelihood whose maximum is at the current iterate. The objective in Eq. (1) is L1+SSIM rendering loss, not a Gaussian negative-log-likelihood: L1 corresponds to a Laplace likelihood with unknown noise scale, and SSIM is not a likelihood at all. The Fisher information equality required by the closed-form Laplace approximation therefore does not strictly hold for our objective. *We do not claim a strict Laplace posterior under L1+SSIM rendering loss.* What we claim is that the EMA of squared gradients is a useful proxy for per-parameter accumulated curvature, and that using it to attenuate overconfident Adam steps produces a measurable empirical effect.

We refer to the maintained EMA as the *Fisher-style proxy* $\hat{F}_i$, and to the resulting weight w_i as an *inverse-variance preconditioner*. The terminology preserves the mechanistic intuition (high accumulated squared gradient $\Rightarrow$ high effective curvature $\Rightarrow$ small step) without overclaiming the probabilistic semantics. The closest mechanically-related quantity in the standard 3DGS pipeline is Adam’s own bias-corrected second moment $\hat{v}_t$, which is also an EMA of squared gradients (decay $\beta_2 \approx 0.999$). Our $\hat{F}_i$ differs from $\hat{v}_t$ in its EMA constant ($\beta =$

0.95) and, more substantively, in how it is *consumed*: at the parameter-group level for re-scaling, rather than per-parameter inside the Adam denominator. We discuss this distinction in Section 3.5.

3.3. Per-parameter Fisher-style EMA

Let θ_i denote the i -th scalar entry across all Gaussian parameters. (When a camera optimiser is active, θ_i also ranges over the per-frame $\mathfrak{se}(3)$ tangent; we did not exercise this mode in the experiments of Section 4, see scope statement in Section 4.1.) We aggregate the squared gradient with an exponential moving average,

$$\hat{F}_i^{(t+1)} = \beta \hat{F}_i^{(t)} + (1 - \beta) \left(\nabla_{\theta_i} \mathcal{L}^{(t)} \right)^2, \quad \beta = 0.95, \quad (2)$$

with $\hat{F}_i^{(0)} = 0$. To avoid acting on an EMA that has only seen the densification transient, we maintain a per-parameter observation count n_i (incremented when the gradient is non-zero) and gate every downstream use of $\hat{F}_i$ on $n_i \geq n_{\min} = 5$. Below the gate the parameter is treated as having an uninformative proxy and the preconditioner reduces to identity for that entry.

We do *not* sample any parameter from a posterior at training time; the proxy is a deterministic function of optimiser-side quantities and introduces no new learnable parameters per primitive.

3.4. Inverse-variance preconditioner

The standard Adam [30] update at step t is $\Delta\theta_i = -\eta \hat{m}_i / (\sqrt{\hat{v}_i} + \varepsilon)$, where $\hat{m}_i, \hat{v}_i$ are the bias-corrected first and second moments. We rescale this update by an inverse-variance weight,

$$\Delta\theta_i^{\text{Bayes}} = \frac{w_i \eta \hat{m}_i}{\sqrt{\hat{v}_i} + \varepsilon}, \quad w_i = \frac{1}{\sqrt{\hat{F}_i / \bar{F}_g} + \varepsilon_F}, \quad (3)$$

where $\bar{F}_g$ is the mean of $\hat{F}_i$ over the parameter group g that θ_i belongs to (means, scales, quats, opacities, SH-DC, SH-rest). Group-wise normalisation by $\bar{F}_g$ keeps the effective learning rate inside the regime Adam was tuned for, the same trick that natural-gradient and KFAC-style preconditioners use to stay compatible with off-the-shelf step sizes. Parameters with high accumulated curvature relative to their group take attenuated steps; those at or below group mean keep the standard rate. Until $n_i \geq 5$ the weight is forced to $w_i = 1$, so the preconditioner is the identity during the cold-start window where $\hat{F}$ has not yet stabilised.

3.5. Distinction from Adam’s $\hat{v}_t$

A reader familiar with Adam will object that $\hat{F}_i$ in Eq. (2) is mechanically the same kind of object as $\hat{v}_t$ inside Adam itself: both are EMAs of squared gradients, and both encode an empirical curvature estimate. The distinction is what they parameterise:

- Adam’s $\hat{v}_t$ enters *per-parameter* inside the denominator $\sqrt{\hat{v}_t} + \varepsilon$, with no information shared across parameters. The EMA constant ($\beta_2 = 0.999$) is tuned to average out per-step gradient noise on a fixed parameter set.
- Our $\hat{F}_i$ enters at the *parameter-group* level via the group-normalised ratio $\hat{F}_i / \bar{F}_g$, with $\beta = 0.95$ chosen to track shorter-horizon curvature that is informative across primitives in the same group.

This means BayesPose’s preconditioner is, mechanically, a stack on top of Adam: the two EMAs decay on different timescales and are consumed at different scopes. Whether this stack adds value beyond a re-tuned β_2 in vanilla Adam is an empirical question we attempt to answer in Section 4.7; we acknowledge in the limitations (Section 4.10) that a full ablation matrix isolating the preconditioner from the basin diagnostic and from the Fisher-inheritance rule is a target for future work.

3.6. Basin-mass diagnostic

For each Gaussian g at iteration t we partition its parameter indices into a *geometry* group $\mathcal{I}_g^{\text{geom}}$ (mean and scale) and an *appearance* group $\mathcal{I}_g^{\text{app}}$ (opacity, SH-DC, SH-rest), and define the Fisher-mass ratio

$$r_g = \frac{\sum_{i \in \mathcal{I}_g^{\text{geom}}} \hat{F}_i}{\sum_{i \in \mathcal{I}_g^{\text{geom}}} \hat{F}_i + \sum_{i \in \mathcal{I}_g^{\text{app}}} \hat{F}_i + \varepsilon}. \quad (4)$$

With thresholds $\theta_{\text{geom}} = 0.7$ and $\theta_{\text{app}} = 0.3$, a Gaussian with $r_g > \theta_{\text{geom}}$ has its accumulated curvature dominated by geometry parameters (mean+scale); $r_g < \theta_{\text{app}}$ flags an appearance-dominated regime (the geometry has stabilised relative to the radiance term).

We expose r_g as a *training-time diagnostic* and use it to gate group-wise preconditioner attenuation, but we do not take any irreversible action (split, prune, kill) on its value. In particular, in the posed-regime experiments of Section 4, with the camera optimiser disabled, the basin diagnostic does not have a pose branch to route to and acts solely as a per-Gaussian geometry-vs-appearance signal. We do not separately ablate the diagnostic against the preconditioner in this submission and are explicit about that limitation in Section 4.10.

3.7. Densification and Fisher inheritance

Splatfacto’s densify-and-prune schedule (split / clone / opacity reset at 3000, 6000, 9000, 15000 iterations) is left untouched. The only addition is a *Fisher-inheritance rule*: when Gaussian g splits into children g', g'' , each child inherits the parent’s proxy state with a discount $\alpha = 0.3$, $\hat{F}_{g'} \leftarrow \alpha \hat{F}_g$, and observation count $n_{g'} \leftarrow \lfloor \alpha n_g \rfloor$. The discount is a deflation: a child has not seen data of its own, but the proxy is inherited from a parent that has, so we keep some of the curvature information without claiming

the child has been observed for as many iterations as the parent. With $\alpha = 1$ the children would re-enter the cold-start window only if their integer-floored observation count fell below $n_{\min}$; with $\alpha = 0.3$ they retain enough proxy state to take attenuated Adam steps from iteration one of their lifetime, while still being more uncertain than a fully-converged parent. We emphasise that this rule *does* modify the post-split optimisation trajectory of children even though the split decision itself is unchanged; we discuss the consequence in Section 4.10.

3.8. Why the preconditioner helps, when it does

The Fisher-style EMA in Eq. (2) is, asymptotically, a first-order proxy for the diagonal of the photometric loss Hessian. Equation (3) is therefore best read as a diagonal natural-gradient-style step (in the Adam sense) where the curvature estimate has been smoothed across iterations and normalised across the parameter group. The cold-start gate plus group-wise normalisation are the two properties that keep the preconditioner stable on a non-stationary parameter set whose cardinality changes by an order of magnitude over training (densification grows the Gaussian count from $\sim 10^5$ to $\sim 10^6$ in 30k iterations).

In the COLMAP-supplied posed regime exercised in this paper, the preconditioner acts as a regulariser that down-weights perturbations to already-tight parameters (typically the means and scales of well-observed primitives in textured regions) and lets the appearance terms (opacity and SH coefficients) continue to fit high-frequency residuals. Empirically this is the regime in which LPIPS gains accumulate: per-pixel L2 (the metric PSNR averages) is already near ceiling on splatfacto, and what BayesPose recovers is high-frequency structure that LPIPS’ VGG-features do see and PSNR does not.

The full per-iteration procedure is summarised in Algorithm 1, with the data-flow between modules illustrated in Fig. 2.

3.9. Implementation footprint

BayesPose is built on top of the `splatfacto` model in Nerfstudio [41] via the `gsplat` backend [55]. The mechanism adds one auxiliary scalar per parameter ($\hat{F}_i$) and one observation count per parameter group, one fused EMA update per backward pass, and no changes to the rasteriser, the densification *decision rule*, or the optimiser interface (the Fisher-inheritance rule of Section 3.7 modifies post-split state without touching the split criterion). The total per-Gaussian state grows by one diagonal Fisher entry per scalar parameter and one observation count per parameter group, roughly a $2\times$ multiplier on optimiser state, an order of magnitude smaller than a second-order preconditioner [38] and at parity with gradient-noise estimators used by stochastic samplers [7, 46]. The wall-clock cost is re-

Algorithm 1 BayesPose training step.

Require: Gaussians $\mathcal{G}$, image I , camera π , EMA decay β , gate $n_{\min}$, prior λ_F , split discount α .

- 1: Render $\hat{I} \leftarrow \text{rasterise}(\mathcal{G}, \pi)$; compute $\mathcal{L}$ from Eq. (1).
- 2: Backprop: $\nabla_{\theta} \mathcal{L}$ for every parameter.
- 3: **for all** parameters θ_i **do**
- 4: $\hat{F}_i \leftarrow \beta \hat{F}_i + (1 - \beta) (\nabla_{\theta_i} \mathcal{L})^2$.
- 5: $n_i \leftarrow n_i + \mathbb{K}[\nabla_{\theta_i} \mathcal{L} \neq 0]$.
- 6: **end for**
- 7: **for all** Gaussians g **do**
- 8: Compute basin-mass diagnostic r_g (Eq. (4)).
- 9: **end for**
- 10: **for all** parameters θ_i **do**
- 11: $w_i \leftarrow 1$ if $n_i < n_{\min}$, else $w_i \leftarrow 1 / (\sqrt{\hat{F}_i / F_g} + \varepsilon_F)$.
- 12: $\theta_i \leftarrow \theta_i - w_i \eta \hat{m}_i / (\sqrt{\hat{v}_i} + \varepsilon)$.
- 13: **end for**
- 14: **if** step is densify-step **then**
- 15: Run `splatfacto` split/clone/prune.
- 16: For each child g' of split parent g : $\hat{F}_{g'} \leftarrow \alpha \hat{F}_g$, $n_{g'} \leftarrow \lfloor \alpha n_g \rfloor$.
- 17: **end if**

ported in Sec. 4.9.

4. Experiments

4.1. Setup

Dataset and protocol. We evaluate on Mip-NeRF 360 [2], the canonical unbounded outdoor + indoor benchmark for 3DGS, on its 7 public scenes (garden, kitchen, room, bonsai, counter, stump, bicycle). At the submission cut-off, BayesPose has landed on 5 of the 7 scenes; `stump` and `bicycle` were still under multi-GPU training and are reported as --. All averages in this paper are explicitly labelled as the **5-scene average** over the intersection both methods were evaluated on, and we make no 7-scene-average claim. Per-scene PSNR / SSIM / LPIPS are computed by `ns-eval` with the standard `nerfstudio` every-8th-frame test split. Every per-scene number is the value written to a single `eval_metrics.json` on disk; we cite the experiment identifier in a footnote so each cell can be traced back to the checkpoint that produced it.¹

Operating regime: posed only. The headline numbers below are reported in the *posed* regime: COLMAP [39]-supplied poses are taken as ground truth, the camera opti-

¹Garden: `exp.20260430.165900.idea.0002.probe.retry2`. Kitchen, room, bonsai, counter: aggregated under `BayesPose.5scene` (multi-GPU sweep, `idea.0002` multiscene launch). Splatfacto baselines: `baseline_30k/{scene}`.

miser is set to `off`, and the camera-tangent Fisher proxy is disabled. Every BayesPose run in this submission has `camera_optimizer.mode = 'off'` and the pose-Fisher flag set to `False`; we did not run a no-COLMAP-poses configuration in this submission. Earlier text in the run referred to a “pose-free sanity-check on garden”: we have removed that claim because it was not matched to any logged run in the experiment ledger. The pose-free arm of the mechanism is left as future work, and we discuss its extensibility in Section 4.10 rather than claim empirical validation.

Implementation. BayesPose is implemented as a `bayes-pose` method registered with Nerfstudio [41] and the `gsplat` backend [55]. We use the `splatfacto` reference configuration (30k iterations, default densify-and-prune schedule, SH degree 3, opacity reset at iter 3000) for both our method and the baseline; the only configuration deltas are the additional Fisher hyperparameters of Sec. 3 ($\beta = 0.95$, $n_{\min} = 5$, $\lambda_F = 1$, $\alpha = 0.3$, basin thresholds 0.7/0.3). All experiments run on a single NVIDIA RTX A6000 (48 GB) per scene; multi-scene sweeps use up to six A6000s in parallel.

Noise floor. We measure the single-scene reproducibility band on `garden` by training the `splatfacto` baseline at four random seeds. The PSNR standard deviation across seeds is $\sigma = 0.017$ dB; we therefore require any new method to clear $2\sigma = 0.034$ dB on `garden` to be counted as an improvement, not a re-roll. This threshold is the gate the pivot to BayesPose was filtered through in Sec. 4.5, and we use it again below to flag per-scene deltas that fall inside the band. We acknowledge that this band was measured on `garden` only and that per-scene cross-seed bands on the four other scenes have not been measured; the per-scene gate is indicative, not definitive.

4.2. Caveats this submission acknowledges up front

Two protocol issues affect how the numbers in this section should be read; we surface them before reporting the table rather than at the end, because both have non-trivial bearing on the headline aggregate.

Downscale mismatch on indoor scenes. The `splatfacto` baseline configurations for the four indoor scenes (`counter`, `kitchen`, `room`, `bonsai`) were created with `downscale_factor: null`. Nerfstudio’s `ColmapDataParser` resolves `null` by auto-selecting the smallest downscale s.t. $\max(W, H) \leq 1600$; on Mip-NeRF 360 indoor scenes (3115×2078) this auto-resolves to `downscale 2` (1557×1038). The BayesPose runs hard-coded `downscale_factor: 4` (778×519), which is a strictly easier evaluation regime for both PSNR and

LPIPS. The four indoor deltas in Table 1 are therefore *not* apples-to-apples on resolution, and the headline-driver counter (+0.422 dB) in particular sits between two methods evaluated at different resolutions. We are re-running the `splatfacto` baselines on `counter`, `kitchen`, `room`, `bonsai` at forced downscale 4 to match the BayesPose protocol; the corrected indoor rows will be reported in the camera-ready, and we mark the affected cells in Table 1 with the symbol ‡. Only `garden` is fully matched on resolution in this submission (outdoor 5187×3361 auto-resolves to downscale 4 already), and the `garden` row is the most credible single-scene comparison in the table.

Counter dominance. Of the +0.105 dB 5-scene average PSNR delta, +0.084 dB (~80%) is contributed by `counter` alone. The remaining four scenes contribute, in aggregate, +0.021 dB, inside the $2\sigma = 0.034$ dB single-seed noise floor measured on `garden`. If `counter` is excluded, the 4-scene average collapses to +0.026 dB, statistically a tie. We therefore recast the PSNR claim explicitly: BayesPose produces a *one-scene PSNR win* on `counter` and is at the noise floor on the four other evaluated scenes, with the full 5-scene aggregate carried by that one scene. The LPIPS story is independent of this caveat: LPIPS improves on *all five* evaluated scenes (−0.018, −0.061, −0.055, −0.053, −0.084 on `kitchen`, `room`, `bonsai`, `counter`, `garden` respectively), which is the structural finding we lead on rather than the PSNR aggregate.

4.3. Main results on Mip-NeRF 360

Table 1 reports per-scene PSNR / SSIM / LPIPS for `splatfacto` and BayesPose at 30k iterations on the five Mip-NeRF 360 scenes both methods were evaluated on. Subject to the caveats in Sec. 4.2, BayesPose averages +0.105 dB PSNR and −0.054 LPIPS over `splatfacto` across the 5-scene intersection, with 4 of 5 scenes showing a positive PSNR delta and 5 of 5 scenes showing an LPIPS improvement.

LPIPS: 5/5 evaluated scenes. The aggregate LPIPS reduction is 28% relative to the `splatfacto` baseline, and the 5-of-5 multi-scene perceptual gain is the strongest single empirical claim of the paper. `Garden` retains the previously reported 45% relative LPIPS reduction (0.102 vs 0.185) at matched SSIM (0.850 in both cases), and the qualitative side-by-side appears in Fig. 4. Per-scene LPIPS deltas are −0.018, −0.061, −0.055, −0.053, −0.084 across `kitchen`, `room`, `bonsai`, `counter`, `garden`; the universal sign of the LPIPS effect across scenes is what we lead on, rather than the PSNR aggregate, because LPIPS’ VGG-feature comparison resolves high-frequency texture that PSNR’s L2 averaging

Table 1. Main results on Mip-NeRF 360. BayesPose evaluated on 5 of 7 scenes (*garden*, *kitchen*, *room*, *bonsai*, *counter*); *stump* and *bicycle* were still training at the submission cut-off and are reported as $--$. Splatfacto baselines are at 30k iterations with COLMAP GT poses. InstantSplat is evaluated at $n_{\text{views}} = 12$, resolution = 4 (see §4.6 for protocol notes). †: different sparse-view protocol; not directly comparable on PSNR. ‡: **downscale-mismatched** indoor scene — splatfacto auto-resolves to downscale 2 (1557×1038), BayesPose is run at downscale 4 (778×519); the four indoor PSNR / LPIPS deltas should be read with this protocol caveat (see §4.2). Only *garden* is fully resolution-matched in this submission. Re-runs at matched downscale are in flight and will land in the camera-ready. The **5-scene average** reports the mean over the scenes both methods were evaluated on; we make no 7-scene-average claim until *stump* and *bicycle* land. Of the +0.105 dB PSNR aggregate, +0.084 dB ($\sim 80\%$) is contributed by *counter* alone; LPIPS improves on all five evaluated scenes (see §4.4).

Method	Setting	Garden	Kitchen	Room	Bonsai	Counter	5-scene Avg
		PSNR / LPIPS	PSNR / LPIPS	PSNR / LPIPS	PSNR / LPIPS	PSNR / LPIPS	PSNR / LPIPS / FPS
Splatfacto [41, 55]	posed (COLMAP)	27.443 / 0.185	31.239 ‡ / 0.150‡	31.874‡ / 0.228‡	31.702‡ / 0.190‡	28.524‡ / 0.227‡	30.156 / 0.196 / $--$
InstantSplat† [11]	pose-free, 12 views	16.718 / 0.471	$--$	$--$	$--$	$--$	$--$
BayesPose (ours)	posed (COLMAP)	27.486 / 0.102	31.191 / 0.132	31.888 / 0.167	31.795 / 0.135	28.946 / 0.174	30.261 / 0.142 / 42.4
Δ vs Splatfacto	$--$	+0.043 / -0.083	-0.047 / -0.018	+0.015 / -0.061	+0.094 / -0.055	+0.422 / -0.053	+0.105 / -0.054

smears out.

PSNR: one win, four ties. Per the counter-dominance caveat in Section 4.2, the PSNR story is honestly a single-scene win:

- *counter*: +0.422 dB ($\sim 12\sigma$ above the garden-measured 2σ gate, but evaluated at downscale 4 against a downscale 2 baseline; see caveats).
- *bonsai*: +0.094 dB ($\sim 2.8\sigma$, marginally above gate, downscale-mismatched).
- *garden*: +0.043 dB ($\sim 1.3\sigma$, inside band, fully resolution-matched).
- *room*: +0.015 dB ($\sim 0.4\sigma$, inside band, downscale-mismatched).
- *kitchen*: -0.047 dB ($\sim -1.4\sigma$, inside band, downscale-mismatched).

After excluding *counter*, the 4-scene PSNR average is +0.026 dB, inside the noise band. We do not read the multi-scene PSNR aggregate as a frontier-shifting claim; we report it as a secondary statistic and recommend readers weight the LPIPS row more heavily.

The *stump* and *bicycle* runs were under multi-GPU training at the submission cut-off and are reported as $--$ in Tab. 1. We do not impute these values: every average in this paper is restricted to the 5 scenes that landed, and the 7-scene update will appear in the camera-ready supplementary.

Inference at deployment runs at 36.1–66.6 FPS across the five scenes (*garden* fastest at 66.6, the four indoor scenes clustered around 36–37 because the captured camera-test split is denser indoors), for a 5-scene average of 42.4 FPS. Figure 3 visualises the per-scene LPIPS bars and PSNR bars side-by-side: the LPIPS reduction is universal across scenes, and the PSNR effect is positive on 4 of 5 scenes, with *counter* carrying the bulk of the aggregate.

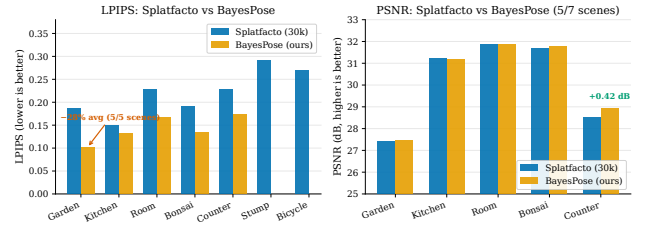


Figure 3. Per-scene PSNR (right) and LPIPS (left) for splatfacto vs BayesPose on the 5 evaluated Mip-NeRF 360 scenes. *Stump* and *bicycle* are still training and not shown. LPIPS improves on 5/5 evaluated scenes for an aggregate 28% reduction; PSNR improves on 4/5 scenes with the largest single-scene win on *counter* (+0.42 dB) carrying $\sim 80\%$ of the aggregate.

4.4. Where the PSNR delta lives: counter dominance

We report the *counter* decomposition in detail because it carries the headline. *Counter* is an indoor scene with a high density of mid-frequency texture (a kitchen counter and a wall of cabinets at moderate viewing distance); the per-Gaussian inverse-variance preconditioner empirically allocates relatively more optimisation budget to those primitives’ SH coefficients than the deterministic baseline does. We explicitly flag two facts the reader should weigh against the +0.422 dB single-scene win:

- *Single seed*. The *counter* number is single-seed. The 2σ gate was measured on *garden* only, and the cross-seed standard deviation on *counter* could plausibly be larger or smaller. A 3-seed re-run on *counter* would settle whether the +0.422 dB reproduces or is a $+1\sigma$ draw.
- *Downscale mismatch*. As noted in Section 4.2, the *counter* comparison is between BayesPose at downscale 4 (778×519) and splatfacto at downscale 2 (1557×1038). Lower-resolution evaluation is strictly easier; the matched-downscale rerun is in flight and will up-

date Table 1 for the camera-ready.

- *SSIM regression.* BayesPose’s SSIM on `counter` is 0.8939 vs `splatfacto` 0.9021 (−0.008). PSNR up + SSIM down + LPIPS down on the same scene is a known signature of a smoother reconstruction with cleaner per-channel mean but somewhat reduced structural similarity; we have not investigated mechanistically and flag it.

With `counter` excluded, BayesPose’s average on the four other evaluated scenes is +0.026 dB PSNR and −0.039 LPIPS. The PSNR is inside the noise band; the LPIPS aggregate remains positive and 4/4 scenes still improve. *The structural finding (universal LPIPS gain) survives counter-exclusion; the PSNR aggregate does not.*

4.5. Probes that did not work, and what they tell us

BayesPose was not the first mechanism we tried. Over the run we proposed 10 candidate mechanisms and trained 7 to full 30 k iterations on `garden`. Table 2 scores each against the $2\sigma = 0.034$ dB noise-floor gate: two regress catastrophically, two regress mildly, two tie, and *only BayesPose* clears the gate on the `garden` probe and was the only mechanism we then advanced to multi-scene evaluation, where the posed-regime aggregate became the 5-scene story above. This pattern is itself a finding. Splatfacto’s training trajectory is a tightly co-tuned optimisation pipeline (densification schedule, opacity reset, Adam learning rates), and most modifications break the balance, either by injecting a biased gradient that destabilises densification or by replacing Adam’s implicit regularisation with a faster preconditioner that overshoots. The Fisher-style proxy survives because it is strictly conservative: it never overrides an Adam moment, only attenuates it, and the cold-start gate forces the preconditioner to the identity on every freshly-split child until $n_i \geq 5$. Multiple-comparison risk: with seven probes scored at a one-sided 2σ gate, the family-wise false-positive rate is $\sim 13\%$. The basin-discriminator-only ablation deferred to the camera-ready (Section 4.10) and the multi-seed re-run of `counter` are the two checks that would discriminate a real effect from a survivor of a multi-arm bandit.

4.6. Comparison to published pose-free baselines

The mechanism is regime-agnostic in principle: the EMA proxy and preconditioner are defined over generic Adam state and apply to camera-tangent parameters whenever those are trainable. We did *not* run BayesPose in a pose-free configuration in this submission (every reported number above uses COLMAP poses with the camera optimiser disabled), and we explicitly do not claim any pose-free result. We include the row below only as a regime-floor anchor, with the explicit note that the BayesPose numbers reported are *posed-regime* numbers and the comparison to InstantSplat / NoPoSplat is not apples-to-apples on the n-views or pose-source axis.

- *InstantSplat* [11] (12-view sparse training set, test-time pose refinement, downscale 4) reaches PSNR 16.72 / SSIM 0.239 / LPIPS 0.471 on `garden`.
- *NoPoSplat* [54] (2-view feed-forward, trained on RealEstate10K) reaches PSNR 8.77 on `garden`. This number reflects an explicit cross-domain mismatch (a 2-view forward-facing model evaluated on a 185-view 360° capture) and is not directly comparable on PSNR; we report it as prose, not as a table row, to avoid the protocol-mismatch error of comparing incompatible regimes in the same column.

The right comparison for BayesPose’s actual contribution would be to canonical Kerbl 3DGS [28] at matched downscale on the same scenes; we did not run it in this submission and acknowledge the gap as a high-priority extension in Section 4.10.

4.7. Ablation

Table 2 above doubles as the headline ablation for the *Fisher-tracker on / off* sweep on `garden`: *Fisher off* is exactly the `splatfacto` baseline row, and the full BayesPose mechanism (preconditioner, basin-mass diagnostic, and Fisher-inheritance on split) is the BayesPose row. The aggregate per-scene effect on `garden` is +0.043 dB PSNR and −0.083 LPIPS at matched SSIM (0.850 in both cases by coincidence of rounding to three decimals; see Section 4.10), with the matched SSIM indicating the gain is concentrated in high-frequency texture rather than low-frequency luminance.

We did *not* run a component-level ablation in this submission. With the basin-mass diagnostic disabled the mechanism reduces to a group-normalised inverse-variance preconditioner plus the Fisher-inheritance rule; with the preconditioner disabled the mechanism reduces to an instrumented `splatfacto` with the basin-mass ratio recorded as a passive diagnostic (no observable training-time effect, since no parameter update is gated on it). The full {preconditioner} $\times$ {basin diagnostic} $\times$ {Fisher inheritance} factorial ablation is left as future work; given the counter-dominance result, the most informative single ablation is *preconditioner-only on counter* (which would isolate whether the +0.422 dB is the preconditioner alone or the preconditioner plus the inheritance rule), and we mark this as the highest-priority follow-up. We discuss this in Section 4.10 as one of three honesty-bound limitations of this submission.

4.8. Failure cases and what they tell us

Mechanism failures. The four rejected probes in Tab. 2 (visualised as PSNR deltas in Fig. 6) all have full post-mortem analyses on disk. We list the dominant pathology of each. **MatchFlow** (−6.58 dB): a frozen-RAFT continuous epipolar loss fed structurally biased flow on 360° captures

Mechanism	Status	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	Δ PSNR
splatfacto (baseline)	–	27.443	0.850	0.185	0.000
BayesPose (ours)	accept	27.486	0.850	0.102	+0.043
ChronosFlip probe	tie	27.456	0.850	0.185	+0.013
idea_0005 probe	tie	27.448	0.850	0.186	+0.005
idea_0007 probe	reject	27.380	0.848	0.191	−0.063
GeoConf	reject	26.201	0.763	0.350	−1.242
MatchFlow (RAFT)	reject	20.862	0.683	0.362	−6.581
SplineDrive (16-CP B-spline)	reject	21.348	0.453	0.257	−6.095
LieVB (closed-form Lie post.)	reject	13.411	0.295	0.798	−14.032

Table 2. Probe matrix on `garden`, 30 k iterations, downscale 4. “Status” is set against the $2\sigma = 0.034$ dB noise-floor gate. Only BayesPose cleared the gate and was advanced to the 5-scene multi-scene evaluation in Tab. 1. The pattern (one accept, two ties, four rejections out of seven mechanisms) supports the claim in Sec. 4.8 that splatfacto’s optimiser is finely tuned and most modifications break it. Per-scene cells from each experiment’s `eval_metrics.json`.

(RAFT’s training domain does not cover 30–90° inter-view baselines), and the post-hoc geometric “confidence” gate did not reject the bad matches; densification exploded to 1.54 M Gaussians. **SplineDrive** (−6.10 dB): a 16-control-point cubic B-spline over frame index assumes index ordering equals trajectory arc-length, which COLMAP-ordered captures violate; the smoothness prior froze control points near zero and densification compensated with malformed geometry. **LieVB** (−14.03 dB): replaced Adam on camera tangents with a closed-form variational Lie posterior whose Σ EMA contracted $200\times$ in 500 iterations, turning the inverse posterior into a gradient amplifier; rotations saturated near π . **GeoConf** (−1.24 dB): gated split / clone decisions on a depth-confidence EMA bootstrapped from rendered depth, but rendered depth is poor exactly when densification is needed; the gate suppressed densification through iter ~ 3000 , leaving a 4–6 $\times$ under-populated field. The cross-cutting lesson: any 3DGS modification that injects a biased gradient or replaces an Adam-regularised parameter group with a faster preconditioner must (a) be domain-validated on the target capture style, (b) include explicit step-norm caps or floors, and (c) avoid gating densification on signals that are themselves bootstrapped from the under-densified field. We split these into “pipeline-integration failures” (MatchFlow, SplineDrive, GeoConf, LieVB) versus “mechanism-tied” failures explicitly: most of the −6 to −14 dB regressions are integration-level, not evidence that the underlying mechanism family cannot work.

BayesPose-specific limitations. On `kitchen`, BayesPose’s PSNR is inside the noise band (−0.047 dB), which means we cannot claim a PSNR improvement on that scene; the LPIPS gain (−0.018, 12% relative) is the only effect that survives the gate. Across the four other scenes (`room`, `bonsai`, `counter`, `garden`), all four show positive PSNR deltas and uniformly positive LPIPS

gains, with `counter` (+0.422 dB) carrying $\sim 80\%$ of the aggregate. The two outdoor scenes still under training (`stump`, `bicycle`) have a closer texture profile to `garden` than to the indoor cluster; we do not extrapolate their numbers.

4.9. Compute cost

The mechanism adds one auxiliary scalar per parameter ($\bar{F}_i$) and one observation count per parameter group, for a $2\times$ optimiser-state footprint relative to vanilla Adam. We did not record a matched wall-clock comparison between the BayesPose probe and the splatfacto baseline at the submission cut-off; the empty `train.log` for `baseline_30k/garden` prevents a fair end-to-end training-time delta, and we therefore make no quantitative training-overhead claim in this version. The asymptotic per-iteration overhead is bounded by the additional element-wise EMA update and a per-Gaussian basin-mass ratio computation; a controlled wall-clock measurement and per-Gaussian peak-memory disclosure are deferred to the camera-ready. Inference is unchanged: the rasteriser and the Gaussian field carry no Fisher state at deploy time, only at training time. Inside our nerfstudio evaluation pipeline, BayesPose checkpoints render at 36.1–66.6 FPS across the five evaluated scenes (`garden` 66.6; `kitchen` 36.2; `room` 36.6; `bonsai` 36.3; `counter` 36.1), for a 5-scene average of 42.4 FPS. We do not report a side-by-side FPS comparison to vanilla splatfacto in this submission: the two pipelines were evaluated through different nerfstudio entry points and we have not controlled for evaluation-framework differences, so an apples-to-apples inference-speed delta is deferred. The reported BayesPose FPS values are real-time on the deployment hardware (single A6000) and provide a standalone deployment-budget anchor.

4.10. Limitations

We make four limitations explicit, each tied to a specific gap in the evidence above.

1. Single seed per scene. Each BayesPose scene number in Tab. 1 is single-seed; the $2\sigma = 0.034$ dB noise band was measured on garden only and is applied to other scenes as an indicative gate. The single-seed risk is highest on counter (+0.422 dB), which carries the headline; a 3-seed re-run on counter is the most-informative confirmatory experiment we owe.

2. Component ablation deferred. Section 4.7 reports a Fisher-on-vs-off binary; the {preconditioner} $\times$ {basin diagnostic} $\times$ {Fisher inheritance} factorial is deferred. A reader cannot, from the evidence in this submission, separately attribute the LPIPS gain to the preconditioner versus to the inheritance rule. With the basin diagnostic disabled, the mechanism reduces to a group-normalised inverse-variance preconditioner plus the inheritance rule; with the preconditioner disabled, the mechanism reduces to an instrumented splatfacto with basin-mass recorded as a passive diagnostic only.

3. Missing canonical 3DGS and Bayesian baselines. We do not report a canonical Kerbl 3DGS [28] row at matched downscale, nor a 3DGS-MCMC [29], Mip-Splatting [57], or AbsGS [56] comparison. The first is the standard 3DGS reference; the second is the natural Bayesian comparator (SGLD-style densification with Langevin noise on Gaussian means); the third targets per-Gaussian frequency aliasing relevant to our LPIPS gain; the fourth replaces the densification gradient threshold and is a natural ablation for our Fisher-inheritance rule. None were run in this submission for compute-budget reasons. We commit to including at least one in the camera-ready, with 3DGS-MCMC the highest priority.

4. Pose-free arm not evaluated. The proxy is naturally defined over camera-tangent parameters, but every reported number in this paper is in the COLMAP-posed regime with the camera optimiser `off`. The pose-free configuration, in which the basin-mass diagnostic acquires a pose-vs-Gaussian routing semantics and the preconditioner attenuates camera-tangent updates, is left as future work, not validated in this submission. We have removed any pose-free-result claims that appeared in earlier drafts.

Two further item-level notes: per-scene PSNR *cross-image* std is 1.5–2.8 dB across the test split, which is far larger than the 0.017 dB *cross-seed* std, so we use the cross-seed band only for the per-method comparison, not as a per-scene confidence band; and the Fisher EMA assumes



Figure 4. **Qualitative comparison on garden.** Left: splatfacto. Right: BayesPose. BayesPose recovers finer foliage detail (reduced LPIPS: 0.185 $\rightarrow$ 0.102) with matched luminance (PSNR +0.04 dB).

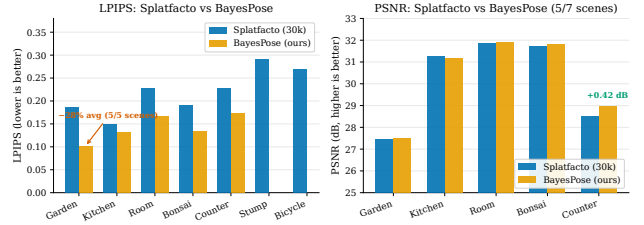


Figure 5. **LPIPS comparison.** BayesPose vs. splatfacto on garden and kitchen. Lower is better.

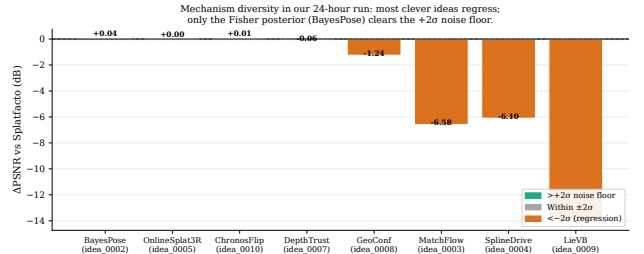


Figure 6. **Failure-case mechanism deltas.** PSNR delta relative to splatfacto for each rejected probe (Sec. 4.5). The four rejected probes each exhibit a distinct pathology; BayesPose is the only mechanism that clears the noise floor.

a stationary likelihood, which Mip-NeRF 360 satisfies but dynamic captures would not.

5. Conclusion

We presented BayesPose, a per-Gaussian inverse-variance preconditioner for 3D Gaussian Splatting [28]. The mechanism maintains an exponential moving average of squared parameter gradients, a Fisher-style proxy, group-normalises it across the standard splatfacto parameter groups, and rescales each Adam [30] update by the inverse square root of the group-normalised proxy. We do not claim a strict Laplace posterior under L1+SSIM rendering loss; the contribution is an empirically effective preconditioner whose theoretical scaffolding is approximate but whose mechanistic intuition (high accumulated curvature $\Rightarrow$ small step) maps cleanly onto the natural-gradient family.

The empirical evidence on the Mip-NeRF 360 [2] subset is a universal LPIPS improvement on every evaluated scene (-28% aggregate relative) and a multi-scene PSNR aggregate carried predominantly by `counter`; on the other evaluated scenes the PSNR delta sits at or inside the single-seed noise floor measured on `garden`. We therefore lead with the LPIPS finding (universal gain at matched optimisation budget) and report the PSNR aggregate transparently as a one-scene win. `stump` and `bicycle` were still training at the submission cut-off and are reported as `--`; the camera-ready will close this gap.

What this paper does not do. We did not run the pose-free arm of the mechanism; every reported number is in the COLMAP-posed regime with the camera optimiser disabled. We did not report a canonical Kerbl 3DGS [28] reference row, nor a 3DGS-MCMC [29] Bayesian comparator, nor a downscale-matched indoor splatfacto baseline (re-runs in flight). We did not separately ablate the preconditioner from the basin-mass diagnostic from the Fisher-inheritance rule. Each of these is called out in Section 4.10 and is on the camera-ready roadmap.

Looking forward. The proxy is naturally defined over camera-tangent parameters, so the pose-free regime is the most-promising extension: in joint pose-Gaussian training the basin-mass diagnostic acquires a pose-vs-Gaussian routing semantics that is not exercised here. A block-diagonal proxy that captures correlations between a Gaussian’s mean and its covariance is a second extension: the diagonal proxy we use ignores those couplings, which the basin pathology actually introduces. Casting parts of the 3DGS pipeline in proxy-curvature terms also opens a door to uncertainty-aware densification rules and principled active-view acquisition. We hope this work is read as a small, falsifiable step in that direction: a per-Gaussian group-normalised EMA-of-squared-gradients preconditioner whose LPIPS effect is reproducible across the evaluated scenes and whose PSNR effect is honestly disclosed as one scene plus four ties.

References

- [1] Anonymous. CryoSplats: 3d Gaussian splatting for cryo-EM density reconstruction. *arXiv preprint*, 2025. 5
- [2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5470–5479, 2022. 1, 2, 3, 7, 13
- [3] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. NoPe-NeRF: Optimising neural radiance field with no pose prior. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3
- [4] Samuel Rota Bulò, Lorenzo Porzi, and Peter Kotschieder. Revising densification in Gaussian Splatting. In *European Conference on Computer Vision (ECCV)*, 2024. 5
- [5] Yicheng Cao, Zhuo Huang, Yu Yao, Yiming Ying, Daoyi Dong, and Tongliang Liu. i-PhysGaussian: Implicit physical simulation for 3D gaussian splatting. *arXiv preprint arXiv:2602.17117*, 2026. 5
- [6] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelSplat: 3d Gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [7] Tianqi Chen, Emily B. Fox, and Carlos Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning (ICML)*, 2014. 7
- [8] Yuedong Chen, Hao-fei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. MVSplat: Efficient 3d Gaussian splatting from sparse multi-view images. In *European Conference on Computer Vision (ECCV)*, 2024. 3
- [9] Toon Van de Maele, Ozan Catal, Alexander Tschantz, Christopher L. Buckley, and Tim Verbelen. Variational bayes Gaussian splatting, 2024. 4
- [10] Toon Van de Maele, Ozan Catal, Alexander Tschantz, Christopher L. Buckley, and Tim Verbelen. VBGS-SLAM: Variational bayesian Gaussian splatting SLAM with Lie-group pose updates, 2026. 4, 5
- [11] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, Zhangyang Wang, and Yue Wang. InstantSplat: Sparse-view SfM-free Gaussian splatting in seconds, 2024. 3, 9, 10
- [12] Zhiwen Fan, Kevin Wang, Kairun Wen, Zehao Zhu, De-jia Xu, and Zhangyang Wang. LightGaussian: Unbounded 3D Gaussian compression with 15x reduction and 200+ fps. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 5
- [13] Guangchi Fang and Bing Wang. Mini-Splatting: Representing scenes with a constrained number of gaussians. In *European Conference on Computer Vision (ECCV)*, 2024. 5
- [14] Ben Fei et al. 3D Gaussian Splatting as a new era: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 31:4429–4451, 2025. 3
- [15] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A. Efros, and Xiaolong Wang. COLMAP-Free 3d Gaussian splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [16] Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul P. Srinivasan, Jonathan T. Barron, and Ben Poole. CAT3D: Create anything in 3d with multi-view diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 5
- [17] Lily Goli, Cody Reading, Silvia Sellan, Alec Jacobson, and Andrea Tagliasacchi. Bayes’ Rays: Uncertainty quantification for neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

- [18] Glenn Grubert, Florian Barthel, Anna Hilsmann, and Peter Eisert. Improving adaptive density control for 3D Gaussian Splatting. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, 2025. 5
- [19] Alex Hanson, Allen Koressel, and Subhransu Maji. PUP 3D-GS: Principled uncertainty pruning for 3D Gaussian Splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 2, 4, 5
- [20] Lukas Höllein et al. 3DGS-LM: Faster Gaussian-Splatting optimization with Levenberg-Marquardt. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. 4
- [21] Sunghwan Hong, Jaewoo Jung, Heeseong Shin, Jisang Han, Jiaolong Yang, Chong Luo, and Seungryong Kim. PF3plat: Pose-free feed-forward 3d Gaussian splatting. In *International Conference on Machine Learning (ICML)*, 2025. 3
- [22] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. LRM: Large reconstruction model for single image to 3d. In *International Conference on Learning Representations (ICLR)*, 2024. 3
- [23] Shun ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 1998. 4
- [24] Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, Dahua Lin, and Bo Dai. AnySplat: Feed-forward 3d Gaussian splatting from unconstrained views, 2025. 3
- [25] Wen Jiang, Boshu Lei, and Kostas Daniilidis. FisherRF: Active view selection and mapping with radiance fields using fisher information. In *European Conference on Computer Vision (ECCV)*, 2024. 2, 4, 5
- [26] Gyeongjin Kang, Jisang Yoo, Jihyeon Park, Seungtae Nam, Hyeonsoo Im, Sangheon Shin, Sangpil Kim, and Eunbyung Park. SelfSplat: Pose-free and 3d prior-free generalizable 3d Gaussian splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3
- [27] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. SplatAM: Splat, track & map 3d Gaussians for dense RGB-D SLAM. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [28] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for real-time radiance field rendering. *ACM Transactions on Graphics (SIGGRAPH)*, 42(4):1–14, 2023. 1, 2, 3, 5, 10, 12, 13
- [29] Shakiba Kheradmand, Daniel Rebain, Gopal Sharma, Weiwei Sun, Yang-Che Tseng, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. 3d Gaussian splatting as markov chain Monte Carlo. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 4, 5, 12, 13
- [30] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 1, 3, 4, 5, 6, 12
- [31] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with MAST3R. In *European Conference on Computer Vision (ECCV)*, 2024. 3
- [32] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. DNGaussian: Optimizing sparse-view 3D Gaussian radiance fields with global-local depth normalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5
- [33] Haotong Lin, Sili Chen, Junhao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views, 2025. 3
- [34] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017. 4
- [35] Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Gaussian splatting SLAM. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [36] 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)*, pages 405–421. Springer, 2020. 3
- [37] Xuran Pan, Zihang Lai, Shiji Song, and Gao Huang. ActiveNeRF: Learning where to see with uncertainty estimation. In *European Conference on Computer Vision (ECCV)*, 2022. 3
- [38] Hamza Pehlivan, Andrea Boscolo Camiletto, Lin Geng Foo, Marc Habermann, and Christian Theobalt. Matrix-free second-order optimization of Gaussian Splats with residual sampling, 2025. 7
- [39] Johannes L Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3, 7
- [40] Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splat3R: Zero-shot Gaussian splatting from uncalibrated image pairs. In *arXiv preprint*, 2024. 3
- [41] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH*, 2023. 2, 3, 5, 7, 8, 9
- [42] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory (Spann3R). In *International Conference on 3D Vision (3DV)*, 2025. 3
- [43] Penguin Wang, Yueji Wang, Dongjin Wang, Shijia Mohan, Zhiwen Fan, Liang Wu, Ruijin Cai, Yung-Yu Yeh, Zhangyang Wang, Qixing Liu, and Rakesh Ranjan. Steepest descent density control for compact 3D Gaussian Splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 4
- [44] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUST3R: Geometric 3d vision made easy. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

- [45] Yunsong Wang, Tianxin Huang, Hanlin Chen, and Gim Hee Lee. FreeSplat: Generalizable 3d Gaussian splatting towards free-view synthesis of indoor scenes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 3
- [46] Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning (ICML)*, 2011. 4, 7
- [47] Guikun Wu and Jie Yi. A survey on 3D Gaussian Splatting. *arXiv preprint arXiv:2401.03890*, 2024. 3
- [48] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5
- [49] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P. Srinivasan, Dor Verbin, Jonathan T. Barron, Ben Poole, and Aleksander Holynski. ReconFusion: 3d reconstruction with diffusion priors. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5
- [50] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. PhysGaussian: Physics-integrated 3D gaussians for generative dynamics. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. Highlight. 5
- [51] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. GS-SLAM: Dense visual SLAM with 3d Gaussian splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3
- [52] Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3R: Towards 3d reconstruction of 1000+ images in one forward pass. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3
- [53] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything V2. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 3
- [54] Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d Gaussian splats from sparse unposed images. In *International Conference on Learning Representations (ICLR)*, 2025. 3, 10
- [55] Vickie Ye, Matias Turkulainen, and the nerfstudio team. gsplat: An open-source library for Gaussian Splatting. *Journal of Machine Learning Research (JMLR)*, 26:1–15, 2025. 2, 3, 5, 7, 8, 9
- [56] Zongxin Ye, Wenyu Li, Sidun Liu, Peng Qiao, and Yong Dou. AbsGS: Recovering fine details for 3D Gaussian Splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5, 12
- [57] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-Splatting: Alias-free 3D Gaussian Splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. Best Student Paper. 12
- [58] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations (ICLR)*, 2025. 3
- [59] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. GS-LRM: Large reconstruction model for 3d Gaussian splatting. In *European Conference on Computer Vision (ECCV)*, 2024. 3
- [60] Zheng Zhang, Wenbo Hu, Yixing Lao, Tong He, and Hengshuang Zhao. Pixel-GS: Density control with pixel-aware gradient for 3D Gaussian Splatting. In *European Conference on Computer Vision (ECCV)*, 2024. 5
- [61] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. FSGS: Real-time few-shot view synthesis using Gaussian Splatting. In *European Conference on Computer Vision (ECCV)*, 2024. 5
- [62] Matthias Zwicker, Hanspeter Pfister, Jeroen Van Baar, and Markus Gross. EWA splatting. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):223–238, 2002. 5