BOUNDED APPEARANCE FIELD FOR IN-THE-WILD 3D GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

ABSTRACT

In unconstrained photo collections, appearance variations and transient occluders violate multi-view consistency, a key assumption of 3D Gaussian Splatting (3DGS). Methods with per-Gaussian appearance features tie spatial appearance capacity to geometric densification, although finer geometry need not require more complex regional appearance. We introduce a Bounded Appearance Field (BAF) that combines a shared-field of fixed capacity with global color adjustments and per-Gaussian local corrections. A fixed-resolution token grid is mixed once per image and queried by Gaussians through trilinear interpolation. This keeps shared field capacity independent of the Gaussian count, while local storage and per-Gaussian computation remain linear in that count. To support adaptation from partial observations of a held-out view, we project color residuals and coverage into the grid, providing spatially aligned appearance evidence for predictions outside the observed region. We additionally use frozen DINOv3 feature distances to supervise a heteroscedastic 2D uncertainty head. Its predicted weights modulate the photometric loss and the gradients driving densification to reduce the influence of transient observations. On both Photo Tourism and NeRF On-the-go datasets, our method converges within 30k updates and matches or surpasses state-of-the-art results. Our code will be available after acceptance.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) enables efficient novel-view synthesis using explicit scene primitives refined through densification. However, unconstrained photo collections challenge its assumption of consistent observations of a static scene. Illumination and camera processing change the color of persistent surfaces, while people and vehicles appear in only some views (Martin-Brualla et al., 2021). Fitting these observations with one static representation averages incompatible colors and even reconstructs transient objects as permanent scene geometry. Since photometric gradients also guide densification, inconsistent observations can affect both color fitting and where geometry grows. Reconstruction therefore requires an appearance model that accommodates changing conditions, together with a mechanism that limits the influence of transient observations on scene optimization and geometry growth.

NeRF-W (Martin-Brualla et al., 2021) models appearance variation with per-image embeddings. WildGaussians (Kulhánek et al., 2024) and Splatfacto-W (Xu et al., 2024a) combine these with learned per-Gaussian features for local corrections. In these representations, densification allocates appearance variables in lockstep with geometry, so their count grows with the Gaussian population N, even though geometric detail and regional appearance complexity are distinct modeling needs. Partial reference observations raise the same organizational question: when adapting to a new appearance, evidence from the observed region should inform predictions outside that region. Primitive-local features alone provide no explicit shared 3D route for this evidence. We therefore ask how regional appearance can be represented with shared capacity independent of densification, while retaining local corrections and transferring partial reference evidence through scene coordinates.

To address this problem, we introduce a Bounded Appearance Field (BAF), the complete appearance module in Figure 1. BAF combines three prediction paths: a bounded shared field produces regional appearance parameters, a global path adjusts each image as a whole, and a per-Gaussian local path provides primitive-specific corrections. The bounded shared field mixes a fixed R^3 token grid once

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.

per image with a Mamba3 (Lahoti et al., 2026) mixer into a 3D field queried by all Gaussians. Densification adds queries while the bounded shared field's token count, parameter count, and per-image mixing workload remain fixed, defining the scope of *bounded*. Projecting observed color residuals and coverage into the same grid provides an explicit 3D route for partial reference evidence to enter appearance prediction. To support learning under transient interference, we build on DINO-based uncertainty weighting (Kulhánek et al., 2024), using frozen DINOv3 feature distances (Siméoni et al., 2025) to supervise a 2D uncertainty head whose weights attenuate unreliable pixels in the reconstruction objective and in the gradients used for densification.

Our contributions are threefold.

- We propose an appearance framework that combines a fixed-capacity shared 3D field with global color adjustments and Gaussian-local corrections, decoupling appearance capacity from densification while keeping local storage and computation linear in the Gaussian count.
- We couple partial reference evidence into the bounded shared field through residual-and-coverage projection, enabling spatially aligned appearance transfer to unobserved regions, and introduce DINOv3-supervised uncertainty weighting for optimization under transient interference.
- Experiments show that our method matches or surpasses state-of-the-art results, and layered ablations examine the contributions of the complete appearance module, the bounded shared field, and reference conditioning.

2 Related Work

Appearance representations. NeRF-W (Martin-Brualla et al., 2021) conditions radiance on per-image embeddings to accommodate appearance changes across photographs. Gaussian-based methods differ in where they represent spatial appearance. WildGaussians (Kulhánek et al., 2024) and Splatfacto-W (Xu et al., 2024a) combine image embeddings with features stored at individual Gaussians to predict spatially varying corrections. GS-W (Zhang et al., 2024) likewise stores appearance features at individual Gaussians, separating per-point intrinsic attributes from per-point dynamic features that capture illumination and weather variation. Other methods obtain spatial features from a shared field. SWAG (Dahmani et al., 2024) uses hash-grid features queried at Gaussian centers, while WilLaGS (Bai et al., 2026) generates appearance triplanes conditioned on a VAE latent. NexusSplats (Tang et al., 2025) attaches a shared embedding to each nexus kernel that manages a group of Gaussians, centralizing appearance learning at an intermediate granularity. Scaffold-GS (Lu et al., 2024) instead organizes Gaussians with sparse anchors, but these anchors adapt to scene content and grow with reconstruction rather than forming a fixed-capacity appearance field.

Reference appearance conditioning. Adapting a scene representation to a new photograph requires inferring its appearance conditions. NeRF-W obtains an image embedding through optimization, whereas Ha-NeRF (Chen et al., 2022) predicts one with a learned image encoder. Other feed-forward approaches infer appearance conditions from reference images (Yang et al., 2023; Wang et al., 2024). Reference images can also provide spatially aligned appearance information: Wild-GS (Xu et al., 2024b) uses rendered depth to lift reference colors into 3D and construct triplanes from which Gaussians query local appearance features.

Transient handling. While appearance modeling accounts for changes on persistent surfaces, transient handling reduces the influence of observations that do not belong to the static scene. Robust-NeRF suppresses distractors with robust losses (Sabour et al., 2023). Gaussian-based alternatives identify distractors through semantic feature clustering or agreement between two models (Sabour et al., 2025; Li et al., 2025). DeSplat (Wang et al., 2025) instead avoids external semantics entirely, initializing per-view distractor Gaussians that are separated from the static scene through alpha compositing. HybridGS (Lin et al., 2025) decouples transients representationally, modeling per-image transient content with 2D Gaussians while 3D Gaussians capture the static scene. NeRF On-the-go (Ren et al., 2024) and WildGaussians use DINO features to estimate uncertainty and downweight inconsistent observations. To limit the influence of transients on geometry growth, RobustSplat++ (Fu et al., 2025) combines delayed Gaussian growth with scale-cascaded mask bootstrapping. All of the

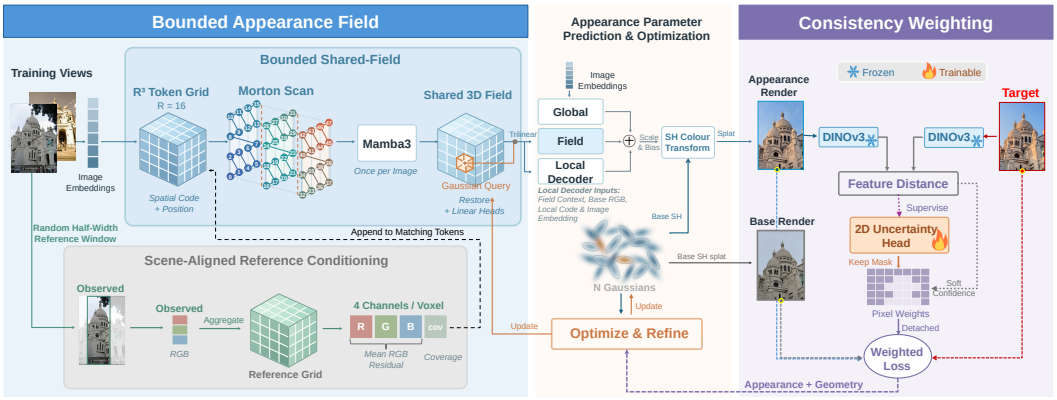

Figure 1: Overview of the proposed method. BAF predicts a bounded affine SH color transform for each Gaussian by summing three prediction paths. Consistency Weighting reduces the influence of transient observations.

above retrain a representation for each scene. DGGS (Bao et al., 2026) instead pursues distractor-free reconstruction in a generalizable feed-forward setting, pairing scene-agnostic reference-based mask prediction during cross-scene training with reference re-selection and distractor-Gaussian pruning at inference to handle unseen scenes.

3 METHOD

Figure 1 demonstrates the pipeline of the proposed method with a Bounded Appearance Field (BAF) module (Section 3.1), an Appearance Parameter Prediction and Optimization module (Section 3.2), and a Consistency weighting module (Section 3.3).

Given a set of input images, we train the standard 3DGS primitives and differentiable rasterizer (Kerbl et al., 2023). The 3D scene contains N Gaussians, where each Gaussian k has center $\mathbf{x}_k \in \mathbb{R}^3$ and RGB SH coefficients $\mathbf{c}_k^{(m)} \in \mathbb{R}^3$, where $m = 0, \ldots, 15$ indexes the SH basis functions. Its covariance and opacity follow standard 3DGS. Each training image $\mathbf{I}_j$ is assigned a learnable appearance embedding $\mathbf{e}_j$ to model its illumination and exposure conditions.

3.1 BOUNDED APPEARANCE FIELD

Bounded Shared-Field. We maintain a single R^3 lattice over a fixed scene box. At lattice site i, a trainable code $\mathbf{s}_i$ stores spatial appearance information and a fixed coordinate $\mathbf{p}_i \in [-1, 1]^3$ identifies its location. These scene-shared quantities are concatenated with $\mathbf{e}_j$ to condition the field on image j:

$$\mathbf{t}_i^{(j)} = [\mathbf{s}_i \| \mathbf{p}_i \| \mathbf{e}_j], \qquad \mathbf{H}^{(j)} = \mathcal{M}\left([\mathbf{t}_i^{(j)}]_{i=1}^{R^3}\right). \tag{1}$$

Here $\|$ denotes concatenation. $\mathcal{M}$ combines a learned input projection with a SISO Mamba3 (Lahoti et al., 2026) mixer, producing the image-conditioned feature grid $\mathbf{H}^{(j)}$. For a fixed image j, we write $\mathbf{H}$ for $\mathbf{H}^{(j)}$ below. We use $R = 16$ and mix tokens in Morton order, then restore their lattice positions for spatial queries. This computation is shared by all Gaussians in an image forward pass and does not grow with N. Only the per-Gaussian codes and local decoding below scale with N. Reference conditioning appends the residual-and-coverage vector $\mathbf{r}_i$ from Equation (2) to each input token.

Scene-aligned reference conditioning. When only part of a reference image is observed, we write its appearance evidence into the same scene-shared 3D lattice. Let $\mathcal{S}_i$ contain the Gaussians whose centers have positive depth, project inside the observed image region, and are assigned to lattice token i. Their mean observed-minus-base color and their normalized count form

$$\mathbf{r}_i = \left[\frac{1}{|\mathcal{S}_i|}\sum_{k\in\mathcal{S}_i}\left(\mathbf{I}(\pi(\mathbf{x}_k)) - \mathbf{b}_k\right) \,\middle\|\, \min\left(\frac{|\mathcal{S}_i|}{8}, 1\right)\right], \tag{2}$$

with $\mathbf{r}_i = \mathbf{0}$ for an empty set. Here π is camera projection and image colors are sampled bilinearly. The first three entries of $\mathbf{r}_i \in \mathbb{R}^4$ describe the observed RGB correction; the last entry measures support, saturating at eight projected Gaussians.

For reference conditioning, the input in Equation (1) becomes $[\mathbf{s}_i \| \mathbf{p}_i \| \mathbf{e}_j \| \mathbf{r}_i]$. Mixing carries this observed appearance evidence into the field queried by the Gaussians. The reconstruction loss trains the mixer and decoders to use this evidence through the affine parameters and rendered image $\mathbf{C}$; reference feature construction is detached.

During training, the reference features come from a full-height window of approximately half the image width at a uniformly random horizontal offset; the window restricts only the reference input, while the full image still provides reconstruction supervision. At test time, we compute $\{\mathbf{r}_i\}$ once using the permitted half-image. We fit the new embedding $\mathbf{e}_j$ on that region with the scene, network parameters, and reference features fixed, recomputing $\mathbf{H}$ at each step. The fitted embedding then renders the scored region.

3.2 Appearance Parameter Prediction and Optimization

Two learned linear heads, $\mathcal{A}$ and $\mathcal{C}$, map $\mathbf{H}$ to a six-channel affine field and a 16-channel context field, respectively. We map Gaussian center $\mathbf{x}_k$ into the scene box's $[-1, 1]^3$ coordinates, denoted $\widetilde{\mathbf{x}}_k$, and query both fields:

$$\begin{aligned} \mathbf{a}_k^{\text{field}} &= 0.01\ \text{Trilerp}(\mathcal{A}(\mathbf{H}), \widetilde{\mathbf{x}}_k), \\ \mathbf{z}_k &= \text{Trilerp}(\mathcal{C}(\mathbf{H}), \widetilde{\mathbf{x}}_k). \end{aligned} \tag{3}$$

Trilerp is the weighted average of the eight lattice vertices surrounding $\widetilde{\mathbf{x}}_k$. The six-vector $\mathbf{a}_k^{\text{field}}$ contributes directly to the Gaussian's color transform, and the context vector $\mathbf{z}_k \in \mathbb{R}^{16}$ enters the local decoder in Equation (4).

The degree-zero SH coefficient gives the Gaussian's base RGB, $\mathbf{b}_k = \kappa_0 \mathbf{c}_k^{(0)} + 0.5$, where $\kappa_0 = 1/\sqrt{4\pi}$. Let $\mathbf{g}_k \in \mathbb{R}^{24}$ be its learned local appearance code. A two-hidden-layer decoder $\mathcal{D}$ combines the base color, local code, field context, and image embedding. A learned linear map $\mathcal{G}$ predicts an image-wide six-vector from $\mathbf{e}_j$. The complete affine parameters are

$$\mathbf{a}_k^{\text{local}} = 0.01\, \mathcal{D}([\mathbf{b}_k \| \mathbf{g}_k \| \mathbf{z}_k], \mathbf{e}_j), \qquad (\boldsymbol{\beta}_k, \boldsymbol{\gamma}_k) = \mathbf{a}_k^{\text{field}} + \mathcal{G}(\mathbf{e}_j) + \mathbf{a}_k^{\text{local}}. \tag{4}$$

Their six outputs are split into an RGB bias $\boldsymbol{\beta}_k$ and an RGB scale increment $\boldsymbol{\gamma}_k$. They transform the SH coefficients channel-wise before rasterization:

$$\widetilde{\mathbf{c}}_k^{(0)} = (1 + \boldsymbol{\gamma}_k)\mathbf{c}_k^{(0)} + \boldsymbol{\beta}_k/\kappa_0, \qquad \widetilde{\mathbf{c}}_k^{(m)} = (1 + \boldsymbol{\gamma}_k)\mathbf{c}_k^{(m)}, \quad m > 0. \tag{5}$$

The affine-output heads start at zero, so training begins from ordinary 3DGS colors. rasterizing $\widetilde{\mathbf{c}}$ produces the appearance-conditioned image $\mathbf{C}$ in Equation (9). Its photometric gradient trains the global and local decoders and passes through both field heads and trilinear interpolation to the mixer and spatial codes. It also updates $\mathbf{e}_j$, $\mathbf{g}_k$, and the base SH coefficients. Interpolation coordinates are held constant during back-propagation; Gaussian geometry receives gradients through rasterization.

3.3 Reconstruction-consistency weighting

Learning an uncertainty map. Let $\mathbf{C}^0$ denote the render produced by the unmodified SH coefficients and let $\mathbf{C}$ denote the appearance-conditioned render from Equation (5). A frozen DINOv3 (Siméoni et al., 2025) encoder F_D compares the target with a detached $\mathbf{C}$ and produces the feature-distance map

$$D^{\text{feat}} = \text{clip}(2 - 2\cos(F_D(\mathbf{I}), F_D(\text{stopgrad}(\mathbf{C}))), 0, 1)\,. \tag{6}$$

Following the heteroscedastic formulation of Kendall & Gal (2017), a trainable 2D head predicts uncertainty from the target-image features, using Softplus and a lower bound of 0.1. We bilinearly resize its output and remove padding to obtain σ^{head} on the head's input-image grid. The distance map D^{feat} is likewise resized and unpadded to the image grid used for feature comparison. Resizing σ^{head} to that grid gives σ^{feat} for the uncertainty loss; resizing it to the original image gives σ^{img} for the reconstruction mask:

$$\mathcal{L}_{\text{unc}} = \text{mean}\left[\frac{D^{\text{feat}}}{2(\sigma^{\text{feat}})^2}\right] + \lambda\, \text{mean}[\log \sigma^{\text{head}}], \qquad w_p^{\text{unc}} = \mathbf{1}\left[\sigma_p^{\text{img}} < \frac{1}{\sqrt{2}}\right]. \tag{7}$$

Here $\mathbf{1}[\cdot]$ is the indicator function and $\lambda = 0.5$. The full-resolution map is thresholded to decide which pixels contribute to reconstruction.

Let D_p be the value at pixel p after bilinearly resizing D^{feat} to the original image. On NeRF On-the-go, we convert it into confidence q_p: it equals one for $D_p \leq \tau$, decreases linearly to zero over $\tau < D_p < \tau + \delta$, and is zero for $D_p \geq \tau + \delta$. We use $\tau = 0.20$ and $\delta = 0.05$, giving the final weight $w_p = w_p^{\text{unc}} q_p$. Photo Tourism uses $w_p = w_p^{\text{unc}}$. These weights suppress high-discrepancy observations in Equation (9). Both masks are linearly introduced from an all-one map during warm-up and held constant during reconstruction back-propagation.

Training objective and densification. With p indexing pixels and c indexing color channels, the total training objective function is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{photo}} + \mathcal{L}_{\text{unc}} \tag{8}$$

where $\mathcal{L}_{\text{unc}}$ trains the uncertainty head. The frozen DINOv3 encoder and stopgrad in Equation (6) keep the feature-distance target fixed during back-propagation.

Let $\text{SSIM}(\mathbf{C}^0, \mathbf{I})_p$ be the per-pixel SSIM score(Wang et al., 2004) computed with an 11×11 window, averaged over RGB channels. The photometric term is determined as L_{photo}:

$$\begin{aligned} \mathcal{L}_{\text{photo}} = \alpha \ \text{mean}_{p,c}[w_p |[\mathbf{C}]_{p,c} - [\mathbf{I}]_{p,c}|] \\ + (1-\alpha) \ \text{mean}_p[w_p (1 - \text{SSIM}(\mathbf{C}^0, \mathbf{I})_p)], \end{aligned} \tag{9}$$

where $\alpha = 0.8$ is the balanced parameter. The first reconstruction term trains the appearance-conditioned render, including the bounded shared field, global and local paths of BAF. The SSIM term acts on the base render and preserves structural supervision for the underlying 3DGS scene. The same w_p multiplies both photometric terms, so it scales each pixel's color and geometry gradients. These include the projected-center gradient magnitudes that 3DGS accumulates to decide splits and duplications, meaning that inconsistent observations contribute less to the statistics that drive densification. Warm-up and opacity-reset protection are specified in Appendix A.5.

4 Experiments

4.1 Datasets and evaluation

We report PSNR, SSIM, and LPIPS (Zhang et al., 201) on three Photo Tourism scenes (Snavely et al., 2006) with appearance variation and six NeRF On-the-go scenes (Ren et al., 2024) with transient occlusions. Photo Tourism uses left-half embedding fitting and right-half scoring following NeRF-W (Martin-Brualla et al., 2021): 128 Adam (Kingma & Ba, 2015) steps from zero at learning rate 0.1, with scene and network parameters fixed and no color correction. NeRF On-the-go uses official splits, full-image scoring, and no reference input or test-time adaptation. Reference-sampling limitations are detailed in Appendix A.5.

4.2 Overall comparison

Photo Tourism. Table 1 shows that Ours improves SSIM and LPIPS over the published 200k-step WildGaussians on all three scenes: PSNR is the highest overall on Brandenburg Gate and within 0.01 dB of the best on Trevi Fountain, while Sacre Coeur trades 0.12 dB PSNR for clear SSIM and LPIPS gains. Gsplat attains slightly lower LPIPS on all three scenes but trails Ours in PSNR by up to 1.7 dB. The efficiency columns sharpen the contrast: WildGaussians is competitive only under its 200k schedule—its 30k variant collapses—whereas Ours reaches its quality within 30k updates, matching the shortest schedule. Its peak memory is also lowest among the methods with reported measurements. Taken together, Ours improves reconstruction fidelity under appearance variation at a fraction of the training cost.

NeRF On-the-go. In Table 2, Ours achieves the best average LPIPS and the second-best average PSNR within 30k updates, the shortest schedule. It leads both metrics on Patio-High, leads LPIPS on Fountain, and nearly ties the PSNR on Spot (24.30 vs. 24.33). Against WildGaussians at the same

Table 1: Photo Tourism. The results of published baselines come from NerfBaselines (Kulhánek & Sattler, 2024), which used NVIDIA A100, except DeSplat (Wang et al., 2025) and 3DGS-HPC (Chen et al., 2026) , which are taken from their original papers. While local memory entries marked † are NVIDIA RTX 5090 peak allocations. The first , second , and third values are highlighted.

Method	Brandenburg Gate			Sacre Coeur			Trevi Fountain			Efficiency	
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	Steps	GPU mem.
NeRF-W (Martin-Brualla et al., 2021)	24.22	0.884	0.213	19.56	0.795	0.260	21.48	0.693	0.331	300k	98.80GiB
WildGaussians (Kulhánek et al., 2024)	27.76	0.927	0.133	22.56	0.859	0.177	23.63	0.766	0.228	200k	18.24GiB
DeSplat (Wang et al., 2025)	25.04	0.920	0.142	20.14	0.868	0.178	23.31	0.775	0.226	200k	–
Scaffold-GS (Lu et al., 2024)	25.45	0.923	0.127	21.85	0.871	0.157	23.21	0.768	0.228	100k	18.34GiB
gsplat (Ye et al., 2025)	26.36	0.931	0.118	22.05	0.876	0.154	22.56	0.765	0.213	100k	4.68GiB
K-Planes (Fridovich-Keil et al., 2023)	23.65	0.859	0.253	19.96	0.762	0.299	19.70	0.662	0.388	30k	3.59GiB
GS-W (Zhang et al., 2024)	24.35	0.903	0.155	19.73	0.824	0.210	20.06	0.724	0.272	30k	21.93GiB
3DGS-HPC (Chen et al., 2026)	20.91	0.878	0.125	17.28	0.833	0.161	17.66	0.724	0.211	30k	–
WildGaussians (Kulhánek et al., 2024)	7.95	0.402	0.689	8.21	0.319	0.661	8.81	0.214	0.605	30k	9.30†GiB
Ours	28.08	0.933	0.124	22.44	0.874	0.162	23.62	0.777	0.217	30k	2.04†GiB

Table 2: NeRF On-the-go. HybridGS (Lin et al., 2025) and NexusSplats (Tang et al., 2025) are from their reported results. The other three baselines are from WildGaussians (Kulhánek et al., 2024). The first , second , and third values are highlighted.

Metric	Method	Steps	Mountain	Fountain	Corner	Patio	Spot	Patio-High	Average
PSNR ↑	NeRF On-the-go (Ren et al., 2024)	250k	20.46	20.79	23.74	20.88	22.80	21.57	21.71
	HybridGS (Lin et al., 2025)	∼41k–93k	21.73	21.11	25.03	21.98	24.33	21.77	22.66
	3DGS (Kerbl et al., 2023)	30k	19.40	19.96	20.90	17.48	20.77	17.29	19.30
	WildGaussians (Kulhánek et al., 2024)	30k	20.43	20.81	24.16	21.44	23.82	22.23	22.15
	NexusSplats (Tang et al., 2025)	30k	20.04	20.59	23.41	20.08	22.64	20.92	21.28
	Ours	30k	21.27	21.22	24.37	20.67	24.30	22.30	22.36
SSIM ↑	NeRF On-the-go (Ren et al., 2024)	250k	0.661	0.661	0.806	0.754	0.800	0.706	0.731
	HybridGS (Lin et al., 2025)	∼41k–93k	0.693	0.674	0.847	0.812	0.794	0.741	0.760
	3DGS (Kerbl et al., 2023)	30k	0.638	0.659	0.713	0.704	0.693	0.604	0.668
	WildGaussians (Kulhánek et al., 2024)	30k	0.653	0.662	0.822	0.800	0.816	0.725	0.746
	NexusSplats (Tang et al., 2025)	30k	0.645	0.649	0.809	0.767	0.762	0.704	0.723
	Ours	30k	0.687	0.686	0.814	0.747	0.757	0.731	0.737
LPIPS ↓	NeRF On-the-go (Ren et al., 2024)	250k	0.186	0.195	0.127	0.133	0.132	0.205	0.163
	HybridGS (Lin et al., 2025)	∼41k–93k	0.284	0.252	0.151	0.169	0.196	0.211	0.211
	3DGS (Kerbl et al., 2023)	30k	0.213	0.185	0.241	0.199	0.316	0.363	0.253
	WildGaussians (Kulhánek et al., 2024)	30k	0.255	0.215	0.045	0.138	0.138	0.206	0.166
	NexusSplats (Tang et al., 2025)	30k	0.287	0.243	0.147	0.169	0.198	0.253	0.216
	Ours	30k	0.203	0.167	0.126	0.134	0.150	0.183	0.161

budget, Ours improves average PSNR and LPIPS ($22.36/0.161$ vs. $22.15/0.166$), while WildGaussians retains an average-SSIM edge driven by Patio and Spot. HybridGS obtains the best average PSNR and SSIM but requires ∼41k–93k steps and yields 31% higher LPIPS. On Mountain, its metric gains over Ours do not translate into cleaner renderings (Figure 3, first row). Overall, Ours delivers the best perceptual fidelity among the six-scene methods at the shortest schedule.

4.3 Ablation Studies and Analysis

Components. Table 3 identifies the global and per-Gaussian paths as the main source of the appearance gains: B is the strongest reference-free configuration on both datasets, while adding the shared-field in C does not improve reference-free quality. On Photo Tourism, adding the partial reference input in D raises all three averages over C. The shared-field therefore matters primarily as the route that carries spatially aligned reference evidence into the full system. Figure 4 shows the qualitative comparison on Photo Tourism. We also use a paired NeRF On-the-go development split to screen Mamba3 (Lahoti et al., 2026) over Diagonal SSM (Gu et al., 2022) in Appendix A.2.



Figure 2: Photo Tourism qualitative comparison. The dashed line separates adaptation and scored halves.

Table 3: Layered appearance ablation. Configuration A disables appearance adaptation, B enables global and per-Gaussian local corrections, C adds the shared-field core, and D adds the reference input. Consistency weighting is enabled in all configurations. All runs share common training settings and matched parameter initialisation, and each is a single fixed-initialisation run evaluated at a specified checkpoint without test-score selection. NeRF On-the-go has no reference input in its protocol, so D does not apply.

Scene	Group	Global/local	Bounded Shared-Field	Reference	PSNR↑	SSIM↑	LPIPS↓
Photo Tourism Average	A	×	×	×	18.654	0.8052	0.2300
	B	✓	×	×	24.523	0.8599	0.1686
	C	✓	✓	×	24.037	0.8587	0.1693
	D	✓	✓	✓	24.712	0.8612	0.1677
NeRF On-the-go Average	A	×	×	×	21.731	0.7270	0.1757
	B	✓	×	×	22.393	0.7375	0.1638
	C	✓	✓	×	22.314	0.7353	0.1632

Frozen backbone and consistency weighting. Keeping BAF and Mamba3 fixed, we replace the frozen feature backbone used to estimate reconstruction consistency. As shown in Table 4, DINOv3 ViT-S/16 is the strongest tested backbone on both benchmarks across all three metrics, motivating its use. We then remove the feature-distance confidence from the uncertainty mask (Table 5). It shows that removing confidence weighting improves development quality but increases the Gaussian

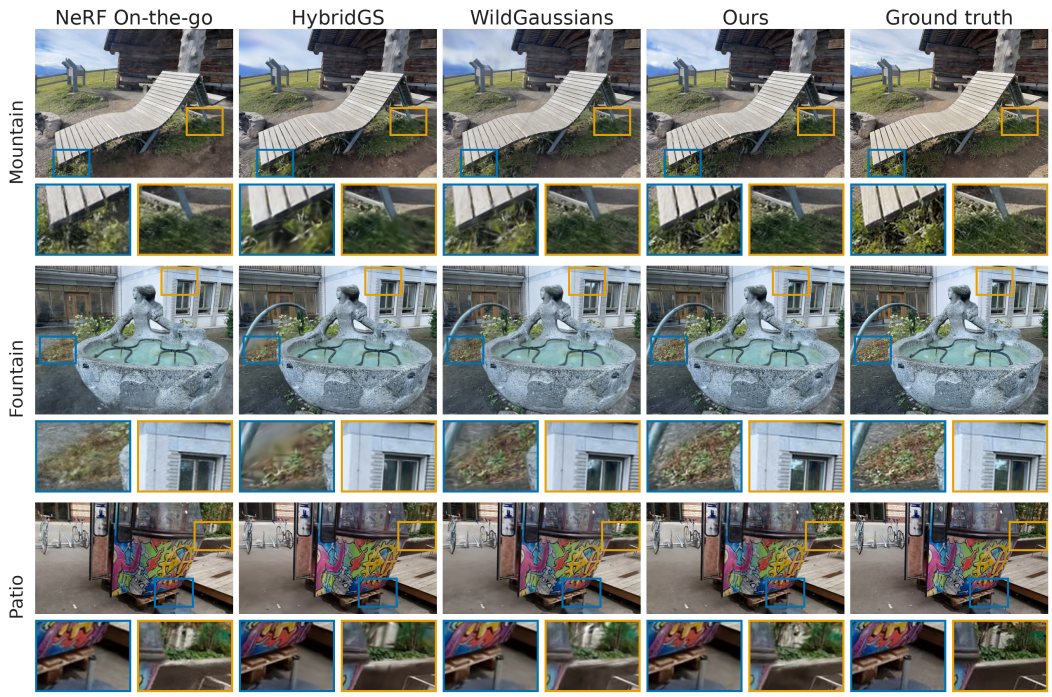

Figure 3: NeRF On-the-go qualitative comparison. Released NeRF On-the-go predictions are rectified to the common camera.

Table 4: Frozen-backbone ablation. BAF and Mamba3 are fixed, all backbone weights are frozen. These separately trained controls use a 16^3 grid and 30k updates; configuration details and missing precision records are described in Appendix A.3.

Backbone	Parameters	Photo Tourism (3 scenes)			NeRF On-the-go (6 scenes)		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
DINOv3 ViT-S/16	21.60M	24.12	0.859	0.168	22.10	0.733	0.165
DINOv2 ViT-S/14	22.06M	22.88	0.850	0.197	19.52	0.613	0.325
DINOv2 ViT-B/14	86.58M	22.53	0.843	0.211	18.50	0.590	0.350

Table 5: Consistency-weighting changes from the fused baseline. Development uses three NeRF On-the-go scenes and official test uses all six scenes. Changes are relative to the fused mask (τ=0.20, δ=0.05). "Learned only" omits distance confidence. ΔN is the Gaussian-count change.

	Setting	ΔPSNR↑	ΔSSIM↑	ΔLPIPS↓	ΔN
Development	Fused (δ=0.10)	+0.053	+0.0035	−0.0015	+1.9%
	Learned only	+0.114	+0.0049	−0.0024	+9.0%
Official test	Learned only	+0.121	+0.0035	−0.0001	+7.6%

population. The official test shows the same quality–compactness trade-off, with little change in LPIPS.

5 Conclusion

We presented BAF, an appearance module for in-the-wild 3D Gaussian Splatting reconstruction, in which a fixed-capacity shared field combines with global and per-Gaussian local paths to predict bounded affine SH color transforms, while DINOv3-based consistency weights attenuate unreliable pixels in both the reconstruction objective and densification. Across Photo Tourism and NeRF



Figure 4: Photo Tourism ablation. A–D use matched views; crops lie in the scored half.

On-the-go, the proposed method improves reconstruction quality over the compared in-the-wild baselines while reducing the training cost: it reaches its quality within 30k updates at the lowest peak memory in the comparison. Ablations attribute the gains predominantly to the global and local paths; the bounded shared field does not consistently improve reference-free reconstruction, and its established role is to carry partial reference evidence, which adds a further average gain.

Limitations and future work. While our bounded shared field keeps the model compact, its fixed capacity may limit the benefit on scenes with stronger appearance variation. Scaling the field under controlled Gaussian budgets is a promising way to improve the quality–cost balance. Additionally, although the reference input provides useful spatial evidence, the sampler does not consider visibility across the observed–scored boundary. Incorporating visibility-aware, crop-confined sampling further strengthen reference conditioning. Finally, existing benchmarks provide no clean-static ground truth, so transient removal is measured only indirectly. Building such targets would enable more direct evaluation. We leave these directions as future work.

AI USE STATEMENT

We used OpenAI Codex (GPT-5-series and GPT-6-series models) to create and edit figures and code, draft and refine manuscript text, and suggest structure, titles, and keywords. It also assisted literature identification, summarisation and analysis, and reference formatting. Kimi Code K3 assisted a subset of these tasks, particularly checking consistency between the manuscript and figures and condensing text. These uses fall within the disclosure-recommended category of the ICLR 2027 policy. The authors reviewed all outputs, read and executed code, checked figures against underlying data, verified references against original sources, and approved all text. We did not use generative AI for tasks requiring disclosure, including synthetic data generation, theoretical or conceptual development, mathematical claims or proofs, or hypothesis formulation or refinement. Nor did we use it for methodology or experimental design or feedback, method implementation, data cleaning or reformatting, or result interpretation. All experimental data underlying the reported results and figures were real measurements obtained by the authors. No experimental results, comparisons, or scientific conclusions were AI-generated. We take full responsibility for the final content, including all AI-assisted work.

REFERENCES

Yuhao Bai, Qianqiu Tan, Lilong Chen, Huanhuan Lv, and Lijun Chen. WilLaGS: Latent-conditional 3D appearance fields for robust gaussian splatting in-the-wild. *arXiv:2608.28240*, 2026.

Yanqi Bao, Jing Liao, Jing Huo, and Yang Gao. Distractor generalizable 3D gaussian splatting. In *ICLR*, 2026.

Jiahao Chen, Yipeng Qin, Ganlong Zhao, Xin Li, Wenping Wang, and Guanbin Li. 3DGS-HPC: Distractor-free 3D Gaussian splatting with hybrid patch-wise classification. In *ICML*, 2026.

Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *CVPR*, 2022.

Hiba Dahmani, Moussab Bennehar, Nathan Piasco, Luis Roldao, and Dzmitry Tsishkou. SWAG: Splatting in the wild images with appearance-conditioned Gaussians. In *ECCV*, 2024.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-Planes: Explicit radiance fields in space, time, and appearance. In *CVPR*, 2023.

Chuanyu Fu, Guanying Chen, Yuqi Zhang, Kunbin Yao, Yuan Xiong, Chuan Huang, Shuguang Cui, Yasuyuki Matsushita, and Xiaochun Cao. RobustSplat++: Decoupling densification, dynamics, and illumination for in-the-wild 3DGS. *arXiv:2512.04815*, 2025.

Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. In *NeurIPS*, 2022.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, 2017.

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

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Jonas Kulhánek and Torsten Sattler. NerfBaselines: Consistent and reproducible evaluation of novel view synthesis methods. *arXiv:2406.17345*, 2024.

Jonas Kulhánek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. WildGaussians: 3D gaussian splatting in the wild. In *NeurIPS*, 2024.

Aakash Sunil Lahoti, Kevin Li, Berlin Chen, Caitlin Wang, Aviv Bick, Zico Kolter, Tri Dao, and Albert Gu. Mamba-3: Improved sequence modeling using state space principles. In *ICLR*, 2026.

Chengqi Li, Zhihao Shi, Yangdi Lu, Wenbo He, and Xiangyu Xu. Robust neural rendering in the wild with asymmetric dual 3D gaussian splatting. In *NeurIPS*, 2025.

Jingyu Lin, Jiaqi Gu, Lubin Fan, Bojian Wu, Yujing Lou, Renjie Chen, Ligang Liu, and Jieping Ye. HybridGS: Decoupling transients and statics with 2D and 3D gaussian splatting. In *CVPR*, pp. 788–797, 2025.

Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-GS: Structured 3D Gaussians for view-adaptive rendering. In *CVPR*, 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 *CVPR*, 2021.

Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Pollefeys, and Songyou Peng. NeRF on-the-go: Exploiting uncertainty for distractor-free NeRFs in the wild. In *CVPR*, 2024.

Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J. Fleet, and Andrea Tagliasacchi. RobustNeRF: Ignoring distractors with robust losses. In *CVPR*, 2023.

Sara Sabour, Lily Goli, George Kopanas, et al. SpotLessSplats: Ignoring distractors in 3D gaussian splatting. *ACM Transactions on Graphics*, 44(2), 2025.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, et al. DINOv3. *arXiv:2508.10104*, 2025.

Noah Snavely, Steven M. Seitz, and ichard Szeliski. Photo tourism: Exploring photo collections in 3D. In *ACM SIGGRAPH*, 2006.

Yuzhou Tang, Dejun Xu, Yongjie Hou, Zhenzhong Wang, and Min Jiang. NexusSplats: Efficient 3D gaussian splatting in the wild. *arXiv:2411.14514*, 2025.

Yihao Wang, Marcus Klasson, Matias Turkulainen, Shuzhe Wang, Juho Kannala, and Arno Solin. DeSplat: Decomposed Gaussian splatting for distractor-free rendering. In *CVPR*, 2025.

Yuze Wang, Junyi Wang, and Yue Qi. WE-GS: An in-the-wild efficient 3D Gaussian representation for unconstrained photo collections. *arXiv:2406.02407*, 2024.

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

Congrong Xu, Justin Kerr, and Angjoo Kanazawa. Splatfacto-W: A nerfstudio implementation of Gaussian splatting for unconstrained photo collections. *arXiv:2407.12306*, 2024a.

Jiacong Xu, Yiqun Mei, and Vishal M. Patel. Wild-GS: Real-time novel view synthesis from unconstrained photo collections. In *NeurIPS*, 2024b.

Yifan Yang, Shuhai Zhang, Zixiong Huang, Yubing Zhang, and Mingkui Tan. Cross-ray neural radiance fields for novel-view synthesis from unconstrained image collections. In *ICCV*, 2023.

Vickie Ye, Ruilong Li, Justin Kerr, et al. gsplat: An open-source library for Gaussian splatting. *Journal of Machine Learning Research*, 2025.

Dongbin Zhang, Chuming Wang, Weitao Wang, Peihao Li, Minghan Qin, and Haoqian Wang. Gaussian in the wild: 3D gaussian splatting for unconstrained image collections. In *ECCV*, 2024.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. Unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 201.

A Additional experimental details and analysis

A.1 Qualitative comparisons and crop provenance

Figures 2 and 3 compare some methods on six fixed views. The NeRF On-the-go panels show Mountain, Fountain, and Corner. Photo Tourism shows two Sacre Coeur views and one Trevi Fountain view. Ours uses the checkpoints corresponding to the reported scene results. The Photo Tourism panels are rendered from saved checkpoints with left-half adaptation; Table 1 reports the recorded terminal-evaluation metrics.

WildGaussians, gsplat, and GS-W images come from their published prediction packages, with gsplat and GS-W matching the NerfBaselines (Kulhánek & Sattler, 2024) configurations in Table 1. The 3DGS images use existing local 30k gsplat baseline runs with appearance optimization, uncertainty, transient masks, and bilateral grids disabled. Their camera parameters and GT pixels match the displayed views. NeRF On-the-go images are uncorrected RGB predictions from the authors' released 250k checkpoints. These two qualitative sources are separate from the published runs reported in Table 2.

The released NeRF On-the-go predictions use distorted image coordinates. We verify image IDs against the official split and rectify predictions using the published radial/tangential calibration and the common pinhole camera. This fixed geometric map uses bilinear resampling, with no parameters fitted to test images. To quantify residual preprocessing differences after camera alignment, we compare independently rectified GT images with the common GT. Their agreement, measured by PSNR, is 32.15, 31.28, and 35.47 dB for Mountain, Fountain, and Corner, respectively. The other methods' GT pixels match exactly.

A.2 Scene-wise appearance ablation and reference conditioning

Table 7 and Table 8 give the scene-level results underlying Table 3. A uses the base SH colors with consistency weighting. B adds the global affine path and the local decoder with 24-dimensional per-Gaussian codes; its field-context input is zero. C adds the 16^3 shared-field and Mamba3. D supplies color residuals and coverage from a half-width reference window, adding four input channels and 256 projection weights at model width 64.

Training and evaluation. Each configuration optimizes the scene and its active network parameters for 30k updates, accumulating four images per update on Photo Tourism and one on NeRF On-the-go. All configurations use DINOv3-based consistency weighting; their learned masks evolve with their respective reconstructions. On Photo Tourism, B–D fit an image embedding for 128 Adam steps on the left half, and all four configurations are scored on the right half. A has no appearance parameters to fit. NeRF On-the-go uses full-image scoring with no reference input or test-time fitting.

Global/local corrections and bounded shared-field. B improves Photo Tourism PSNR over A by 5.58–6.19 dB across the three scenes, with higher SSIM and lower LPIPS throughout. On NeRF On-the-go, B improves PSNR on five of six scenes and lowers LPIPS on all six. These gains establish the importance of image-dependent appearance corrections, particularly under the wider appearance variation in Photo Tourism. Adding the shared-field in C lowers Photo Tourism PSNR on every scene; the largest decrease is 0.995 dB on Sacre Coeur. On NeRF On-the-go, C improves PSNR only on Mountain and Patio-High, by 0.031 and 0.008 dB, while its average LPIPS is 0.0006 lower than B. The shared-field's contribution without reference input is therefore small and metric-dependent on NeRF On-the-go, and unfavorable in Photo Tourism PSNR.

Mixer selection. Table 6 reports paired development runs with two mixers, using a BF16 Mamba3 scan and an FP32 Diagonal SSM scan. Mamba3 has higher two-scene average PSNR, higher SSIM, and lower LPIPS than Diagonal SSM. Therefore, Mamba3 is retained for the reported method.

Reference conditioning. The reported D results exceed C in PSNR and SSIM on all three Photo Tourism scenes. The PSNR differences are 1.260, 0.175, and 0.591 dB for Brandenburg Gate, Sacre Coeur, and Trevi Fountain. LPIPS decreases on Brandenburg Gate and Trevi Fountain and increases slightly on Sacre Coeur, where B remains strongest across all three metrics. The full

Table 6: Mixer screening on NeRF On-the-go. For each scene, development uses a fixed 12.5% holdout of its official training images

Scene	Mixer	PSNR↑	SSIM↑	LPIPS↓
Mountain	Mamba3 (Lahoti et al., 2026)	19.269	0.6388	0.2622
	Diagonal SSM (Gu et al., 2022)	19.145	0.6359	0.2664
Spot	Mamba3 (Lahoti et al., 2026)	16.514	0.6234	0.3691
	Diagonal SSM (Gu et al., 2022)	16.537	0.6240	0.3699
Average	Mamba3 (Lahoti et al., 2026)	17.891	0.6311	0.3157
	Diagonal SSM (Gu et al., 2022)	17.841	0.6299	0.3182
Mamba3 (Lahoti et al., 2026) − Diagonal SSM (Gu et al., 2022)		+0.050	+0.0012	−0.0025

Table 7: Scene-wise appearance ablation on Photo Tourism. Configurations follow Table 3: A uses base SH colours, B adds global/local corrections, C adds the shared core, and D adds reference input. All retain consistency weighting.

	Brandenburg Gate			Sacre Coeur			Trevi Fountain		
Group	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
A	21.008	0.8860	0.1708	17.069	0.8215	0.2351	17.884	0.7081	0.2842
B	26.856	0.9326	0.1251	23.255	0.8766	0.1569	23.459	0.7706	0.2238
C	26.822	0.9309	0.1248	22.260	0.8736	0.1610	23.030	0.7715	0.2222
D	28.082	0.9332	0.1243	22.435	0.8738	0.1615	23.621	0.7767	0.2174

Table 8: Scene-wise appearance ablation on NeRF On-the-go. Configurations follow Table 3: A uses base SH colours, B adds global/local corrections, and C adds the shared core. All retain consistency weighting. Averages are computed from the displayed scenes.

Metric	Group	Mountain	Fountain	Corner	Patio	Spot	Patio-High
PSNR ↑	A	20.193	21.220	23.028	20.448	23.733	21.763
	B	21.319	21.173	24.188	20.830	24.613	22.236
	C	21.349	21.058	24.167	20.707	24.361	22.244
SSIM ↑	A	0.6678	0.6838	0.7831	0.7461	0.7518	0.7292
	B	0.6891	0.6846	0.8078	0.7598	0.7567	0.7268
	C	0.6888	0.6819	0.8072	0.7511	0.7551	0.7279
LPIPS ↓	A	0.2262	0.1701	0.1656	0.1387	0.1619	0.1918
	B	0.2066	0.1648	0.1359	0.1335	0.1522	0.1896
	C	0.2032	0.1665	0.1337	0.1348	0.1522	0.1887

system's advantage is therefore concentrated in Brandenburg Gate and Trevi Fountain. The shared grid provides the route for reference evidence in this configuration; isolating the benefit of spatial correspondence requires matched run selection, crop-confined sampling, and equal-capacity reference controls.

Figures 4 and 5 use the same views and crops as the full-system comparisons. In the NeRF On-the-go panels, B and C give similar local structure, consistent with their small scene-average differences. The Fountain view also shows the limited benefit of appearance adaptation on this scene.

A.3 Frozen-backbone sensitivity and grid resolution

Table 4 uses a separate set of 30k runs with BAF, Mamba3, and a 16^3 grid. Reference conditioning is enabled on Photo Tourism and disabled on NeRF On-the-go. The saved Photo Tourism DINOv3 control configurations omit the local-decoder precision and TF32 setting, whereas the main-result configurations specify FP32 and disabled TF32.

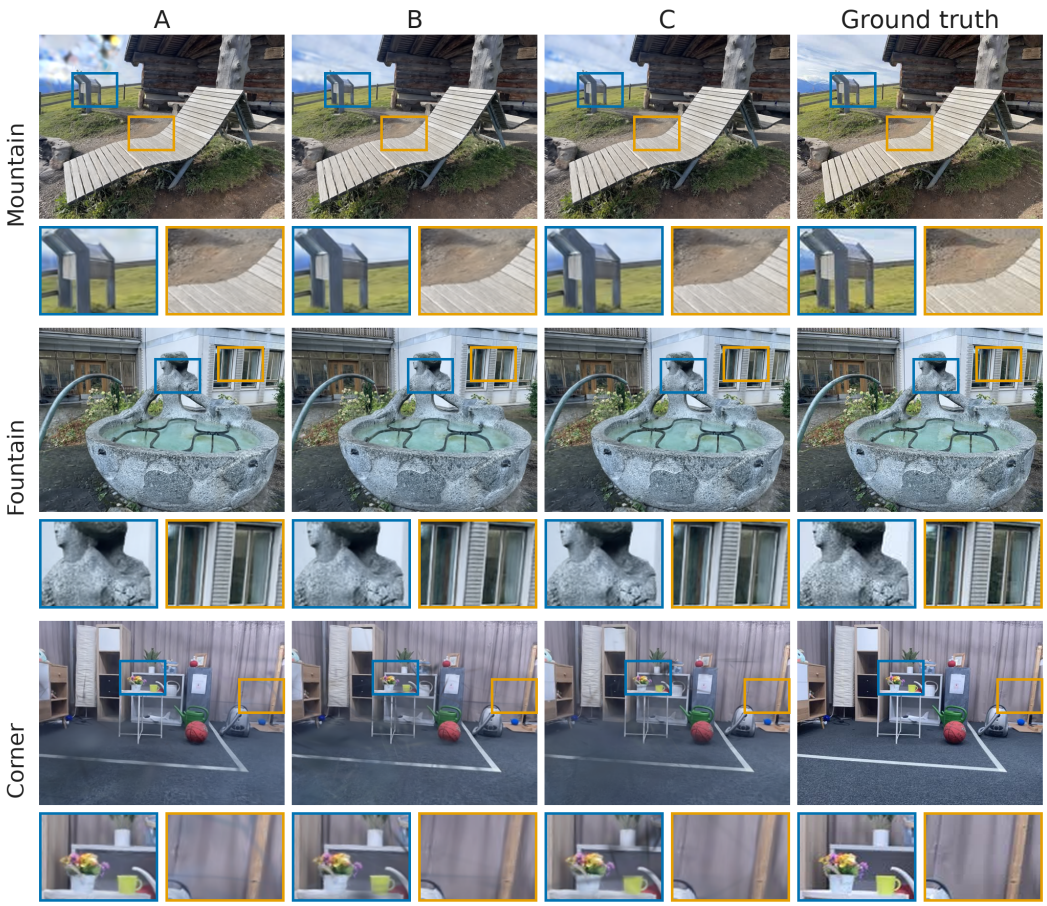

Figure 5: NeRF On-the-go appearance ablation. A uses base SH colors, B adds global and per-Gaussian local corrections, and C adds the shared-field. All configurations retain consistency weighting and use the same views, crops, and enlargement scale. Rows show Mountain, Fountain, and Corner; evaluation uses full images without reference input or test-time adaptation.

Table 9: Scene-wise backbone differences by occlusion level on NeRF On-the-go. Values are DINOv3 ViT-S/16 minus DINOv2 ViT-B/14. Scenes follow the dataset's reported occlusion groups: low (Mountain, Fountain), medium (Corner, Patio), and high (Spot, Patio-High).

Scene	ΔPSNR↑	ΔSSIM↑	ΔLPIPS↓
Mountain	+5.91	+0.149	−0.141
Fountain	+0.95	+0.058	−0.078
Corner	+0.99	+0.033	−0.054
Patio	+2.96	+0.147	−0.119
Spot	+6.00	+0.234	−0.425
Patio-High	+4.78	+0.236	−0.294

Backbone sensitivity. All backbone weights are frozen. DINOv2 ViT-B/14 has approximately four times as many parameters as ViT-S/14, yet lower PSNR in Table 4. Capacity alone therefore does not explain the observed ordering. The substitutions jointly vary pretraining, architecture, and patch size. Table 9 decomposes the DINOv3 ViT-S/16–DINOv2 ViT-B/14 difference by scene. DINOv3 improves all three metrics on every scene, with the largest group-average PSNR gain in the high-occlusion group. Within the low-occlusion group, Mountain has the second-largest PSNR gain overall and Fountain the smallest. Occlusion level thus captures only part of the scene variation in backbone sensitivity.

Grid resolution. To examine how spatial resolution affects reconstruction, a separate Photo Tourism experiment tests $R = 16, 24, 32$ at 30k updates, with reference conditioning enabled and the local decoder in bfloat16 for all three resolutions. Each scan window spans all R^3 tokens. Its $R = 16$ baseline therefore uses a different decoder precision from the main runs. On Sacre Coeur, PSNR is $21.67/22.08/22.32$ dB and LPIPS is $0.1682/0.1604/0.1598$ at the three resolutions. Increasing R from 16 to 32 raises Trevi Fountain PSNR by 0.15 dB and lowers Brandenburg Gate PSNR by 0.15 dB. The quality response to grid resolution thus varies by scene.

A.4 Consistency weighting and densification diagnostics

For consistency weighting and densification, we formed a fixed development split from official training lists: a 12.5% holdout with at least eight images, yielding $104/15$, $86/12$, and $147/21$ train/development images for Mountain, Patio, and Spot. Official test images are excluded. Candidate variants were screened on development data before official evaluation.

Table 5 reports changes relative to the fused mask ($\tau = 0.20$, $\delta = 0.05$), using paired comparisons on each development scene. Both development variants slightly improve the reconstruction metrics while increasing the Gaussian population. On the official six-scene evaluation, the learned-only variant has per-scene PSNR changes from -0.369 to $+0.578$ dB. Across these comparisons, the fused mask trades average reconstruction quality for a smaller Gaussian population, with scene-dependent effects on PSNR.

A.5 Implementation details

Field construction and evaluation. On the first appearance or reference-field call, we form the scene box from the 1st and 99th percentiles of the Gaussian centers, expand each span by 5%, and fix the bounds for subsequent densification. Degenerate spans are clamped to 10^{-6}. Lattice codes use $F = 2$ Fourier frequencies per axis ($d_s = 12$). Each coordinate is centered by the lattice mean, divided by twice its 97th-percentile absolute deviation (clamped to 10^{-6}), and shifted by $1/2$. For each axis, sine features precede cosine features. Local Gaussian features use the same initialization rule before optimization.

Tokens are Morton-sorted using 10-bit coordinate quantization, with x as the least significant interleaved axis. SISO Mamba3 uses model width 64, state dimension 16, head dimension 64, and chunk size 64. Each 4096-token window starts with a fresh state and is zero-padded when incomplete. We restore lattice order before interpolation. The scan uses bfloat16 inputs; the local decoder and interpolation use FP32. We query the field by trilinear interpolation at detached Gaussian centers. The scene-box bounds coincide with the outermost lattice sites, and query coordinates are clamped to the box.

Reference sampling. Training uses a full-height reference window covering approximately half the image width, with a uniformly sampled horizontal position. At test time, the left half supplies reference pixels and embedding-fitting supervision, and the right half is scored. For odd widths, the extra column belongs to the left half.

Gaussians at positive depth whose centers project into the reference window contribute bilinearly sampled color residuals to their nearest lattice sites, with positions clamped to the scene box. Centers and base colors are detached. Residuals and coverage follow Equation (2), with unclipped residuals and zero values at empty sites. Sampling uses border padding without additional visibility, uncertainty, or sky filtering. Bilinear sampling accesses the full image after center selection, so pixels across the reference boundary can enter the sampled colors.

Consistency estimation and mask scheduling. The frozen DINOv3 backbone runs in bfloat16 and evaluation mode. Inputs use the pretrained channel normalization and are center-padded to the patch grid. We discard class and register tokens and crop interpolated outputs to remove padding. Target features are cached by image index; features of detached renders are recomputed. Feature-distance inputs have a longest-side limit of 350 pixels on both benchmarks. The uncertainty head uses the original Photo Tourism resolution and the same 350-pixel limit on NeRF On-the-go.

The 2D head applies Dropout2d ($p = 0.5$), BatchNorm2d, and a zero-initialized 1×1 convolution to target features. An offset $\log(\exp(1)-1)$ before Softplus and a lower clamp of 0.1 initialize its output to one. Image-space maps are resized bilinearly by aligning the outer edges of the input and output images. The learned keep mask is warmed from one to its final value over the intervals in Table 11. On NeRF On-the-go, feature-distance confidence is warmed separately before multiplication with the learned mask; a zero-width distance transition uses $\mathbf{1}[D < \tau]$. For 500 steps at startup and after each opacity reset, the uncertainty loss is detached and feature-distance confidence is skipped, with the learned keep mask retained. During Photo Tourism test fitting, the fixed learned mask weights both L1 and SSIM on the appearance-transformed render.

Scene initialization and optimization. Both presets use antialiased rasterization, near/far planes 0.01 and 10^{10}, and disable EXIF exposure loading and random backgrounds. Non-sky Gaussians are initialized from SfM points with opacity 0.1; their stored log-scales are the logarithms of the RMS distances to the three nearest neighbors. Photo Tourism uses RGB-to-SH initialization and random quaternions. NeRF On-the-go stores RGB directly in the DC slot, clamps SH coefficients above one at render time, and uses identity quaternions. Image embeddings use independent normal initialization with standard deviation 0.01.

Table 10 lists the learning rates. The mean rate is multiplied by the trainer's scene-scale factor and decays per step by $0.01^{1/30000}$; the other Gaussian rates are constant. Gaussian groups use fused Adam with $\beta = (0.9, 0.999)$ and $\epsilon = 10^{-15}$; other groups use Adam defaults. Weight decay is zero for all groups. Each image loss is divided by the number of images accumulated per update before back-propagation. Adam then updates the parameters, and refinement uses the recorded projected-center gradients.

Table 10: Initial learning rates. Shared entries apply to both benchmarks. The Gaussian-mean rate is additionally scaled by scene size and decayed during training.

Parameter group	Photo Tourism	NeRF On-the-go
Gaussian means	1.6×10^{-5}	1.6×10^{-4}
Gaussian scales	5×10^{-4}	5×10^{-3}
Gaussian opacity	5×10^{-2}	
Gaussian rotation	10^{-3}	
DC SH coefficients	2.5×10^{-3}	
Higher-order SH coefficients	1.25×10^{-4}	
Image embeddings	3×10^{-3}	
Appearance-network parameters	5×10^{-4}	
Local Gaussian features	5×10^{-3}	
Uncertainty head	10^{-3}	

Densification and sky handling. Table 11 summarizes the training schedules. Photo Tourism uses absolute 2D gradients for densification; NeRF On-the-go uses the signed/absolute-gradient union of WildGaussians (Kulhánek et al., 2024). Both use opacity pruning at 0.005, 3D growth/pruning thresholds $0.01/0.1$, and 2D growth/pruning thresholds $0.05/0.15$, with screen-size pruning disabled. On Photo Tourism, dividing each image loss by $A = 4$ also scales the accumulated 2D gradients; the densification threshold is applied without compensation.

The NeRF On-the-go Mip filter derives each Gaussian's smoothing scale from its minimum valid training-camera depth and the maximum horizontal focal length. Valid projections have depth above 0.2 and lie within the image bounds extended by 15% on each side. The filter broadens each Gaussian and compensates its opacity for the increased volume. It is updated every 100 steps and after densification, and is disabled on Photo Tourism.

Following WildGaussians (Kulhánek et al., 2024), sky initialization samples $100{,}000$ Fibonacci points around the initial point-cloud mean at ten times the 97th-percentile radius. A point is retained if its projected center lies in the upper two-thirds of at least one training view and passes the positive-depth and image-boundary gates. Sky Gaussians use RGB $(237, 247, 252)/255$ and opacity 0.999999, are excluded from growth and pruning, and have center/scale/rotation gradients zeroed. At each opacity

Table 11: Training schedules. Steps count optimizer updates. Opacity resets occur within the refinement period.

Setting	Photo Tourism	NeRF On-the-go
Images accumulated per update	4	1
SH-degree increase interval	150	1000
Mask warm-up interval	5000–10000	2000–10000
Opacity-reset interval	2250	3000
Densification window	600–15000	500–15000
Densification interval	60	100
Densification gradient threshold	2×10^{-4}	3×10^{-4}

reset, Photo Tourism restores their saved opacity logits for entries whose pre-reset opacity exceeds 0.5. NeRF On-the-go uses no such restoration.

A.6 Training records and resource measurements

Table 12: Photo Tourism reconstruction quality and training resources. Metrics average scored-half scene results equally. [a]Local WildGaussians (Kulhánek et al., 2024) with the official 200k schedule stopped at 30k; [b]NerfBaselines (Kulhánek & Sattler, 2024) report; [c]Ours from Table 1.

Method	Updates	PSNR↑	SSIM↑	LPIPS↓	Time	GPU mem.	GPU
WildGaussians[a]	30k	8.320	0.3118	0.6518	71.98min	9.30GiB	NVIDIA RTX 5090
WildGaussians[b]	200k	24.650	0.8507	0.1793	618.27min	18.24GiB	NVIDIA A100
Ours[c]	30k	24.712	0.8612	0.1677	133.18min	2.04GiB	NVIDIA RTX 5090

WildGaussians training schedule. The local 30k WildGaussians runs in Table 12 use NerfBaselines (Kulhánek & Sattler, 2024) v1.2.12 and the strict NeRF-W evaluator. They retain the official 200k learning-rate horizon, 100k densification endpoint, 15k opacity-reset interval, and 35k uncertainty warm-up onset. Terminal evaluation at 30k therefore coincides with an opacity reset and precedes uncertainty warm-up.

Reconstruction quality and resource use. Ours achieves comparable PSNR and better SSIM and LPIPS than the published 200k WildGaussians result within 30k updates. Its reported GPU memory usage is 2.04 GiB, compared with 9.30 GiB for local WildGaussians at 30k and 18.24 GiB for the published 200k configuration. At the same 30k update budget, Ours improves all three reconstruction metrics over local WildGaussians, with a longer recorded training time (133.18 versus 71.98 minutes).

Measurement conditions. The local 30k entries use NVIDIA RTX 5090, and the published 200k WildGaussians entry uses NVIDIA A100. Local WildGaussians time sums training iterations and excludes separate evaluation. Our logs measure wall-clock time from the training-loop start to the terminal save; the published 200k time comes from NerfBaselines (Kulhánek & Sattler, 2024).