PORTGS: PAIRED OPACITY REALLOCATION FOR TRANSIENT-FREE 3D GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recovering static scenes from casually captured multi-view images with transient distractors remains challenging for 3D Gaussian Splatting (3DGS), as transient objects introduce cross-view inconsistencies while occluding static content, leaving the available static supervision incomplete. Existing transient-free 3DGS methods mainly suppress unreliable observations, disentangle scene components, or stabilize reconstruction. However, even after transient-aware training has established a reasonably reliable static representation, occlusion-induced missing evidence can leave it under-constrained, making further refinement from reliable but incomplete observations a distinct representation-level challenge. To address this problem, we propose PortGS, which coordinates representation retention and refinement through capacity-participation decoupling, expanding learnable capacity without automatically granting it independent rendering participation. PortGS organizes the scene into persistent base-plastic Gaussian pairs, with base primitives retaining established scene parameters and plastic primitives providing additional geometric and appearance capacity. We further introduce base-conditioned paired opacity reallocation, in which a trainable allocation variable partitions the base opacity reference between the paired primitives. This makes plastic rendering participation relative to the retained base rather than independently parameterized. Base and plastic are then jointly rasterized as a unified static representation under the inherited transient-aware supervision. Experiments on *RobustNeRF* and *NeRF On-the-go* datasets demonstrate the effectiveness of PortGS across both benchmarks and multiple transient-aware 3DGS baselines.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) has emerged as an efficient representation for high-quality novel-view synthesis. However, vanilla 3DGS fits all training observations with a single static scene representation, making it vulnerable to casually captured image collections containing transient distractors (Tang et al., 2025). Such distractors, including pedestrians and vehicles, introduce view-specific observations while occluding the static content that should ultimately be reconstructed (Ren et al., 2024). Consequently, transient-free 3DGS must address two distinct but coupled challenges: suppressing cross-view inconsistent supervision and recovering static content from evidence that remains incomplete due to transient occlusion (Yang et al., 2026).

Existing transient-free methods, spanning both Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) and 3DGS, have improved reconstruction robustness by addressing unreliable observations, static-transient entanglement, and unstable optimization dynamics. Robust-supervision methods (Martin-Brualla et al., 2021; Kulhanek et al., 2024; Sabour et al., 2025) use uncertainty modeling, learned reliability estimation, or feature-based masking to reduce the influence of view-inconsistent observations. Scene-decomposition methods (Wu et al., 2022; Wang et al., 2025b;a) assign persistent static content and view-specific transients to different scene variables. More recent optimization-oriented approaches (Ungermann et al., 2024; Du et al., 2026) further improve static reconstruction through densification control, progressive filtering and recovery, or cross-model consistency. Taken together, these methods substantially improve how a static representation is formed and protected from transient interference. A different challenge arises after transient-aware training has already established a reasonably reliable static representation. Static evidence may remain

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.

incomplete due to occlusion, leaving the established representation in need of further refinement without unnecessarily disrupting what has already been recovered.

This question exposes two straightforward but imperfect refinement strategies. One solution is to continue optimizing the established Gaussians in place. In this case, however, the same parameters are simultaneously responsible for retaining the established scene estimate and absorbing subsequent refinement, entangling retention and refinement within a single representation. An alternative is to introduce additional Gaussians to provide extra geometric and appearance capacity. However, if these new primitives are assigned independent opacity, increasing learnable capacity automatically introduces an independent route for rendering participation. Thus, further refinement requires not only additional capacity, but also an explicit organization of how the established representation is retained and how the added capacity is allowed to participate.

Based on this observation, we formulate transient-free static refinement through a capacity-participation decoupling principle. The static representation should gain additional geometric and appearance capacity without automatically granting that capacity independent rendering participation. We instantiate this principle in PortGS, a paired-Gaussian refinement framework that separates retention from refinement through persistent base-plastic Gaussian pairs. Base primitives retain the established scene parameters, whereas plastic primitives provide independently trainable geometry and appearance for subsequent refinement. The persistent pairing defines the base reference used to coordinate rendering participation through paired opacity reallocation.

To regulate how the added capacity participates in rendering, PortGS introduces base-conditioned paired opacity reallocation. Instead of assigning plastic an independent primitive opacity, a learnable allocation variable partitions the base opacity reference between the paired primitives. The base reference specifies the pairwise opacity available for allocation, while the allocation variable determines how this reference is divided between base and plastic, making refinement participation relative to the retained representation rather than independently additive. Both primitives are jointly rasterized as a unified static representation under the inherited transient-aware supervision, without requiring an additional transient-prediction module. Our main contributions are as follows:

1. We formulate the refinement of transient-free static reconstruction as a representation-level coordination problem, and introduce capacity-participation decoupling to separate learnable-capacity expansion from independent rendering participation.
2. We propose persistent base-plastic Gaussian pairs that separate retention of established scene parameters from independently learnable geometric and appearance refinement capacity, organizing both roles within a unified static representation.
3. We develop base-conditioned paired opacity reallocation, where a trainable allocation fraction partitions the base opacity reference, making refinement participation relative to the retained representation rather than independently parameterized.

2 Related Work

2.1 Novel View Synthesis

Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) represent scenes as continuous radiance fields, with subsequent work improving rendering fidelity (Barron et al., 2021; Huang et al., 2022), efficiency (Müller et al.; Chen et al., 2022a), and reconstruction under less constrained capture settings (Lin et al., 2021; Yu et al., 2021; Wang et al., 2021; Liu et al., 2026). 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) later introduced an explicit Gaussian representation with differentiable rasterization for high-quality real-time rendering, inspiring extensions in rendering quality (Yu et al., 2024; Lu et al., 2024), dynamic reconstruction (Wu et al., 2024; Zhao et al., 2025), and 3D generative modeling (Tang et al., 2024). PortGS builds on 3DGS to refine static representations under transient occlusion and incomplete static evidence.

2.2 Transient-Free Reconstruction

Handling Unreliable Observations. Prior work improves transient-free reconstruction by modeling appearance variation and suppressing unreliable observations. NeRF-W (Martin-Brualla et al.,

2021) introduced per-image appearance embeddings and uncertainty-aware transient modeling, followed by view-dependent and cross-view modeling (Chen et al., 2022b; Yang et al., 2023; Li et al., 2023). In 3DGS, SWAG (Dahmani et al., 2024), GS-W (Zhang et al., 2024), and WildGaussians (Kulhanek et al., 2024) extend in-the-wild reconstruction through appearance adaptation and transient-aware filtering (Xu et al., 2024a; Wang et al., 2024b; Xu et al., 2024b), while feed-forward approaches move beyond per-scene optimization (Furutani et al., 2026; Pan et al., 2026). Robust-NeRF (Sabour et al., 2023) established residual-driven robust estimation, with later methods incorporating geometric or object-level priors (Chen et al., 2024; Ungermann et al., 2024). NeRF On-the-go (Ren et al., 2024) and SpotLessSplats (Sabour et al., 2025) further use pretrained semantic or perceptual features (Oquab et al., 2023), while recent work explores adaptive reliability estimation (Markin et al., 2024; Liang et al., 2026; Park et al., 2026a), granularity-aware classification (Chen et al., 2026), supervision recovery (Kong et al., 2025; Shen et al., 2026), and cross-view constraints (Hui et al., 2026; Seo et al., 2026; Seo & Heo, 2026; Yang et al., 2026). PortGS is complementary to these methods. It retains the underlying transient-aware supervision and instead refines an established static representation under incomplete static evidence.

Static-Transient Disentanglement. Explicit representation factorization provides another approach to static-transient entanglement. D^{2}NeRF (Wu et al., 2022) decomposes dynamic scenes into static and time-varying radiance fields, while DeSplat (Wang et al., 2025b) separates globally shared static Gaussians from per-view distractor Gaussians and HybridGS (Lin et al., 2025) models transients with view-specific 2D Gaussians. DeGauss (Wang et al., 2025a) and ForestSplats (Park et al., 2026b) further introduce dynamic-foreground or deformable-transient representations. In contrast, we operate on an established static representation, separating retention from newly introduced refinement capacity rather than explicitly decomposing static and transient content.

Optimization Dynamics and Static Refinement. Another line of work regulates supervision, Gaussian structure, or representation capacity during training. MemE (Wang et al., 2024a) exploits the memorization dynamics of consistent and distractor observations, while RobustSplat (Fu et al., 2025b) delays densification until static structure and transient masks become more reliable. RobustSplat++ (Fu et al., 2025a), curriculum-based distractor modeling (Li & Cheung, 2025), and CleanSplat (Liang et al., 2026) further coordinate optimization stages with appearance modeling, mask evolution, pruning, or structural growth. DualSplat (Wang et al., 2026) and PDF-GS (Seo et al., 2026) adopt successive filtering and refinement stages, while AsymGS (Li et al., 2025) uses cross-model consistency to suppress unstable artifacts. Beyond transient-free reconstruction, MVGS (Du et al., 2026) regulates multi-view gradients, and LapisGS (Shi et al., 2025) progressively adds enhancement Gaussians over retained layers. We instead focus on how added refinement capacity participates relative to an established representation through persistent base-plastic pairing and base-conditioned paired opacity reallocation.

3 METHOD

Given posed multi-view images $\mathcal{D} = (I_v, \mathcal{C}_v)_{v=1}^{V}$ containing transient distractors, where $\mathcal{C}_v$ denotes the camera parameters of view v, together with an established static Gaussian state $\mathcal{G}^a$ reached during transient-aware training, PortGS continues refinement from this state. Each Gaussian in this state is retained as a base primitive and augmented with a persistent plastic counterpart to form a base-plastic pair. Base retains the established scene parameters, while plastic provides independently trainable geometry and appearance for further refinement. A learnable pairwise allocation fraction controls how the added plastic capacity participates in rendering through base-conditioned paired opacity reallocation. Base and plastic are jointly rasterized as a unified static representation using the 3DGS renderer and the supervision of the underlying transient-aware method.

3.1 PRELIMINARIES

Given posed multi-view images, 3D Gaussian Splatting (3DGS) represents a scene as a set of anisotropic Gaussian primitives $\mathcal{G} = \{g_i\}_{i=1}^{N}$ (Kerbl et al., 2023), where $g_i = (\boldsymbol{\mu}_i, \mathbf{s}_i, \mathbf{q}_i, \alpha_i, \mathbf{h}_i)$ denotes its center, scale, rotation, primitive opacity, and spherical-harmonic (SH) appearance coefficients, respectively. Let $\mathbf{S}_i = \mathrm{diag}(\mathbf{s}_i)$ and $\mathbf{R}_i = \mathbf{R}(\mathbf{q}_i)$. The 3D covariance is $\boldsymbol{\Sigma}_i = \mathbf{R}_i\mathbf{S}_i\mathbf{S}_i^\top\mathbf{R}_i^\top$, which is projected to screen space as $\boldsymbol{\Sigma}_i' = \mathbf{J}_i\mathbf{W}\boldsymbol{\Sigma}_i\mathbf{W}^\top\mathbf{J}_i^\top$, where $\mathbf{W}$ is the world-to-camera transformation and $\mathbf{J}_i$ is the projection Jacobian.

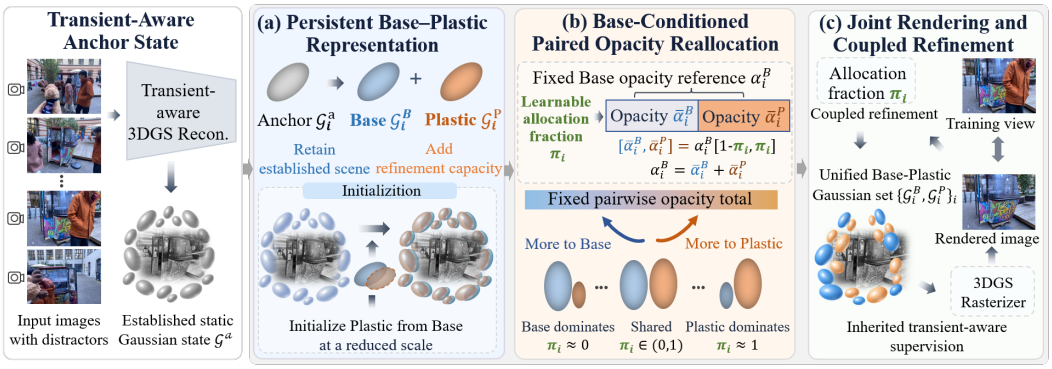


Figure 1: Overview of PortGS. Starting from an established static Gaussian state obtained by an underlying transient-aware 3DGS method (left), PortGS proceeds through three coordinated stages. (a) Each anchor Gaussian is retained as a base primitive and paired with a plastic primitive initialized from the base at a reduced scale, separating retention from refinement capacity. (b) A learnable allocation fraction π_i partitions the fixed base opacity reference α_i^B into the effective primitive opacities $\bar{\alpha}_i^B$ and $\bar{\alpha}_i^P$ while preserving the pairwise opacity total. (c) Base and plastic primitives are jointly rasterized with the standard 3DGS renderer under the inherited transient-aware supervision, where the shared allocation fraction induces coupled refinement.

Since PortGS reallocates primitive opacity before rasterization, we distinguish primitive opacity from the pixel-wise alpha used for compositing. Given projected mean $\boldsymbol{\mu}_i'$, the Gaussian footprint is $G_i(\mathbf{x}) = \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i')^\top(\boldsymbol{\Sigma}_i')^{-1}(\mathbf{x}-\boldsymbol{\mu}_i')\right]$, yielding $\tilde{\alpha}_i(\mathbf{x}) = \alpha_i G_i(\mathbf{x})$. Standard front-to-back compositing gives

$$\hat{\mathbf{C}}(\mathbf{x}) = \sum_{i\in\mathcal{N}(\mathbf{x})} T_i(\mathbf{x})\tilde{\alpha}_i(\mathbf{x})\mathbf{c}_i(\mathbf{d}), \qquad T_i(\mathbf{x}) = \prod_{j<i}\left(1-\tilde{\alpha}_j(\mathbf{x})\right), \tag{1}$$

where $\mathcal{N}(\mathbf{x})$ contains the depth-ordered contributing Gaussians and $\mathbf{c}_i(\mathbf{d})$ is the view-dependent color. PortGS preserves this standard projection and compositing formulation, modifying only the primitive-opacity allocation within each base-plastic pair before rasterization.

3.2 Persistent Base-Plastic Representation

We instantiate the established state $\mathcal{G}^a$ as the base representation $\mathcal{G}^B = \mathcal{G}^a = \{g_i^B\}_{i=1}^N$, where each $g_i^B = (\boldsymbol{\mu}_i^B, \mathbf{s}_i^B, \mathbf{q}_i^B, \alpha_i^B, \mathbf{h}_i^B)$ follows the Gaussian parameterization in Sec. 3.1. All base parameters are frozen during PortGS refinement, preserving the geometry, appearance, and primitive opacity encoded by the established state. In particular, α_i^B remains fixed and serves as the base opacity reference for the paired reallocation introduced in Sec. 3.3. Freezing the base parameters, however, does not fix the effective rendering contribution of the base primitives, because their effective opacities supplied to the rasterizer are determined through paired opacity reallocation.

For each base Gaussian g_i^B, we introduce a plastic counterpart with independently trainable geometry and appearance parameters, together with an associated allocation variable. Specifically, we define

$$g_i^P = \left(\boldsymbol{\mu}_i^P, \mathbf{s}_i^P, \mathbf{q}_i^P, \pi_i, \mathbf{h}_i^P\right), \tag{2}$$

where $\pi_i \in [0,1]$ is the learnable allocation fraction associated with the base-plastic pair. The plastic primitives form the set $\mathcal{G}^P = \{g_i^P\}_{i=1}^N$. Each plastic primitive inherits the position, rotation, and SH appearance of its paired base at initialization and is assigned a reduced initial scale:

$$\boldsymbol{\mu}_i^{P,0} = \boldsymbol{\mu}_i^B, \qquad \mathbf{s}_i^{P,0} = \kappa\,\mathbf{s}_i^B, \qquad \mathbf{q}_i^{P,0} = \mathbf{q}_i^B, \qquad \mathbf{h}_i^{P,0} = \mathbf{h}_i^B, \tag{3}$$

where $\kappa \in (0,1)$ controls the initial plastic scale. We initialize the allocation fraction as $\pi_i^0 = 0$, so introducing plastic capacity leaves the established rendering unchanged at insertion. Thereafter, plastic geometry, appearance, and the pairwise allocation fraction are optimized while all base parameters remain fixed.

The base-plastic correspondence $\mathcal{P} = (g_i^B, g_i^P)_{i=1}^N$ provides the default one-to-one indexing relation used to assign a base opacity reference during paired reallocation. This correspondence is an optimization relation rather than a same-surface constraint: after initialization, plastic may translate, rotate, rescale, and change appearance independently of its associated base. Base and plastic do not denote static and transient scene components; instead, $\mathcal{G}^B \cup \mathcal{G}^P$ jointly constitutes the final static representation. Persistent pairing therefore organizes retention and refinement at the representation level, while paired opacity reallocation determines how the added plastic capacity participates in rendering.

3.3 Base-Conditioned Paired Opacity Reallocation

Let α_i^B denote the fixed primitive-opacity parameter of the i-th base Gaussian, and let $\pi_i \in [0, 1]$ denote the learnable allocation fraction associated with the corresponding base-plastic pair. Rather than assigning each plastic primitive an independent primitive opacity, PortGS uses α_i^B as the pairwise opacity reference and partitions this reference between base and plastic:

$$\bar{\alpha}_i^B = \alpha_i^B(1 - \pi_i), \qquad \bar{\alpha}_i^P = \alpha_i^B \pi_i, \tag{4}$$

where $\bar{\alpha}_i^B$ and $\bar{\alpha}_i^P$ denote the effective primitive opacities supplied to the rasterizer for the paired base and plastic primitives, respectively. Accordingly,

$$\bar{\alpha}_i^B + \bar{\alpha}_i^P = \alpha_i^B, \tag{5}$$

so increasing π_i transfers effective opacity from base to plastic while preserving the pairwise total. Because plastic participation is scaled by the retained base opacity reference α_i^B, its rendering contribution is defined relative to the established representation rather than through an independent opacity degree of freedom. The resulting conservation law turns plastic participation from an independently estimated opacity into an allocated share of the retained base reference. PortGS can therefore expand the representation through additional plastic geometry and appearance while constraining rendering participation within a fixed pairwise opacity budget.

With the initialization $\pi_i^0 = 0$ defined in Sec. 3.2, Eq. 4 gives $\bar{\alpha}_i^B = \alpha_i^B$ and $\bar{\alpha}_i^P = 0$ at insertion. Subsequent optimization can progressively reallocate rendering participation from base to plastic.

3.4 Joint Rendering and Coupled Refinement

At each iteration, base and plastic primitives are combined into a single Gaussian set and jointly rasterized in one standard 3DGS rendering pass. Let $\boldsymbol{\theta}_i^{B/P} = (\boldsymbol{\mu}_i^{B/P}, \mathbf{s}_i^{B/P}, \mathbf{q}_i^{B/P}, \mathbf{h}_i^{B/P})$ collect the geometry and appearance parameters of each primitive. For view v, the rendered image is $\hat{I}_v = \mathcal{R}\big(\mathcal{C}_v; \{(\boldsymbol{\theta}_i^B, \bar{\alpha}_i^B), (\boldsymbol{\theta}_i^P, \bar{\alpha}_i^P)\}_{i=1}^N\big)$, where $\mathcal{R}$ denotes the standard 3DGS rasterizer described in Sec. 3.1. Base and plastic therefore share the same projection, depth ordering, and front-to-back compositing rather than being rendered separately and blended in image space.

The fixed pairwise primitive-opacity total $\bar{\alpha}_i^B + \bar{\alpha}_i^P = \alpha_i^B$ does not imply a fixed rendered contribution. As base and plastic diverge in projected footprint, depth, or appearance during refinement, redistributing the same pairwise primitive opacity can alter pixel-wise alpha, transmittance, and rendered color under standard alpha compositing. Paired opacity reallocation therefore constrains how rendering participation is assigned without preventing the pair from modifying the rendered image.

Optimization retains the supervision signals and training objective of the underlying transient-aware baseline. PortGS leaves the baseline loss $\mathcal{L}$ and its supervision unchanged, modifying only the Gaussian representation andhe paired opacity allocation used for rendering. Because both effective opacities depend on the same allocation variable z_i, paired opacity reallocation couples the optimization of base and plastic participation. Although the base parameters are frozen, the effective base opacity $\bar{\alpha}_i^B$ remains differentiable with respect to z_i. From Eq. 4,

$$\frac{\partial \bar{\alpha}_i^B}{\partial \pi_i} = -\alpha_i^B, \qquad \frac{\partial \bar{\alpha}_i^P}{\partial \pi_i} = \alpha_i^B, \tag{6}$$

so increasing π_i induces equal-and-opposite changes in base and plastic effective opacity. Applying the chain rule gives

$$\frac{\partial \mathcal{L}}{\partial \pi_i} = \alpha_i^B \left(\frac{\partial \mathcal{L}}{\partial \bar{\alpha}_i^P} - \frac{\partial \mathcal{L}}{\partial \bar{\alpha}_i^B} \right). \tag{7}$$

Table 1: Quantitative comparison on *NeRF On-the-go* dataset across different occlusion levels. Results are reported for low-, medium-, and high-occlusion scenes using PSNR/SSIM/LPIPS. "Ours on X" denotes PortGS applied to the corresponding underlying method. The best results are highlighted in **bold**, and the second-best results are underlined.

Method	Low Occlusion						Medium Occlusion						High Occlusion						Mean		
	Mountain			Fountain			Corner			Patio			Spot			Patio-High			Mean		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
3DGS	20.83	0.648	0.257	19.97	0.618	0.252	22.83	0.794	0.194	17.16	0.667	0.237	18.09	0.632	0.485	16.37	0.507	0.500	19.208	0.644	0.321
WildGaussians	19.63	0.681	0.253	20.48	0.661	0.224	25.35	0.862	0.111	21.07	0.804	0.134	24.87	0.872	0.099	22.60	0.801	0.170	22.333	0.780	0.165
SpotlessSplats	21.17	0.665	0.244	20.58	0.631	0.226	25.30	0.848	0.122	21.41	0.799	0.118	24.46	0.824	0.149	21.95	0.742	0.230	22.478	0.752	0.182
3DGS-HPC	21.26	0.734	0.152	21.45	0.703	0.143	24.81	0.886	0.075	20.46	0.805	0.112	23.95	0.893	0.083	22.68	0.831	0.125	22.435	0.809	0.115
DeGauss	21.25	0.741	0.177	18.80	0.651	0.204	26.39	0.891	0.068	21.78	0.822	0.098	26.58	0.906	0.062	22.97	0.823	0.126	22.962	0.806	0.123
DeSplat	20.25	0.709	0.158	20.40	0.672	0.165	26.16	0.882	0.076	18.71	0.775	0.138	25.58	0.890	0.096	22.48	0.823	0.137	22.263	0.792	0.128
RobustSplat	21.31	0.731	0.204	20.89	0.701	0.199	26.34	0.898	0.102	21.44	0.827	0.140	25.31	0.906	0.102	22.65	0.836	0.146	22.990	0.817	0.149
DualSplat	21.65	0.765	**0.123**	21.72	0.721	**0.120**	26.73	0.907	**0.047**	21.79	0.837	**0.073**	25.59	0.911	**0.045**	23.59	0.852	**0.084**	23.512	0.832	**0.082**
Ours on DeSplat	21.13	0.722	0.152	20.54	0.674	0.169	26.33	0.884	0.076	18.97	0.778	0.140	25.65	0.891	0.095	22.65	0.824	0.138	22.545	0.796	0.128
Ours on RobustSplat	21.64	0.742	0.197	21.10	0.706	0.198	26.84	0.902	0.098	21.89	0.863	0.097	26.40	0.910	0.097	22.97	0.839	0.144	23.473	0.827	0.139
Ours on DualSplat	**23.54**	**0.813**	0.139	**23.43**	**0.799**	0.129	**27.26**	**0.913**	0.091	**22.09**	**0.872**	0.090	**27.68**	**0.934**	0.050	**25.43**	**0.892**	0.088	**24.905**	**0.871**	0.098

Thus, increasing plastic participation is evaluated together with the corresponding reduction in base participation rather than through an isolated plastic-opacity path. We call this shared-variable interaction a coupled refinement signal. The coupling follows directly from paired opacity reallocation under standard automatic differentiation and requires neither an additional coupling loss nor a custom backward rule.

4 EXPERIMENTS

Experimental Setup. We evaluate PortGS on *NeRF On-the-go* (Ren et al., 2024) and *RobustNeRF* (Sabour et al., 2023). For *NeRF On-the-go*, we use six scenes following the camera-undistortion protocol of RobustSplat (Fu et al., 2025b), with $4\times$ downsampling for *Patio* and $8\times$ for the remaining scenes. For *RobustNeRF*, we evaluate five scenes at $8\times$ downsampling. We compare with vanilla 3DGS (Kerbl et al., 2023) and representative transient-free methods, including WildGaussians (Kulhanek et al., 2024), SpotLessSplats (Sabour et al., 2025), and 3DGS-HPC (Chen et al., 2026) for robust supervision; DeSplat (Wang et al., 2025b) and DeGauss (Wang et al., 2025a) for scene decomposition; and RobustSplat (Fu et al., 2025b) and DualSplat (Wang et al., 2026) for optimization dynamics. All baselines are re-run using their official implementations under matched dataset and evaluation settings.

Implementation Details. PortGS starts from an established static Gaussian state of the underlying transient-aware method. This state is retained as base, while each paired plastic primitive inherits its position, rotation, and SH appearance and is initialized with scale factor $\kappa = 0.25$. The allocation fractions are initialized as $\pi_i^0 = 0$ and optimized directly; during refinement, base parameters remain frozen, while plastic geometry, appearance, and allocation fractions are optimized with a fresh Adam optimizer. For the main cross-method evaluation, we use a common fixed-topology protocol after PortGS is introduced, disabling densification, pruning, opacity reset, and SH-degree increase. This isolates the base-plastic representation and paired opacity reallocation from method-specific topology evolution while retaining the underlying renderer and transient-aware supervision. Compatibility with topology evolution is evaluated separately using pair-aware splitting in the appendix.

4.1 COMPARISON WITH EXISTING METHODS

Evaluation on *NeRF On-the-go* Dataset. As shown in Table 1, PortGS applied to DualSplat achieves the best overall performance and the highest PSNR across all six evaluated scenes. This advantage extends across different occlusion settings rather than being concentrated in a single scene category, indicating that PortGS can further refine static representations under diverse observation conditions. The qualitative comparisons in Fig. 2 further illustrate these improvements. Compared with existing transient-free methods, PortGS produces cleaner reconstructions with fewer residual artifacts in both outdoor and indoor scenes. The highlighted regions show that PortGS better preserves fine structures and suppresses boundary artifacts around previously inconsistent regions, which is consistent with its design of introducing additional refinement capacity while constraining its rendering participation relative to the established representation.

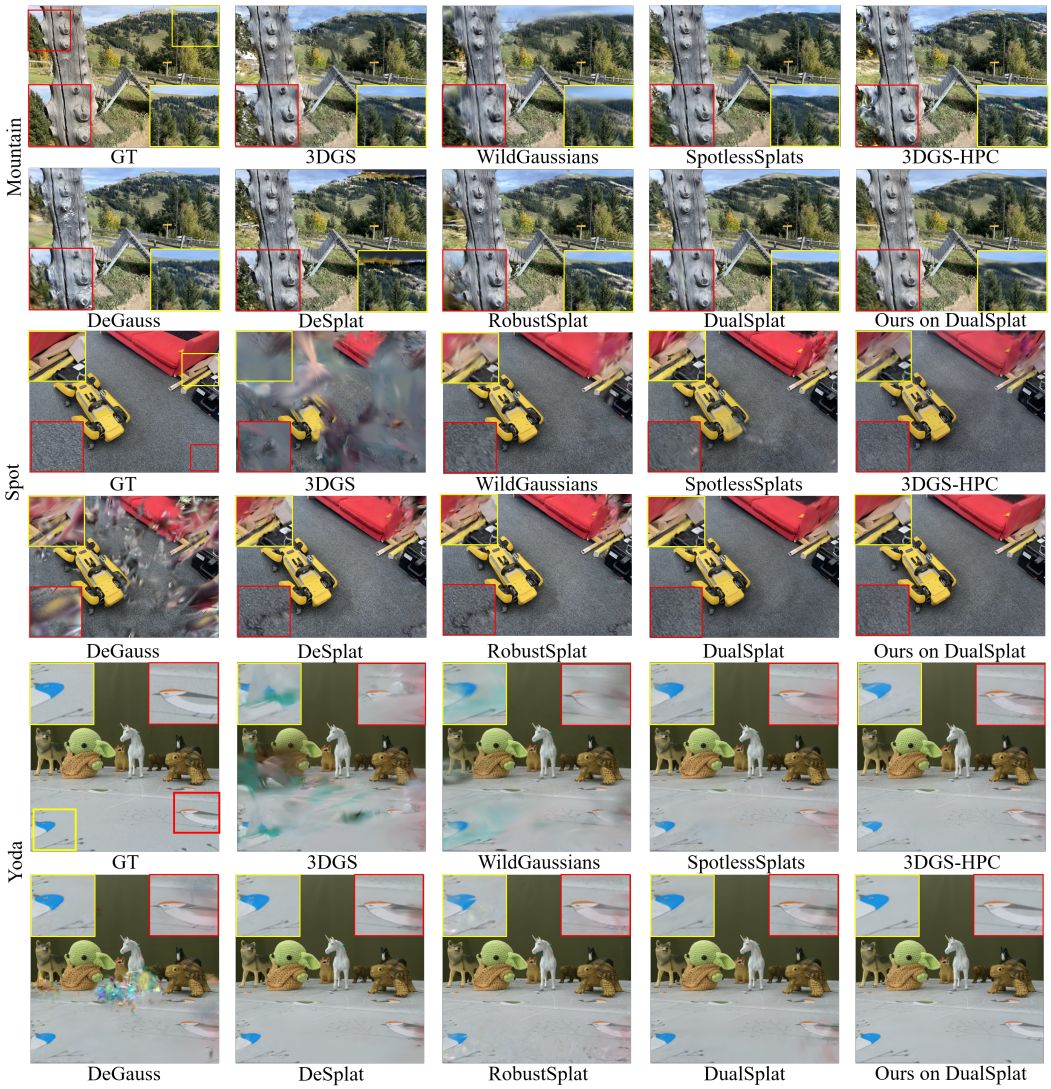


Figure 2: Qualitative comparison of novel view synthesis with existing transient-free 3DGS methods. We compare representative baselines and the best-performing PortGS integration (*Ours on DualSplat*) on *NeRF On-the-go* and *RobustNeRF*. The red and yellow boxes highlight challenging regions with transient-induced artifacts, incomplete reconstruction, and recovered fine details.

Evaluation on *RobustNeRF* Dataset. The *RobustNeRF* dataset provides a complementary evaluation under a more controlled indoor distractor setting. PortGS on DualSplat ranks first in PSNR across all five scenes, demonstrating that the refinement benefit persists under controlled transient interference. Compared with the casually captured indoor and outdoor sequences of *NeRF On-the-go*, *RobustNeRF* adopts more controlled indoor captures with deliberately varied distractor configurations. The consistent improvement under this setting indicates that the effectiveness of PortGS is not specific to the broader scene and occlusion variability of *NeRF On-the-go*. The qualitative results in Fig. 2 provide further visual evidence. PortGS reduces residual artifacts and improves reconstruction completeness in controlled indoor scenes, leading to sharper structures and more coherent appearances in previously affected regions.

PortGS Integration Ability Analysis. To evaluate whether PortGS provides a general refinement mechanism rather than being specific to a particular transient-aware pipeline, we integrate PortGS into DeSplat, RobustSplat, and DualSplat. As shown in Table 3, all three integrations achieve consistent PSNR improvements across the evaluated scenes. Despite their different transient-handling

Table 2: Quantitative comparison on *RobustNeRF* under controlled transient occlusion. We report PSNR/SSIM/LPIPS on five indoor scenes. "Ours on X" denotes PortGS applied to the corresponding underlying method. The best results are in **bold**, and the second-best results are underlined.

Method	Android			Crab1			Crab2			Statue			Yoda			Mean		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
3DGS	23.02	0.771	0.126	31.00	0.949	0.078	31.19	0.913	0.134	21.12	0.796	0.158	30.44	0.917	0.138	27.354	0.869	0.127
WildGaussians	24.76	0.826	0.090	31.14	0.950	0.076	30.51	0.904	0.153	22.52	0.859	0.117	30.85	0.911	0.148	27.956	0.890	0.117
SpotlessSplats	24.27	0.809	0.088	34.93	0.964	0.060	34.24	0.931	0.093	22.26	0.817	0.118	34.64	0.937	0.106	30.068	0.892	0.093
3DGS-HPC	24.52	0.825	0.069	33.90	0.968	0.050	34.44	0.938	0.095	22.77	0.862	0.090	35.08	0.943	**0.089**	30.142	0.907	0.079
DeGauss	24.82	0.830	0.120	35.96	0.968	0.097	34.82	0.937	0.153	22.75	0.864	0.108	34.72	0.941	0.149	30.614	0.908	0.125
DeSplat	24.24	0.819	0.089	36.09	0.969	0.045	34.13	0.931	0.098	22.65	0.850	0.096	34.03	0.934	0.099	30.228	0.901	0.085
RobustSplat	24.69	0.831	0.124	36.51	0.971	0.096	34.72	0.940	0.153	22.70	0.865	0.108	35.15	0.944	0.152	30.754	0.910	0.127
DualSplat	24.64	0.830	0.065	36.68	**0.972**	0.055	34.55	0.940	**0.091**	22.88	0.868	**0.065**	34.81	0.944	0.090	30.712	0.911	**0.073**
Ours on DeSplat	24.42	0.822	0.088	36.52	0.970	**0.042**	34.28	0.934	0.095	22.76	0.851	0.098	34.38	0.937	0.095	30.472	0.903	0.084
Ours on RobustSplat	24.80	0.835	0.122	36.79	0.971	0.093	34.93	0.942	0.155	22.91	0.869	0.107	35.30	0.946	0.150	30.946	0.913	0.125
Ours on DualSplat	**26.05**	**0.870**	**0.052**	**37.65**	0.957	0.059	**37.36**	**0.951**	**0.091**	**25.98**	**0.923**	0.077	**38.19**	**0.955**	0.091	**33.046**	**0.931**	0.074

Table 3: PSNR gains from integrating PortGS into three transient-aware 3DGS methods. We report the per-scene PSNR improvement obtained by applying PortGS to DeSplat, RobustSplat, and DualSplat on *NeRF On-the-go* and *RobustNeRF*. Positive values indicate improvement over the corresponding native method. The largest gain within each scene is highlighted in **bold**, and the second-largest gain is underlined.

Method	*NeRF On-the-go*							*RobustNeRF*					
	Mountain	Fountain	Corner	Patio	Spot	Patio-High	Mean	Android	Crab1	Crab2	Statue	Yoda	Mean
Ours on DeSplat	+0.884	+0.145	+0.174	+0.263	+0.070	+0.170	+0.284	+0.180	+0.432	+0.146	+0.103	+0.341	+0.240
Ours on RobustSplat	+0.329	+0.204	+0.503	**+0.452**	+1.088	+0.321	+0.483	+0.112	+0.280	+0.207	+0.219	+0.149	+0.193
Ours on DualSplat	**+1.887**	**+1.710**	**+0.530**	+0.300	**+2.091**	**+1.843**	**+1.394**	**+1.414**	**+0.970**	**+2.810**	**+3.101**	**+3.378**	**+2.335**

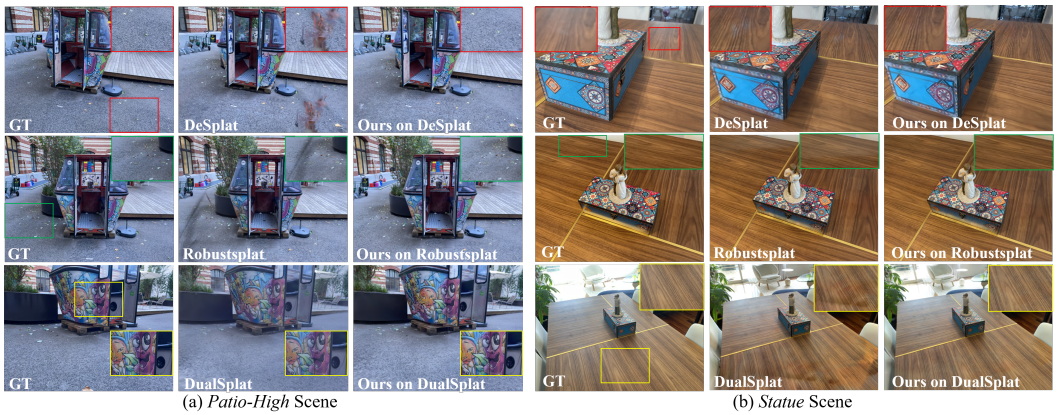
strategies and optimization procedures, each underlying method benefits from introducing the proposed Base-Plastic refinement mechanism. The qualitative comparisons in Fig. 3 further verify this integration capability. Across DeSplat, RobustSplat, and DualSplat, PortGS consistently reduces block-like artifacts and alleviates blurry or hazy regions, leading to sharper structures and more coherent appearances. These improvements suggest that PortGS does not replace the underlying transient suppression mechanism; instead, it refines the established static representation by introducing additional capacity while regulating how this capacity participates in rendering. Together, the quantitative and qualitative results indicate that PortGS functions as a representation-level refinement mechanism applicable to different transient-aware 3DGS pipelines.

4.2 Ablation Study

We organize the ablation around the two representation-level decisions that motivate PortGS. First, we examine how an established static representation should be further refined once transient-aware supervision has produced a reliable but still under-constrained scene estimate. Second, after introducing additional refinement capacity, we study how this capacity should participate in rendering. These two groups respectively test the retention–refinement organization and the rendering-participation organization of PortGS.

Representation organization. We first test the most direct alternative to PortGS by continuing to optimize the established anchor Gaussians in place under the same late-stage refinement budget. We then introduce the same additional plastic representation as PortGS but keep the base geometry and appearance trainable, isolating whether the established representation should remain retained once extra refinement capacity is available. As shown in Table 4, the full frozen-base PortGS configuration achieves the highest PSNR in all four displayed settings. These results show that simply extending optimization of the same Gaussians is less effective for late-stage refinement, while preserving the established base and assigning subsequent refinement to plastic is more effective than jointly modifying both.

Rendering-participation organization. We next isolate how the added plastic capacity should obtain rendering participation. *Independent opacity* gives plastic its own opacity path, while *base-conditioned Additive* conditions plastic participation on the base opacity reference without reducing



(a) *Patio-High* Scene (b) *Statue* Scene

Figure 3: Qualitative evaluation of PortGS integration across transient-aware 3DGS methods. We show the novel view synthesis results to visualize the refinement effect of applying PortGS to DeSplat, RobustSplat, and DualSplat on representative scenes from NeRF On-the-go and RobustNeRF. The highlighted regions show artifact reduction and improved reconstruction quality after PortGS refinement.

Table 4: Core ablations of PortGS. We report PSNR/SSIM/LPIPS for Corner and Patio scenes on RobustSplat (RS) and DualSplat (DS), respectively. The representation block compares in-place continuation, jointly trainable base-plastic refinement, and the frozen-base PortGS design. The participation block compares complete independent, base-conditioned additive, and paired reallocation under matched plastic initialization. Bold indicates the best unrounded value for each metric within each block and scene.

Configuration	RS-Corner	RS-Patio	DS-Corner	DS-Patio
In-place continuation	26.35 / 0.898 / 0.103	21.66 / 0.860 / 0.099	26.77 / 0.905 / 0.099	21.89 / 0.866 / 0.089
Trainable base + plastic	26.48 / 0.901 / **0.096**	21.72 / **0.863** / **0.094**	26.95 / 0.910 / 0.092	21.98 / 0.870 / **0.086**
Frozen base + plastic	**26.84** / **0.902** / 0.098	**21.89** / 0.863 / 0.097	**27.24** / **0.913** / **0.091**	**22.09** / **0.872** / 0.090
Independent opacity	26.67 / 0.900 / 0.101	21.84 / 0.864 / 0.096	27.07 / 0.911 / 0.093	21.09 / 0.871 / 0.087
base-cond. Additive	26.69 / 0.900 / 0.101	21.86 / **0.864** / **0.096**	27.09 / 0.911 / 0.092	22.06 / 0.871 / **0.086**
Paired Reallocation	**26.85** / **0.902** / **0.098**	**21.90** / 0.863 / 0.097	**27.24** / **0.913** / **0.091**	**22.10** / **0.872** / 0.090

the base contribution. PortGS instead uses *Paired Reallocation*, where increased plastic participation comes with a corresponding reduction in the paired base participation. Table 4 shows that base-conditioned Additive improves over Independent opacity in all four PSNR comparisons, indicating that added capacity benefits from participating relative to the retained representation rather than through an independent opacity path. Paired Reallocation further achieves the highest PSNR in four settings, supporting an additional benefit from explicit reallocation beyond base conditioning alone. It also improves lower-tail view quality, as further analyzed in Table A-1 of the appendix.

Additional analyses in the appendix examine intervention timing, refinement duration, plastic initialization scale, and PortGS's compatibility with pair-aware topology evolution.

5 Conclusion

We consider the further refinement of an established static 3DGS representation when transient-aware supervision suppresses unreliable observations but occlusion still leaves static evidence incomplete. PortGS addresses this problem through capacity-participation decoupling. This principle separates preservation of the established representation from the introduction of new refinement capacity, while preventing the added capacity from becoming an unconstrained new explanation of the scene. We realize this idea through persistent base–plastic pairing and paired opacity reallocation, which coordinate retention and refinement within a unified static representation. Experiments on *NeRF On-the-go* and *RobustNeRF*, together with integrations into multiple transient-aware baselines, validate the effectiveness of this design.

AI USE STATEMENT

In this work, we used generative AI tools to assist with writing refinement and polishing, as well as retrieval and discovery of related research works. AI-assisted literature suggestions were manually verified through independent inspection of the original publications. We have not used generative AI tools for generating experimental results, conducting experiments, implementing the proposed methods, or making scientific claims. All AI-assisted manuscript revisions were reviewed by the authors to ensure correctness and consistency with the contributions of this work. We take responsibility for the final content of this submission.

REFERENCES

Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-NeRF: A multiscale representation for anti-aliasing neural radiance fields. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5835–5844. IEEE, 2021.

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial radiance fields. In *European Conference on Computer Vision*, pp. 333–350. Springer, 2022a.

Jiahao Chen, Yipeng Qin, Lingjie Liu, Jiangbo Lu, and Guanbin Li. NeRF-HuGS: Improved neural radiance fields in non-static scenes using heuristics-guided segmentation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19436–19446. IEEE, 2024.

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 *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, pp. 1–19, 2026.

Xingyu Chen, Qi Zhang, Xiaoyu Li, Yue Chen, Ying Feng, Xuan Wang, and Jue Wang. Hallucinated neural radiance fields in the wild. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12933–12942. IEEE, 2022b.

Hiba Dahmani, Moussab Bennehar, Nathan Piasco, Luis Roldao, and Dzmitry Tsishkou. SWAG: Splatting in the wild images with appearance-conditioned gaussians. In *European Conference on Computer Vision*, pp. 325–340. Springer, 2024.

Xiaobiao Du, Yida Wang, and Xin Yu. MVGS: Multi-view regulated gaussian splatting for novel view synthesis. In *European Conference on Computer Vision*, pp. 564–581. Springer, 2026.

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 preprint arXiv:2512.04815*, 2025a.

Chuanyu Fu, Yuqi Zhang, Kunbin Yao, Guanying Chen, Yuan Xiong, Chuan Huang, Shuguang Cui, and Xiaochun Cao. RobustSplat: Decoupling densification and dynamics for transient-free 3DGS. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27126–27136. IEEE, 2025b.

Yuto Furutani, Takashi Otonari, Kaede Shiohara, and Toshihiko Yamasaki. Wild3R: Feed-forward 3D gaussian splatting from unconstrained sparse photo collection. *arXiv preprint arXiv:2606.11894*, 2026.

Xin Huang, Qi Zhang, Ying Feng, Hongdong Li, Xuan Wang, and Qing Wang. HDR-NeRF: High dynamic range neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18377–18387. IEEE, 2022.

Zeji Hui, Amirali Khodadadian Gostar, WeiQin Chuah, Alireza Bab-Hadiashar, and Ruwan Tennakoon. Joint modeling of corruption-driven and information-limited uncertainty for robust 3D gaussian splatting. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 688–697. IEEE, 2026.

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

Hanyang Kong, Xingyi Yang, and Xinchao Wang. RoGSplat: Robust gaussian splatting via generative priors. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25735–25745. IEEE, 2025.

Jonas Kulhanek, Songyou Peng, Zuzana Kukelova, Marc Pollefeys, and Torsten Sattler. WildGaussians: 3D gaussian splatting in the wild. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, volume 37, pp. 21271–21288, 2024.

Chengqi Li, Zhihao Shi, Yangdi Lu, Wenbo He, and Xiangyu Xu. Robust neural rendering in the wild with asymmetric dual 3D gaussian splatting. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, volume 38, Main Conference, pp. 50166–50191, 2025.

Peihao Li, Shaohui Wang, Chen Yang, Bingbing Liu, Weichao Qiu, and Haoqian Wang. NeRF-MS: Neural radiance fields with multi-sequence. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18545–18554. IEEE, 2023.

Ruiqi Li and Yiu-ming Cheung. Modeling and identifying distractors with curriculum for robust 3D gaussian splatting. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10122–10131, 2025.

Shengjia Liang, Tong Chen, Qichuan Geng, Yuan Xiong, and Zhong Zhou. CleanSplat: curriculum structural gaussian splatting for spot-free novel view synthesis. *Virtual Reality & Intelligent Hardware*, 8(2):235–249, 2026.

Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. BARF: Bundle-adjusting neural radiance fields. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5721–5731. IEEE, 2021.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 788–797. IEEE, 2025.

Xinxin Liu, Xue Wang, Guoqing Zhou, and Qing Wang. ManifoldNeuS: Manifold-aware view optimizability for pose-free neural surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 261–271, 2026.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20654–20664. IEEE, 2024.

Alexander Markin, Vadim Pryadilshchikov, Artem Komarichev, Ruslan Rakhimov, Peter Wonka, and Evgeny Burnaev. T-3DGS: Removing transient objects for 3D scene reconstruction. *arXiv preprint arXiv:2412.00155*, 2024.

Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural radiance fields for unconstrained photo collections. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7206–7215, 2021.

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 Computer Vision (ECCV)*, pp. 405–421, 2020.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics, volume=41, number=4, pages=1–15, year=2022, publisher=ACM New York, NY, USA*.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Tianbo Pan, Xingyi Yang, Shizun Wang, and Xinchao Wang. Visual geometry transformer in the wild: Distractor-free 3D reconstruction. *arXiv preprint arXiv:2606.22787*, 2026.

Wongi Park, Jiyeon Lim, Minjae Lee, Myeongseok Nam, Seongjun Choi, Jungwoo Kim, Soomok Lee, William J Beksi, and Sang-Hyun Lee. Rectifying mask via entropy for distractor-free 3DGS in ambiguous scenarios. *arXiv preprint arXiv:2606.29496*, 2026a.

Wongi Park, Myeongseok Nam, Siwon Kim, Sangwoo Jo, and Soomok Lee. ForestSplats: Deformable transient field for gaussian splatting in the wild. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 6978–6987. IEEE, 2026b.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8931–8940. IEEE, 2024.

Sara Sabour, Suhani Vora, Daniel Duckworth, Ivan Krasin, David J Fleet, and Andrea Tagliasacchi. RobustNeRF: Ignoring distractors with robust losses. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20626–20636. IEEE, 2023.

Sara Sabour, Lily Goli, George Kopanas, Mark Matthews, Dmitry Lagun, Leonidas Guibas, Alec Jacobson, David Fleet, and Andrea Tagliasacchi. SpotLessSplats: Ignoring distractors in 3D gaussian splatting. *ACM Transactions on Graphics*, 44(2):1–11, 2025.

Kangmin Seo and Jae-Pil Heo. Per-view gaussian predictions enable training-free distractor filtering in feed-forward 3DGS. *arXiv preprint arXiv:2608.26951*, 2026.

Kangmin Seo, MinKyu Lee, Tae-Young Kim, ByeongCheol Lee, JoonSeoung An, and Jae-Pil Heo. PDF-GS: Progressive distractor filtering for robust 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 468–477, June 2026.

Yedong Shen, Shiqi Zhang, Sha Zhang, Yifan Duan, Xinran Zhang, Wenhao Yu, Lu Zhang, Jiajun Deng, and Yanyong Zhang. GA-GS: Generation-assisted gaussian splatting for static scene reconstruction. *arXiv preprint arXiv:2604.04331*, 2026.

Yuang Shi, Géraldine Morin, Simone Gasparini, and Wei Tsang Ooi. LapisGS: Layered progressive 3D gaussian splatting for adaptive streaming. In *2025 International Conference on 3D Vision (3DV)*, pp. 991–1000. IEEE, 2025.

Jiadong Tang, Yu Gao, Dianyi Yang, Liqi Yan, Yufeng Yue, and Yi Yang. DroneSplat: 3D gaussian splatting for robust 3D reconstruction from in-the-wild drone imagery. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 833–843. IEEE, 2025.

Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. DreamGaussian: Generative gaussian splatting for efficient 3D content creation. In *International Conference on Learning Representations*, volume 2024, pp. 33879–33896, 2024.

Paul Ungermann, Armin Ettenhofer, Matthias Nießner, and Barbara Roessle. Robust 3D gaussian splatting for novel view synthesis in presence of distractors. In *DAGM German Conference on Pattern Recognition*, pp. 153–167. Springer, 2024.

Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. IBRNet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2021.

Rui Wang, Quentin Lohmeyer, Mirko Meboldt, and Siyu Tang. DeGauss: Dynamic-static decomposition with gaussian splatting for distractor-free 3D reconstruction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6294–6303. IEEE, 2025a.

Xu Wang, Zhiru Wang, Shiyun Xie, Chengwei Pan, and Yisong Chen. DualSplat: Robust 3D gaussian splatting via pseudo-mask bootstrapping from reconstruction failures. *arXiv preprint arXiv:2604.21631*, 2026.

Yihao Wang, Marcus Klasson, Matias Turkulainen, Shuzhe Wang, Juho Kannala, and Arno Solin. DeSplat: Decomposed gaussian splatting for distractor-free rendering. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 722–732. IEEE, 2025b.

Yukun Wang, Kunhong Li, Minglin Chen, Longguang Wang, Shunbo Zhou, Kaiwen Xue, and Yulan Guo. Distractor-free novel view synthesis via exploiting memorization effect in optimization. In *European Conference on Computer Vision*, pp. 477–493. Springer, 2024a.

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

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320. IEEE, 2024.

Tianhao Wu, Fangcheng Zhong, Andrea Tagliasacchi, Forrester Cole, and Cengiz Oztireli. D^2NeRF: Self-supervised decoupling of dynamic and static objects from a monocular video. *Advances in Neural Information Processing Systems*, 35:32653–32666, 2022.

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

Jiacong Xu, Yiqun Mei, and Vishal M Patel. Wild-GS: Real-time novel view synthesis from unconstrained photo collections. *Advances in Neural Information Processing Systems*, 37:103334–103355, 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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15855–15865. IEEE, 2023.

Yunseo Yang, Youngho Yoon, and Kuk-Jin Yoon. PriSplat: Propagating reliable multi-view information for distractor-free 3DGS. In *The European Conference on Computer Vision 2026 (ECCV 2026)*. The European Conference on Computer Vision 2026 (ECCV 2026), 2026.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelNeRF: Neural radiance fields from one or few images. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4576–4585. IEEE, 2021.

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

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 *European Conference on Computer Vision*, pp. 341–359. Springer, 2024.

Chuyue Zhao, Xin Huang, Kun Yang, Xue Wang, and Qing Wang. Generalizable 3D gaussian splatting for novel view synthesis. *Pattern Recognition*, 161:111271, 2025.

A Appendix

A.1 Dissuasion and Additional Ablation Study

Lower-tail view quality. We further examine whether different participation strategies affect reconstruction quality on lower-performing evaluation views. As shown in Table A-1, paired reallocation achieves the highest CVaR_{20} on both scenes, while also giving the best P20 on Corner and remaining competitive on Patio. These results indicate that paired reallocation provides a more consistent advantage in the lower tail of the view-wise quality distribution, complementing the terminal reconstruction results reported in the main paper.

Schedule sensitivity. We further study when PortGS should be introduced and how long the subsequent refinement should continue. As shown in Table A-2, in-place continuation clearly outperforms PortGS when intervention occurs at 15k or 18k. At these earlier stages, the static representation is still undergoing substantial scene formation, and freezing base prematurely restricts the geometry

Table A-1: Lower-tail view quality under different participation laws on RobustSplat. P20 denotes the 20th-percentile view-wise PSNR, while CVaR_{20} denotes the mean PSNR over the worst-performing 20% of evaluation views.

	Independent		Base-cond. Additive		Paired reallocation	
Scene	P20↑	CVaR_{20} ↑	P20↑	CVaR_{20} ↑	P20↑	CVaR_{20} ↑
Corner	23.918	21.424	23.860	21.477	**24.288**	**21.823**
Patio	19.358	18.172	**19.382**	18.203	19.290	**18.232**

and appearance updates that remain necessary. Once the representation becomes sufficiently established, the relation reverses, and PortGS outperforms in-place continuation at both 20k and 25k. This transition is consistent with the intended role of PortGS as a late-stage retain-and-refine mechanism rather than an alternative to early-stage scene formation. With the anchor fixed at 20k, we observe that extending the refinement duration from 4k to 10k updates produces a different behavior for the two strategies. PortGS remains nearly unchanged over the longer refinement horizon, whereas in-place continuation degrades on both scenes. We interpret this behavior as being consistent with the retain-and-refine design, rather than as evidence that longer refinement is universally beneficial.

Table A-2: Schedule sensitivity. Intervention timing varies the anchor step t_0 at which PortGS is introduced while fixing the subsequent refinement length to $\Delta t = 4$k updates; refinement duration varies the number of post-anchor refinement updates Δt while fixing the anchor at $t_0 = 20$k. We report PSNR for PortGS and matched in-place continuation on *Ours-on-RobustSplat*, both initialized from the same anchor state for each comparison.

		Intervention timing t_0				Refinement duration Δt	
Scene	Method	15k	18k	20k	25k	4k	10k
Corner	In-place	**26.74**	**26.72**	26.50	26.43	26.50	26.35
	PortGS	23.30	22.40	**26.79**	**26.67**	**26.79**	**26.83**
Patio	In-place	**21.76**	**21.85**	21.70	21.70	21.70	21.66
	PortGS	19.24	18.46	**21.90**	**21.85**	**21.90**	**21.89**

Plastic initialization scale. We vary the initial plastic scale over $\kappa \in \{0.1, 0.25, 0.5, 1.0\}$. As reported in Table A-3, PortGS is relatively insensitive to κ, particularly across the reduced-scale settings from 0.1 to 0.5. Nevertheless, $\kappa = 0.25$ yields the highest mean PSNR across the four evaluated baseline settings, while full-scale initialization ($\kappa = 1.0$) is slightly weaker overall and shows a degradation on RobustSplat Corner. We therefore use $\kappa = 0.25$ as a moderate reduced-scale default throughout the main experiments.

Table A-3: Sensitivity to the plastic initialization scale κ. We report PSNR for the corresponding configuration and scene on *Ours-on-RobustSplat*. The last row averages the four settings. † denotes the default value used in the main experiments.

	$\kappa = 0.10$	$\kappa = 0.25^\dagger$	$\kappa = 0.50$	$\kappa = 1.00$
RS-Corner	26.84	**26.85**	26.83	26.71
RS-Patio	21.90	21.90	21.91	**21.93**
DS-Corner	**27.24**	**27.24**	27.23	27.21
DS-Patio	22.08	22.10	22.09	**22.13**
Mean	24.515	**24.523**	24.515	24.495

Compatibility with topology evolution. The main experiments adopt a fixed-topology refinement protocol to provide a common post-anchor setting across different underlying methods and to isolate the base-plastic representation and paired opacity reallocation from baseline-specific topology evolution. We additionally evaluate a pair-aware plastic splitting variant, together with a moment-reset control that applies the same plastic Adam-moment reset without modifying the Gaussian topology.

Table A-4: Compatibility with topology evolution. We evaluate whether PortGS remains compatible with topology updates under a controlled topology-evolution setting. We compare fixed-topology refinement with an optimizer-reset-only control and a pair-aware plastic splitting variant. All variants use the same initialization and refinement settings within this comparison. Each setting reports PSNR/SSIM/LPIPS. Bold indicates the best unrounded result for each metric and scene.

Configuration	RS-Corner	RS-Patio	DS-Corner	DS-Patio
Fixed topology	26.725 / 0.901 / 0.098	21.847 / 0.863 / **0.096**	27.197 / 0.912 / **0.091**	**22.076 / 0.871 / 0.089**
Optimizer-reset control	26.731 / 0.901 / 0.098	21.853 / 0.863 / **0.096**	27.198 / 0.912 / **0.091**	**22.076 / 0.871 / 0.089**
Pair-aware plastic split	**26.748 / 0.902 / 0.097**	**21.860 / 0.864 / 0.096**	**27.219 / 0.913 / 0.091**	22.074 / **0.871 / 0.089**

As shown in Table A-4, pair-aware splitting improves the PSNR on both Corner and Patio under RobustSplat. Under DualSplat, it improves Corner while remaining nearly unchanged on Patio. The effect therefore varies across backbone-scene settings, but the topology-evolving variant remains broadly comparable to, and can slightly improve upon, the fixed-topology protocol. These results indicate that fixed topology is not required by the PortGS formulation; rather, it provides a controlled cross-backbone protocol for isolating the proposed representation and participation mechanisms.

More generally, the main experiment considers a controlled plastic-only pair-aware splitting operation rather than the complete native topology-update schedules of the underlying methods. Extending paired opacity reallocation to backbone-specific densification, pruning, and structural evolution therefore remains an open direction for future work.