RELO-S4DGS: RELIABILITY MODELING FOR LOCALITY OPTIMIZATION IN SPARSE-VIEW DYNAMIC GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent 4D Gaussian Splatting methods have achieved impressive quality and efficiency for dynamic scene reconstruction, yet sparse-view inputs severely underconstrain scene geometry, leading to geometric ambiguity and view-inconsistent reconstructions. We observe substantial locality degradation in anchor-based Gaussian representations under sparse supervision: to fit limited observations or compensate for insufficient spatial coverage of anchors, anchors tend to decode excessively large offsets, placing Gaussians far from their source anchors and undermining representation locality. Based on this observation, we propose **ReLo-S4DGS**, which, to the best of our knowledge, is the first anchor-based Gaussian Splatting framework for sparse-view dynamic scene reconstruction. To address locality degradation, ReLo-S4DGS models anchor reliability and suppresses low-reliability rendering contributions to favor reliable local representations, we further introduce dynamic-aware locality correction to rectify excessive offsets and geometric regularization to constrain local geometry, thereby mitigating sparse-view overfitting. Extensive experiments on standard dynamic scene benchmarks demonstrate state-of-the-art performance of our method.

1 INTRODUCTION

Dynamic novel-view synthesis aims to reconstruct time-varying 3D scenes from multi-view videos and enable free-viewpoint rendering. Recent advances in 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) and its dynamic extensions, such as 4DGS (Yang et al., 2024a) and 4DGaussians (Wu et al., 2024), have achieved remarkable rendering quality and efficiency, yet most existing methods rely on densely captured multi-view videos. When the number of training cameras is substantially reduced, limited spatial supervision makes dynamic scene reconstruction severely underconstrained. Existing sparse-view methods often introduce additional depth (Wang et al., 2025c), tracking (Ozeki et al., 2026), or generative priors (Pan et al., 2026) to compensate for missing observations. While these priors provide useful complementary cues, the scene representation can still become unreliable when geometric supervision is insufficient. Reconstructing dynamic scenes with reliable geometry and strong novel-view generalization from only a few cameras therefore remains challenging. It thus requires further investigation into the structural prior inherent in the scene representation and its behavior under sparse supervision.

Anchor-based Gaussian representations organize scene content in a structured manner by decoding local Gaussians from anchors. Representative methods such as Scaffold-GS (Lu et al., 2024) and its dynamic extensions such as LocalDyGS (Wu et al., 2025) rely on an important locality assumption: each anchor mainly represents its spatial neighborhood, while decoded offsets account for only limited local variations. Under dense multi-view supervision, strong geometric constraints naturally maintain such locality, allowing numerous local anchors to jointly reconstruct the scene. Under sparse supervision, however, this locality becomes particularly important yet more fragile. As shown in Fig. 1, we quantify anchor locality using the normalized decoded offset ratio, defined as the ℓ_2 norm of the opacity-weighted decoded offsets over visible timestamps normalized by the anchor scale, where larger values indicate weaker locality. Macro-averaged across all six scenes of the Neural 3D Video (Li et al., 2022) dataset, the proportion of anchors with a ratio ≥ 1 increases from 24.6% under dense-view training to 94.0% with only four training views. Insufficient multi-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

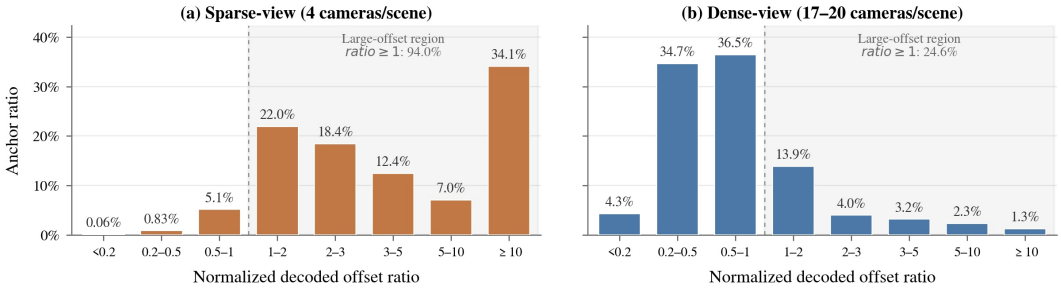

Figure 1: Macro-averaged normalized decoded-offset ratio distributions across six scenes of Neural 3D Video dataset under sparse-view (4 cameras) and dense-view (17–20 cameras) training.

view constraints allow geometrically inaccurate Gaussians to still fit the training images well, while insufficient spatial anchor coverage further encourages existing anchors to decode Gaussians far from their source anchors. Such large offsets weaken the association between anchors and their local neighborhoods, giving individual anchors excessive freedom for view-specific fitting and degrading generalization to unseen views.

However, a large offset does not necessarily indicate an erroneous representation. In low-dynamic regions, it may suggest that an anchor has drifted away from the local structure it should represent; in highly dynamic regions, it may instead arise from genuine motion or insufficient local coverage. Simply suppressing all large offsets would therefore restrict both undesirable non-local fitting and legitimate dynamics. In other words, large offsets are a symptom of potential locality breakdown, rather than a sufficient criterion for identifying erroneous representations. This raises a more fundamental question: rather than determining whether an offset is large, we should determine whether the local representation provided by an anchor is *reliable*. To this end, we accumulate evidence from local geometric risk, temporal repeatability, and cross-view optimization agreement, and model anchor-level reliability using a Beta-distribution-based reliability measure. The estimated reliability suppresses low-reliability rendering contributions, thereby directing optimization toward more reliable local representations. Dynamic-aware locality correction further restores degraded locality according to anchor dynamicity: low-dynamic anchors are recentered to restore their correspondence with local structures, whereas high-dynamic anchors establish new local support around displaced structures to avoid excessive spatial spans within a single anchor.

To address the challenges of sparse-view dynamic reconstruction described above, we propose **ReLo-S4DGS**, a reliability modeling framework for locality optimization in sparse-view dynamic scenes. Specifically, reliability modeling suppresses unreliable rendering contributions and directs optimization toward reliable local representations, whereas dynamic-aware locality correction restores degraded anchor locality while preserving genuine scene dynamics. We further incorporate anchor-level locality regularization and sparse keyframe depth supervision to provide complementary geometric constraints in poorly observed regions and improve overall geometric consistency.

Our main contributions are summarized as follows:

- We identify anchor locality degradation under sparse supervision, which is characterized by significantly larger decoded offsets. Based on this observation, we propose **ReLo-S4DGS**, which, to the best of our knowledge, is the first anchor-based Gaussian Splatting framework for sparse-view dynamic scene reconstruction.
- We introduce reliability modeling to estimate anchor reliability from complementary geometric, temporal, and cross-view evidence and favor reliable local representations, dynamic-aware locality correction to restore degraded locality while preserving genuine dynamics, and geometric regularization to improve local and scene-level geometric consistency. These modules together optimize the locality of anchors under sparse supervision.
- Extensive experiments on three standard dynamic scene benchmarks demonstrate that ReLo-S4DGS achieves state-of-the-art performance under sparse-view settings.

2 Related Work

Novel View Synthesis. Novel view synthesis reconstructs 3D scenes from observed images and renders unseen viewpoints. NeRF (Mildenhall et al., 2020) establishes implicit neural radiance fields for high-quality rendering, while 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) introduces explicit Gaussian primitives for real-time rendering. Recent work extends 3DGS to sparse-view reconstruction (Zhu et al., 2024; Li et al., 2024a; Zhang et al., 2024; Park et al., 2025a), surface reconstruction (Guédon & Lepetit, 2024; Huang et al., 2024), fast optimization (Ren et al., 2026), and various other applications (Chen et al., 2024; Keetha et al., 2024). Together, These works extend 3DGS beyond static novel-view synthesis and provide a foundation for dynamic scene modeling.

Dynamic Scene Modeling. Dynamic Gaussian methods differ mainly in how temporal variations are represented. Dynamic 3D Gaussians (Luiten et al., 2024) tracks persistent Gaussians over time, while Deformable-3DGS (Yang et al., 2024b), 4DGaussians (Wu et al., 2024), and E-D3DGS (Bae et al., 2024) model dynamics through implicit features or deformation fields. 4DGS (Yang et al., 2024a), Spacetime-GS (Li et al., 2024b), FreeTimeGS (Wang et al., 2025b), and Gaussian-Flow (Lin et al., 2024) instead adopt explicit spatio-temporal parameterizations, whereas Ex4DGS (Lee et al., 2024) and SplineGS (Park et al., 2025b) model motion through interpolation or continuous trajectories. DASH (Chen et al., 2025) introduces dynamic-static decomposition with 4D hash encoding, while LocalDyGS (Wu et al., 2025) and 4D-Scaffold-GS (Cho et al., 2026) extend anchor-based local representations to dynamic scenes. LongVolCap (Xu et al., 2024) and ClipGStream (Liang et al., 2026) target long sequences, while SPIN-4DGS (Kim et al., 2026) addresses fast motion with large inter-frame displacements. Despite these advances, most dynamic Gaussian methods assume relatively sufficient multi-view supervision, making sparse-view dynamic reconstruction particularly challenging. SplatFields (Mihajlovic et al., 2024), Sparse4DGS (Hu et al., 2025), and MonoFusion (Wang et al., 2025c) introduce additional neural, geometric, or monocular priors. 4C4D (Zhou et al., 2026) improves geometric learning under sparse supervision, whereas SparseCam4D (Pan et al., 2026) supplements missing observations using ViewCrafter (Yu et al., 2024) with VGGT (Wang et al., 2025a) conditioning. Our method differs from those methods in that it focuses on the degradation of anchor locality under sparse supervision and optimizes local dynamic representations through reliability modeling.

3 Method

In this section, we present ReLo-S4DGS, a framework for sparse-view dynamic scene reconstruction. As illustrated in Fig. 2, our method combines beta-evidence reliability modeling, reliability-aware opacity modulation, dynamic-aware locality correction, and geometric regularization.

3.1 Preliminary

We adopt the anchor-based dynamic representation of LocalDyGS (Wu et al., 2025), built on Scaffold-GS (Lu et al., 2024). Each anchor $\mathbf{a}_i$ mixes a static feature $\mathbf{f}_i^s$ and a spatio-temporal feature $\mathbf{f}_i^d(t)$ queried from a spatio-temporal hash grid, using an MLP-predicted weight $m_i(t)$:

$$\mathbf{f}_i(t) = (1 - m_i(t))\mathbf{f}_i^s + m_i(t)\mathbf{f}_i^d(t). \tag{1}$$

Given the fused feature $\mathbf{f}_i(t)$ and viewing direction $\mathbf{v}$, decoders F_{dec} predict K local Gaussians:

$$\{\mathbf{x}_{ik}(t), \tilde{\alpha}_{ik}(t), \mathbf{s}_{ik}(t), \mathbf{q}_{ik}(t), \mathbf{c}_{ik}(t)\}_{k=1}^{K} = F_{\text{dec}}\left(\mathbf{a}_i, \mathbf{f}_i(t), \mathbf{v}\right). \tag{2}$$

where $\mathbf{x}_{ik}$, $\tilde{\alpha}_{ik}$, $\mathbf{s}_{ik}$, $\mathbf{q}_{ik}$, and $\mathbf{c}_{ik}$ denote Gaussian center, raw opacity, scale, rotation, and color.

3.2 Beta-Evidence Reliability Modeling

We assess anchor reliability from three complementary items: spatial structure, temporal persistence, and optimization consistency. During training, the corresponding statistics are accumulated over fixed, non-overlapping update intervals, and anchor reliability is updated at the end of each update interval. Specifically, **local geometric risk** characterizes spatial locality, **temporal repeatability** measures persistent support over time, and **cross-view optimization agreement** evaluates the consistency of geometric updates across views.

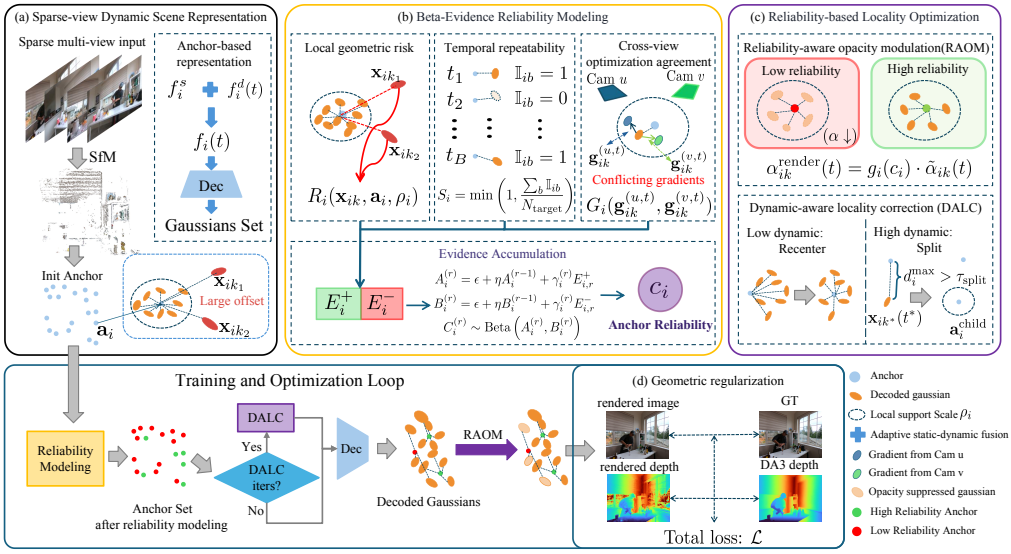

Figure 2: **Overview of ReLo-S4DGS.** Our framework combines (a) sparse-view dynamic scene representation, (b) beta-evidence reliability modeling, (c) reliability-based locality optimization, and (d) geometric regularization. Specifically, anchors decode time-varying local Gaussians, whose geometric, temporal, and cross-view evidence is accumulated to estimate anchor reliability. This reliability modulates Gaussian opacity, and dynamic-aware correction recenters low-dynamic anchors or expands support for high-dynamic anchors. Geometric regularization further constrains local geometry through locality and keyframe depth supervision.

Local geometric risk. We measure contribution-weighted deviations beyond anchor's local support:

$$R_i(\mathbf{x}_{ik}, \mathbf{a}_i, \rho_i) = \frac{\sum_{t,k} w_{ik}(t) \left[1 - \exp\left(-\frac{1}{2}\left[\frac{\|\mathbf{x}_{ik}(t) - \mathbf{a}_i\|_2}{\rho_i} - \tau_l\right]_+^2\right)\right]}{\sum_{t,k} w_{ik}(t) + \epsilon}. \tag{3}$$

where ρ_i denotes the local support scale of anchor i, computed as the median distance to its K_{NN} nearest anchors, $w_{ik}(t)$ denotes the rendering contribution at time t, computed as the product of the rendering opacity and the projected 2D area, τ_l is the locality-risk threshold, $[\cdot]_+ = \max(\cdot, 0)$ denotes the positive-part operator, and ϵ is a small positive constant. The bounded mapping assigns higher risk to deviations beyond the local support while limiting the influence of extreme offsets.

Temporal repeatability. We measure persistent anchor support across temporal regions:

$$u_{ib} = \sum_{(t,k)\in b} w_{ik}(t), \mathbb{I}_{ib} = \mathbb{I}\left[(u_{ib} > \tau_{\text{abs}}) \wedge (u_{ib} \geq \tau_{\text{rel}} \max_{b'} u_{ib'})\right], S_i = \min\left(1, \frac{\sum_b \mathbb{I}_{ib}}{N_{\text{target}}}\right). \tag{4}$$

where b indexes temporal regions and u_{ib} is the support of anchor i in region b. $\mathbb{I}_{ib}$ marks valid support when u_{ib} exceeds both τ_{abs} and τ_{rel} times of the anchor's maximum support across regions. $S_i \in [0, 1]$ measures temporal repeatability, and N_{target} is the target number of supported regions.

Cross-view optimization agreement. Finally, we measure the agreement of optimization gradients for the same decoded Gaussian across views at the same time:

$$G_i(\mathbf{g}_{ik}^{(u,t)}, \mathbf{g}_{ik}^{(v,t)}) = \text{Avg}_{t,k,u\neq v} \frac{\mathbf{g}_{ik}^{(u,t)} \cdot \mathbf{g}_{ik}^{(v,t)}}{\|\mathbf{g}_{ik}^{(u,t)}\|_2 \|\mathbf{g}_{ik}^{(v,t)}\|_2 + \epsilon}, \qquad \mathbf{g}_{ik}^{(v,t)} = \frac{\partial \mathcal{L}^{(v,t)}}{\partial \mathbf{x}_{ik}(t)}. \tag{5}$$

where $\mathbf{g}_{ik}^{(v,t)}$ is the position gradient of the k-th Gaussian decoded from anchor i at time t under view v, and $\mathcal{L}$ denotes the overall training objective. Averaging is performed over valid view pairs, times,

and Gaussians within the current update interval. Positive and negative G_i indicate consistent and conflicting geometric updates, respectively.

The three items defined above provide complementary supporting and contradicting evidence:

$$E_i^+ = S_i + [G_i]_+, \qquad E_i^- = R_i + [-G_i]_+. \tag{6}$$

where E_i^+ and E_i^- denote supporting and contradicting evidence for anchor i, respectively. Let r index the update intervals. At the end of interval r, the evidence updates an anchor-level Beta state. To emphasize recent and sufficiently observed evidence, historical evidence is decayed and current evidence is weighted by observation strength:

$$A_i^{(r)} = \epsilon + \eta A_i^{(r-1)} + \gamma_i^{(r)} E_{i,r}^+, \qquad B_i^{(r)} = \epsilon + \eta B_i^{(r-1)} + \gamma_i^{(r)} E_{i,r}^-. \tag{7}$$

where η controls historical evidence decay and $\gamma_i^{(r)} \in [0, 1]$ denotes the current observation strength. Reliability updates are skipped for insufficiently observed anchors, with more details provided in the appendix. The updated parameters define the anchor-level Beta reliability state:

$$C_i^{(r)} \sim \mathrm{Beta}\left(A_i^{(r)}, B_i^{(r)}\right), \qquad c_i = \mathbb{E}[C_i^{(r)}] = \frac{A_i^{(r)}}{A_i^{(r)} + B_i^{(r)}}. \tag{8}$$

The posterior mean c_i serves as the reliability score of anchor i.

3.3 Reliability-Aware Opacity Modulation

The reliability estimated in the previous subsection reflects anchor trustworthiness, yet unreliable anchors may still fit sparse observations through substantial rendering contributions. We therefore introduce **Reliability-Aware Opacity Modulation (RAOM)** to down-weight unreliable anchors and steer optimization toward reliable local representations, while retaining a minimum gate to avoid premature deactivation. Specifically, we define an anchor-level opacity gate:

$$g_i(c_i) = g_{\min} + (1 - g_{\min})\,\mathrm{clip}\left(\frac{c_i - c_{\mathrm{low}}}{c_{\mathrm{high}} - c_{\mathrm{low}}}, 0, 1\right), \quad \alpha_{ik}^{\mathrm{render}}(t) = g_i(c_i) \cdot \tilde{\alpha}_{ik}(t). \tag{9}$$

where c_{low} and c_{high} define the modulation range, $g_{\min}$ is the minimum gate value, $\tilde{\alpha}_{ik}(t)$ and $\alpha_{ik}^{\mathrm{render}}(t)$ are the original and modulated opacities. The gate equals 1 for $c_i \geq c_{\mathrm{high}}$, equals $g_{\min}$ for $c_i \leq c_{\mathrm{low}}$, and varies linearly for $c_{\mathrm{low}} < c_i < c_{\mathrm{high}}$. Thus, unreliable anchors are softly down-weighted without being completely deactivated, while reliable anchors retain their original opacity.

3.4 Dynamic-Aware Locality Correction

We further introduce **Dynamic-Aware Locality Correction (DALC)** to explicitly restore degraded anchor locality. Since large Gaussian displacements may reflect either local drift or genuine motion beyond current local support, DALC distinguishes them by anchor dynamicity and applies recentering or local support expansion accordingly. Specifically, we define anchor dynamicity by the relative contributions of the dynamic and static feature:

$$D_i = \mathrm{Avg}_t \frac{\|m_i(t)\mathbf{f}_i^d(t)\|_2}{\|(1 - m_i(t))\mathbf{f}_i^s\|_2 + \epsilon}. \tag{10}$$

where $\mathbf{f}_i^d(t)$, $\mathbf{f}_i^s$, and $m_i(t)$ follow equation 1. We rank anchors by D_i, treating the bottom $P\%$ as low-dynamic anchors and the top $(100 - P)\%$ as high-dynamic anchors.

For low-dynamic anchors, we test whether the anchor lies inside the convex hull of its decoded Gaussian centers at the reference time $t' = 0$. If not, we recenter it to their opacity-weighted mean:

$$\mathbf{a}_i \notin \mathrm{Conv}\left(\{\mathbf{x}_{ik}(t')\}_{k=1}^K\right), \quad \mathbf{a}_i \leftarrow \frac{\sum_k \tilde{\alpha}_{ik}(t')\mathbf{x}_{ik}(t')}{\sum_k \tilde{\alpha}_{ik}(t') + \epsilon}. \tag{11}$$

where $\mathrm{Conv}(\cdot)$ denotes the convex hull. This moves the anchor to a more plausible local position.

For high-dynamic anchors, large displacements may instead indicate that genuine motion extends beyond the current local support. Within each maintenance interval, we record

$$d_i^{\max} = \max_{t,k}\left(\frac{\|\mathbf{x}_{ik}(t) - \mathbf{a}_i\|_2}{\rho_i}\right), \quad (t^*, k^*) = \arg\max_{t,k}\left(\frac{\|\mathbf{x}_{ik}(t) - \mathbf{a}_i\|_2}{\rho_i}\right). \tag{12}$$

When $d_i^{\max} > \tau_{\text{split}}$, we create a new local anchor at the corresponding Gaussian position:

$$\mathbf{a}_i^{\text{child}} = \mathbf{x}_{ik^*}(t^*), \qquad d_i^{\max} > \tau_{\text{split}}. \tag{13}$$

where τ_{split} denotes the maximum tolerated normalized displacement. The child feature is zero-initialized to establish new local support for genuine motion while preserving anchor locality.

3.5 Geometric Regularization with Reliability and Keyframe Depth

To further improve geometric consistency, reinforce the locality prior, and provide more reliable geometric evidence for reliability modeling beyond sparse RGB supervision, we introduce complementary constraints at two levels: anchor-level locality and scene-level depth.

At the anchor level, we impose a reliability-weighted locality loss:

$$\mathcal{L}_{\text{local}} = \frac{\sum_{i,k} \tilde{\alpha}_{ik}(1-c_i)\left[\frac{\|\mathbf{x}_{ik}-\mathbf{a}_i\|_2}{\rho_i} - \tau_l\right]_+^2}{\sum_{i,k}\tilde{\alpha}_{ik} + \epsilon}. \tag{14}$$

where $\mathbf{x}_{ik}$ and $\tilde{\alpha}_{ik}$ denote the current decoded Gaussian center and raw opacity, respectively, while ρ_i, τ_l, and $[\cdot]_+$ follow equation 3. The loss constrains only displacements exceeding the locality-risk threshold, thereby preserving reasonable local variations within the anchor support. The factor $(1-c_i)$ further imposes stronger geometric regularization on less reliable anchors. During optimization, $\tilde{\alpha}_{ik}$ and c_i are detached, so gradients propagate only through the decoded Gaussian centers. This selectively pulls unreliable Gaussian centers back toward their local support while avoiding unnecessary constraints on valid local representations near the anchor.

At the scene level, we introduce keyframe-based DepthAnything V3 (DA3) (Lin et al., 2025) depth supervision to provide complementary geometric constraints. To reduce inference and storage overhead, we uniformly select a small number of temporal keyframes for depth prediction and apply supervision only when these frames are sampled. DA3 can incorporate camera parameters during inference, producing depth predictions $z^D(u)$ that are scale-consistent with the Gaussian-rendered depth $\hat{z}(u)$. Therefore, no additional depth normalization or scale alignment is required, and we directly adopt a log-depth Charbonnier loss:

$$\mathcal{L}_{\text{depth}} = \frac{1}{|\Omega|}\sum_{u\in\Omega}\sqrt{\left[\log(\hat{z}(u)+\epsilon) - \log(z^D(u)+\epsilon)\right]^2 + \delta^2}. \tag{15}$$

where Ω denotes the valid depth pixels and δ is the Charbonnier smoothing parameter. The log-depth formulation reduces the dominance of large depth values, while the Charbonnier penalty improves robustness to noisy depth predictions. During depth supervision, opacity gradients are detached so that the depth prior primarily constrains Gaussian geometry without altering opacity.

3.6 Loss Function

We retain $\mathcal{L}_1$, SSIM, and scale losses of Scaffold-GS, with added locality and depth regularization:

$$\mathcal{L} = (1-\lambda_{\text{SSIM}})\mathcal{L}_1 + \lambda_{\text{SSIM}}\mathcal{L}_{\text{SSIM}} + \lambda_{\text{scale}}\mathcal{L}_{\text{scale}} + \lambda_{\text{local}}\mathcal{L}_{\text{local}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}}. \tag{16}$$

where $\lambda_{(\cdot)}$ denotes the corresponding loss weights. $\mathcal{L}_{\text{depth}}$ is activated for the selected keyframes.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate our method on Neural 3D Video (Li et al., 2022), NVIDIA Dynamic Scenes (Yoon et al., 2020), and Technicolor (Sabater et al., 2017). For Neural 3D Video, following 4C4D (Zhou et al., 2026), we use four training views (**Cam01, Cam10, Cam13, Cam20**) and evaluate on the remaining views at half resolution. For NVIDIA Dynamic Scenes, following the resolution setting of SparseCam4D (Pan et al., 2026), we use half-resolution images, with three views

(**Cam01, Cam06, Cam10**) for training and the remaining views for evaluation. For Technicolor, following DASH (Chen et al., 2025), we use full-resolution images, with three views (**Cam00, Cam09, Cam15**) for training and the remaining views for evaluation.

Implementation Details. We train ReLo-S4DGS for 20K iterations with $K = 10$ Gaussians per anchor. For point-cloud initialization, ReLo-S4DGS and the compared methods use COLMAP reconstructions obtained from the sparse training views at frame 0. The only exception is 4C4D on Neural 3D Video, for which we retain its original MASt3R (Leroy et al., 2024) dense point-cloud initialization following the official setting (Zhou et al., 2026). Reliability evidence is accumulated during training, and the reliability state is updated every 500 iterations with the decay factor $\eta = 0.9$. We divide each sequence into 8 temporal regions and set $N_{\text{target}} = 3$, $\tau_l = 8$, $\tau_{\text{abs}} = 30$, and $\tau_{\text{rel}} = 0.25$. The local reference scale ρ_i is computed using KNN with $K_{\text{NN}} = 8$ and recomputed for all anchors after initialization and each DALC operation. RAOM and locality regularization warm up linearly over 8K–16K iterations, with $g_{\min} = 0.3$, $c_{\text{low}} = 0.3$, and $c_{\text{high}} = 0.8$. DALC is performed at 8K and 13K iterations with $P\% = 80\%$ and $\tau_{\text{split}} = \tau_l = 8$. For depth supervision, we use DA3-BASE and select keyframes every 15 frames starting from frame 0, with the Charbonnier parameter set to $\delta = 0.001$. We set $\lambda_{\text{SSIM}} = 0.2$, $\lambda_{\text{scale}} = 0.01$, $\lambda_{\text{local}} = 5 \times 10^{-7}$, and $\lambda_{\text{depth}} = 0.1$. More details are provided in the appendix.

4.2 Results

Quantitative Comparisons. We compare ReLo-S4DGS with the representative dynamic Gaussian method 4DGaussians (Wu et al., 2024), the recent DASH (Chen et al., 2025), the anchor-based LocalDyGS (Wu et al., 2025), and the sparse-view method 4C4D (Zhou et al., 2026). As shown in Tab. 1, ReLo-S4DGS achieves the best results on eight out of nine metrics across the three datasets, demonstrating consistent novel-view rendering performance under different sparse-view settings. On Neural 3D Video, our method achieves the best performance across all metrics, improving PSNR and SSIM over 4C4D by 1.06 dB and 0.0179, respectively. On NVIDIA Dynamic Scenes, ReLo-S4DGS ranks first across all three metrics. On Technicolor, our method achieves the best PSNR and SSIM while obtaining the second-best LPIPS. These quantitative results demonstrate the effectiveness of our reliability modeling and locality optimization for sparse-view dynamic reconstruction.

Qualitative Comparisons. As shown in Fig. 3, ReLo-S4DGS consistently produces cleaner and more coherent reconstructions under sparse-view supervision. On Neural 3D Video, 4DGaussians, DASH, and LocalDyGS exhibit noticeable noise and artifacts around the moving foreground and window regions, while 4C4D introduces pronounced streak-like artifacts under strong illumination. In contrast, our method preserves the dynamic foreground while reconstructing the background structures outside the window with substantially fewer artifacts. On NVIDIA Dynamic Scenes, existing methods show degraded facial and body structures, and DASH even fails to faithfully reconstruct the dynamic umbrella motion, whereas our method maintains a more complete and coherent reconstruction of both the person and the moving object. On Technicolor, ReLo-S4DGS likewise produces the cleanest scene structures with fewer spurious artifacts around object boundaries and background regions. These qualitative results demonstrate that our method maintains reliable local geometry under sparse observations without sacrificing dynamic modeling capability.

4.3 Ablation Study and Analysis

Component Analysis. We evaluate the contribution of each major component of ReLo-S4DGS on Neural 3D Video, as reported in Tab. 2 and Fig. 4. Removing any component consistently degrades reconstruction performance, demonstrating their complementary roles. Without Reliability-Aware Opacity Modulation, PSNR decreases from 23.43 dB to 23.11 dB, accompanied by stronger artifacts in poorly observed background regions. Removing Dynamic-Aware Locality Correction reduces PSNR and SSIM to 23.16 dB and 0.8343, respectively, and produces distorted local structures around object boundaries. Removing Anchor-Level Locality Regularization increases LPIPS from 0.1379 to 0.1592, with visibly degraded fine structures and background details. Removing Scene-Level Keyframe Depth Supervision degrades reconstruction performance and leads to less coherent geometric details. In contrast, the full model consistently reconstructs cleaner local structures across all examples and achieves the best performance on all three metrics, validating the complementary effects of reliability modeling, locality optimization, and geometric regularization.

Table 1: Quantitative comparisons on Neural 3D Video (Li et al., 2022), NVIDIA Dynamic Scenes (Yoon et al., 2020), and Technicolor (Sabater et al., 2017) datasets. The first and second best results are highlighted in red and yellow, respectively.

Method	Neural 3D Video			NVIDIA Dynamic Scenes			Technicolor		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
4DGaussians	21.19	0.8000	0.1741	23.37	0.7482	0.1529	18.76	0.5585	0.2957
DASH	20.49	0.7758	0.2032	17.34	0.4173	0.3249	21.97	0.7314	0.2277
LocalDyGS	21.93	0.8105	0.1629	22.73	0.7206	0.1489	21.34	0.7302	0.3001
4C4D	22.37	0.8206	0.1478	21.80	0.7171	0.1444	20.98	0.6594	0.2792
Ours	23.43	0.8385	0.1379	23.72	0.7520	0.1344	22.76	0.7547	0.2551



Figure 3: Visual comparison of results produced by different methods.

Analysis of Reliability Evidence. We further analyze the three evidence sources used in reliability modeling in Tab. 3. Removing any source degrades all evaluation metrics, indicating that spatial, temporal, and cross-view evidence provide complementary cues for estimating anchor reliability. Removing local geometric risk decreases PSNR from 23.43 dB to 23.17 dB, highlighting the importance of identifying Gaussian displacements that exceed local anchor support. Removing temporal repeatability increases LPIPS from 0.1379 to 0.1464, showing the benefit of persistent temporal sup-

Table 2: Component ablation study on the Neural 3D Video dataset.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
w/o Reliability-Aware Opacity Modulation	23.11	0.8337	0.1454
w/o Dynamic-Aware Locality Correction	23.16	0.8343	0.1424
w/o Anchor-Level Locality Regularization	23.03	0.8309	0.1592
w/o Scene-Level Keyframe Depth Supervision	22.92	0.8293	0.1513
Full Model	23.43	0.8385	0.1379

Table 3: Ablation study of evidence sources in beta-evidence reliability modeling on the Neural 3D Video dataset. "Local", "Temporal", and "Cross-view" denote local geometric risk, temporal repeatability, and cross-view optimization agreement respectively.

Method	Local	Temporal	Cross-view	PSNR ↑	SSIM ↑	LPIPS ↓
w/o Local		✓	✓	23.17	0.8357	0.1434
w/o Temporal	✓		✓	23.23	0.8352	0.1464
w/o Cross-view	✓	✓		23.24	0.8345	0.1433
Full Model	✓	✓	✓	23.43	0.8385	0.1379



(a) w/o Reliability-Aware Opacity Modulation (b) w/o Dynamic-Aware Locality Correction (c) w/o Anchor-Level Locality Regularization (d) w/o Scene-Level Keyframe Depth Supervision

Figure 4: Qualitative component ablation on Neural 3D Video.

port for reliable representation. Removing cross-view optimization agreement decreases SSIM from 0.8385 to 0.8345, demonstrating the value of consistent geometric updates across views. Combining all three evidence sources yields the best performance across all metrics, supporting the effectiveness of multi-source reliability modeling.

5 Conclusion

In this work, we address sparse-view dynamic scene reconstruction from the perspective of locality degradation in anchor-based Gaussian representations. We observe that insufficient spatial constraints can drive anchors to explain limited observations through non-local Gaussian displacements, weakening the locality prior and leading to unreliable representations. Based on this observation, we introduce ReLo-S4DGS, which performs reliability modeling for locality optimization under sparse supervision. By combining reliability-aware rendering regulation, dynamic-aware locality correction, and geometric regularization, our method suppresses unreliable non-local representations while preserving genuine scene dynamics. Extensive experiments on three standard dynamic scene benchmarks demonstrate consistent improvements over representative dynamic Gaussian and sparse-view reconstruction methods across varying sparse-view configurations, while ablation studies validate the effectiveness of the proposed components.

AI USE STATEMENT

Large language models are used only for language editing, including grammar correction, wording refinement, and improving readability. They are not used to generate technical contributions, experimental results, or scientific conclusions. All technical content is developed, reviewed, and verified by the authors.

REFERENCES

Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In *European Conference on Computer Vision (ECCV)*, 2024.

Jie Chen, Zhangchi Hu, Peixi Wu, Huyue Zhu, Hebei Li, and Xiaoyan Sun. Dash: 4d hash encoding with self-supervised decomposition for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26349–26359, 2025.

Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21476–21485, 2024. doi: 10.1109/CVPR52733.2024.02029.

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

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

Dongdong Hu, Yang Zhou, Xiaofeng Huang, Haibing Yin, and Zhu Li. Sparse4dgs: Flow-geometry assisted 4d gaussian splatting for dynamic sparse view synthesis. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10642–10651, 2025.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. doi: 10.1145/3641519.3657428.

Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. SplaTAM: Splat, track & map 3D gaussians for dense RGB-D SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`.

Seung-gyeom Kim, Areum Kim, Yongjae Yoo, and Sukmin Yun. Implicit 4d gaussian splatting for fast motion with large inter-frame displacements. In *International Conference on Learning Representations*, volume 2026, pp. 43331–43349, 2026.

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

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European conference on computer vision*, pp. 71–91. Springer, 2024.

Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20775–20785, June 2024a.

Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, and Zhaoyang Lv. Neural 3d video synthesis from multi-view video, 2022. URL https://arxiv.org/abs/2103.02597.

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

Jie Liang, Jiahao Wu, Chao Wang, Jiayu Yang, Xiaoyun Zheng, Kaiqiang Xiong, Zhanke Wang, Jinbo Yan, Feng Gao, and Ronggang Wang. Clipgstream: Clip-stream gaussian splatting for any length and any motion multi-view dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 41022–41032, June 2026.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21136–21145, June 2024. doi: 10.1109/CVPR52733.2024.01997.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20654–20664, 2024.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pp. 800–809, 2024. doi: 10.1109/3DV62453.2024.00044.

Marko Mihajlovic, Sergey Prokudin, Siyu Tang, Robert Maier, Federica Bogo, Tony Tung, and Edmond Boyer. SplatFields: Neural gaussian splats for sparse 3d and 4d reconstruction. In *European Conference on Computer Vision (ECCV)*. Springer, 2024.

Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, 2020.

Kazuki Ozeki, Shun Kenney, Yuto Shibata, Eisuke Takeuchi, Takuya Narihira, Kazumi Fukuda, Ryosuke Sawata, Yuki Mitsufuji, and Yoshimitsu Aoki. 4d reconstruction from sparse dynamic cameras, 2026. URL https://arxiv.org/abs/2606.04593.

Weihong Pan, Xiaoyu Zhang, Zhuang Zhang, Zhichao Ye, Nan Wang, Haomin Liu, and Guofeng Zhang. Sparsecam4d: Spatio-temporally consistent 4d reconstruction from sparse cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33237–33247, 2026.

Hyunwoo Park, Gun Ryu, and Wonjun Kim. Dropgaussian: Structural regularization for sparse-view gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21600–21609, 2025a.

Jongmin Park, Minh-Quan Viet Bui, Juan Luis Gonzalez Bello, Jaeho Moon, Jihyong Oh, and Munchurl Kim. Splinegs: Robust motion-adaptive spline for real-time dynamic 3d gaussians from monocular video. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pp. 26866–26875, June 2025b.

Shiwei Ren, Tianci Wen, Yongchun Fang, and Biao Lu. Fastgs: Training 3d gaussian splatting in 100 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26094–26103, June 2026.

Neus Sabater, Guillaume Boisson, Benoit Vandame, Paul Kerbiriou, Frederic Babon, Matthieu Hog, Remy Gendrot, Tristan Langlois, Olivier Bureller, Arno Schubert, et al. Dataset and pipeline for multi-view light-field video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1743–1753. IEEE, 2017.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025a.

Yifan Wang, Peishan Yang, Zhen Xu, Jiaming Sun, Zhanhua Zhang, Yong Chen, Hujun Bao, Sida Peng, and Xiaowei Zhou. Freetimegs: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In *CVPR*, 2025b. URL https://zju3dv.github.io/freetimegs.

Zihan Wang, Jeff Tan, Tarasha Khurana, Neehar Peri, and Deva Ramanan. Monofusion: Sparse-view 4d reconstruction via monocular fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8252–8263, October 2025c.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20310–20320, June 2024.

Jiahao Wu, Rui Peng, Jianbo Jiao, Jiayu Yang, Luyang Tang, Kaiqiang Xiong, Jie Liang, Jinbo Yan, Runling Liu, and Ronggang Wang. Localdygs: Multi-view global dynamic scene modeling via adaptive local implicit feature decoupling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9519–9529, 2025.

Zhen Xu, Yinghao Xu, Zhiyuan Yu, Sida Peng, Jiaming Sun, Hujun Bao, and Xiaowei Zhou. Representing long volumetric video with temporal gaussian hierarchy. *ACM Transactions on Graphics*, 43(6):1–18, 2024. doi: 10.1145/3687919.

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

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

Jae Shin Yoon, Kihwan Kim, Orazio Gallo, Hyun Soo Park, and Jan Kautz. Novel view synthesis of dynamic scenes with globally coherent depths from a monocular camera, 2020. URL https://arxiv.org/abs/2004.01294.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

Jiawei Zhang, Jiahe Li, Xiaohan Yu, Lei Huang, Lin Gu, Jin Zheng, and Xiao Bai. Cor-gs: Sparse-view 3d gaussian splatting via co-regularization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Junsheng Zhou, Zhifan Yang, Liang Han, Wenyuan Zhang, Kanle Shi, Shenkun Xu, and Yushen Liu. 4C4D: 4 camera 4d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 145–163, 2024. doi: 10.1007/978-3-031-72933-1_9.

A More Implementation Details

Reliability statistics and update. Reliability statistics are accumulated over non-overlapping update intervals of 500 training iterations. Let r index these update intervals. Within each interval, the quantities required for local geometric risk, temporal repeatability, and cross-view optimization agreement are accumulated from valid observations, while no EMA is applied to the per-iteration statistics. At the end of interval r, R_i, S_i, and G_i are computed to obtain the supporting and contradicting evidence E_i^+ and E_i^-.

To emphasize recent and sufficiently observed evidence, historical evidence is decayed and the evidence from each update interval is weighted according to its observation strength. Specifically, let $n_i^{(r)}$ denote the number of valid decoded-Gaussian observations received by anchor i within interval r. A decoded Gaussian is counted as a valid observation only if it passes the current opacity/AABB selection and has a positive screen-space radius after rasterization. Since $n_i^{(r)}$ counts decoded-Gaussian observations rather than visible frames, multiple active offset slots of the same anchor may contribute multiple observations within a single iteration. We define the observation strength as

$$\gamma_i^{(r)} = \min\left(1, \frac{n_i^{(r)}}{20}\right). \tag{17}$$

Thus, the contribution of the current evidence gradually increases with the number of valid observations and reaches full strength once $n_i^{(r)} \geq 20$.

For sufficiently observed anchors, the Beta-state update in equation 7 is applied using the evidence from the current interval, with $\eta = 0.9$. Anchors with $n_i^{(r)} < 5$ are regarded as insufficiently observed, and their Beta states are kept unchanged without applying historical decay. For anchors satisfying $n_i^{(r)} \geq 5$, the update is performed only when the current interval contains at least one valid supporting or contradicting evidence term.

These observation thresholds are introduced only to filter out extreme cases in which an anchor receives little or no effective supervision, rather than to define a finely tuned decision boundary. Since each update interval spans 500 training iterations, anchors that regularly contribute to rendering can typically accumulate sufficient decoded-Gaussian observations with ease. Consequently, the reliability update is largely insensitive to these thresholds within a reasonable range, while the filtering prevents unobserved anchors from accumulating spurious evidence or changing their reliability solely due to missing observations.

After each reliability update, the temporary statistics of the current interval are cleared, while the accumulated Beta state and the resulting posterior-mean reliability score are retained for subsequent training.

Anchor maintenance and locality correction. RAOM is activated after 8K iterations with a linear warm-up and reaches its full strength at 16K iterations. For each DALC operation, we first recompute the local reference scale ρ_i for all current anchors using the median distance to their $K_{\mathrm{NN}} = 8$ nearest neighbors, and then reevaluate anchor dynamicity using the current model. At 8K and 13K iterations, anchors are ranked by dynamicity and divided into the bottom $P\%$ low-dynamic anchors and the remaining high-dynamic anchors. For low-dynamic anchors, all K Gaussians are decoded at the reference time $t' = 0$. If the anchor lies outside their convex hull, its position is recentered to the raw-opacity-weighted mean of the decoded Gaussian centers. For high-dynamic anchors, we maintain the maximum decoded displacement between consecutive DALC iterations. An anchor is split when

$$\frac{d_i^{\max}}{\rho_i} > \tau_{\mathrm{split}}, \tag{18}$$

and the child anchor is placed at the corresponding maximally displaced Gaussian position with zero-initialized anchor features. After each DALC operation, the maximum-displacement statistics are cleared and accumulated again until the next DALC iteration.

Geometric supervision and training schedule. The reliability-weighted locality regularization follows the same linear warm-up as RAOM, starting after 8K iterations and reaching its full weight at 16K. During this loss, the opacity weights and anchor reliability are detached such that the regularization primarily optimizes Gaussian geometry.

For scene-level geometric supervision, depth priors are precomputed on temporal keyframes and incorporated through the log-depth Charbonnier loss described in the main paper. During depth rendering, opacity gradients are detached to prevent the depth prior from being satisfied by modifying opacity rather than geometry.

Anchor densification is performed every 100 iterations from 3K to 12K iterations. Densification is terminated before the second DALC operation at 13K, so that all anchors introduced during densification can still be included in the subsequent locality correction and have their spatial positions refined.

B Beta Evidence Modeling and Its Expectation

We aim to quantify how reliable the local representation of each anchor is during training. Such reliability cannot be directly determined from a single view or optimization step, since individual observations under sparse-view supervision can be noisy or view-dependent. Instead, it should be inferred from multiple supporting and contradicting cues aggregated over successive update intervals. We therefore model the reliability of anchor i as a latent variable $C_i \in [0, 1]$. For simplicity, we omit the update-interval index r in the following derivation.

This formulation is inspired by the Beta–Bernoulli evidence interpretation. In the classical setting, a Beta distribution models an unknown probability through accumulated positive and negative observations. Similarly, our reliability modeling provides two complementary types of evidence: supporting evidence E_i^+ and contradicting evidence E_i^-. Unlike binary Bernoulli observations, these cues are continuous-valued, and we therefore interpret them as fractional positive and negative evidence. Their accumulated evidence states are represented by the two positive parameters A_i and B_i, respectively, as defined in equation 7. We then model the latent anchor reliability as

$$C_i \sim \mathrm{Beta}(A_i, B_i), \tag{19}$$

where $A_i > 0$ and $B_i > 0$. Its probability density function is

$$p(C_i = c) = \frac{c^{A_i-1}(1-c)^{B_i-1}}{\mathrm{B}(A_i, B_i)}, \qquad 0 < c < 1, \tag{20}$$

where the Beta function is defined as

$$\mathrm{B}(a, b) = \int_0^1 x^{a-1}(1-x)^{b-1}\, dx = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}. \tag{21}$$

The Beta distribution thus provides a natural state for maintaining the competition between accumulated supporting and contradicting evidence. Increasing A_i shifts the reliability toward larger values, whereas increasing B_i favors smaller reliability values. Since subsequent optimization requires a scalar reliability score rather than the full distribution, we use the expectation of C_i.

According to the definition of expectation,

$$\mathbb{E}[C_i] = \int_0^1 c\, p(C_i = c)\, dc \tag{22}$$

$$= \int_0^1 c\, \frac{c^{A_i-1}(1-c)^{B_i-1}}{\mathrm{B}(A_i, B_i)}\, dc \tag{23}$$

$$= \frac{1}{\mathrm{B}(A_i, B_i)} \int_0^1 c^{A_i}(1-c)^{B_i-1}\, dc \tag{24}$$

$$= \frac{\mathrm{B}(A_i + 1, B_i)}{\mathrm{B}(A_i, B_i)}. \tag{25}$$

Table 4: Sensitivity analysis of key hyperparameters on the Neural 3D Video dataset.

Parameter	Value	PSNR ↑	SSIM ↑	LPIPS ↓
τ_l	6	23.52	0.8378	0.1414
	8	23.38	0.8366	0.1406
	10	23.45	0.8377	0.1421
τ_{abs}	10	23.34	0.8352	0.1451
	20	23.51	0.8397	0.1383
	30	23.38	0.8366	0.1406
	40	23.36	0.8355	0.1450
τ_{rel}	0.10	23.35	0.8360	0.1447
	0.25	23.38	0.8366	0.1406
	0.40	23.49	0.8372	0.1414
τ_{split}	6	23.38	0.8366	0.1440
	8	23.38	0.8366	0.1406
	10	23.41	0.8370	0.1435

Using the relation between the Beta and Gamma functions, together with

$$\Gamma(x+1) = x\Gamma(x), \tag{26}$$

we further obtain

$$\mathbb{E}[C_i] = \frac{\Gamma(A_i+1)\Gamma(B_i)}{\Gamma(A_i+B_i+1)} \frac{\Gamma(A_i+B_i)}{\Gamma(A_i)\Gamma(B_i)} \tag{27}$$

$$= \frac{A_i\Gamma(A_i)}{(A_i+B_i)\Gamma(A_i+B_i)} \frac{\Gamma(A_i+B_i)}{\Gamma(A_i)} \tag{28}$$

$$= \frac{A_i}{A_i+B_i}. \tag{29}$$

Therefore, we use

$$c_i = \mathbb{E}[C_i] = \frac{A_i}{A_i+B_i} \tag{30}$$

as the anchor-level reliability score. This expectation provides a bounded and smooth summary of the accumulated evidence state: stronger supporting evidence increases c_i toward one, whereas stronger contradicting evidence decreases it toward zero. It therefore provides a natural scalar reliability measure for the subsequent opacity modulation and locality regularization.

C Additional Ablation Results

Several hyperparameters in ReLo-S4DGS are introduced to convert continuous reliability and locality statistics into operational criteria during training. Importantly, these parameters are not intended to define precise semantic boundaries, but rather to identify broad structural tendencies, such as excessive non-local displacement, persistent temporal support, or relatively high anchor dynamicity. We therefore further examine whether the proposed method depends on careful parameter tuning.

We conduct parameter sensitivity experiments on Neural 3D Video by varying one hyperparameter at a time while keeping all other settings unchanged. Since evaluating every frame from all test cameras introduces substantial rendering overhead, we uniformly evaluate every 30th frame for each camera in these additional experiments.

As shown in Table 4, reconstruction quality remains highly stable under moderate variations of the tested parameters. In particular, changing τ_l, τ_{abs}, τ_{rel}, or τ_{split} around their default settings produces only small fluctuations in PSNR, SSIM, and LPIPS, and no single parameter value consistently dominates across all metrics. This suggests that the performance of ReLo-S4DGS is not tied to a



Figure 5: **Qualitative analysis of the dynamicity-based anchor partition under different P.** The first row shows the ground-truth and full-model renderings for reference. The remaining rows separately render the bottom $P\%$ low-dynamic anchors and the top $(100 - P)\%$ high-dynamic anchors for $P \in \{60, 70, 80, 90\}$. As P increases, the high-dynamic partition becomes progressively concentrated on moving foreground regions, while the low-dynamic partition consistently preserves most static scene content.

narrowly tuned operating point. Instead, these thresholds mainly determine a reasonable range in which the corresponding statistics trigger locality regulation or structural maintenance.

This behavior is consistent with the role of these statistics in our method. For example, τ_l does not attempt to determine an exact boundary between local and non-local Gaussians, but identifies sufficiently large deviations from the current anchor support. Similarly, τ_{split} is used to detect pronounced motion beyond the existing local support rather than to estimate a precise motion magnitude. Consequently, moderate changes to these thresholds alter the triggering boundary but preserve the underlying structural tendency captured by the statistics.

We further study this property for the dynamicity-based partition controlled by P. The dynamicity score D_i is designed to provide a relative ordering of anchors according to their motion characteristics, rather than a binary static/dynamic classification. Accordingly, P only specifies the percentile

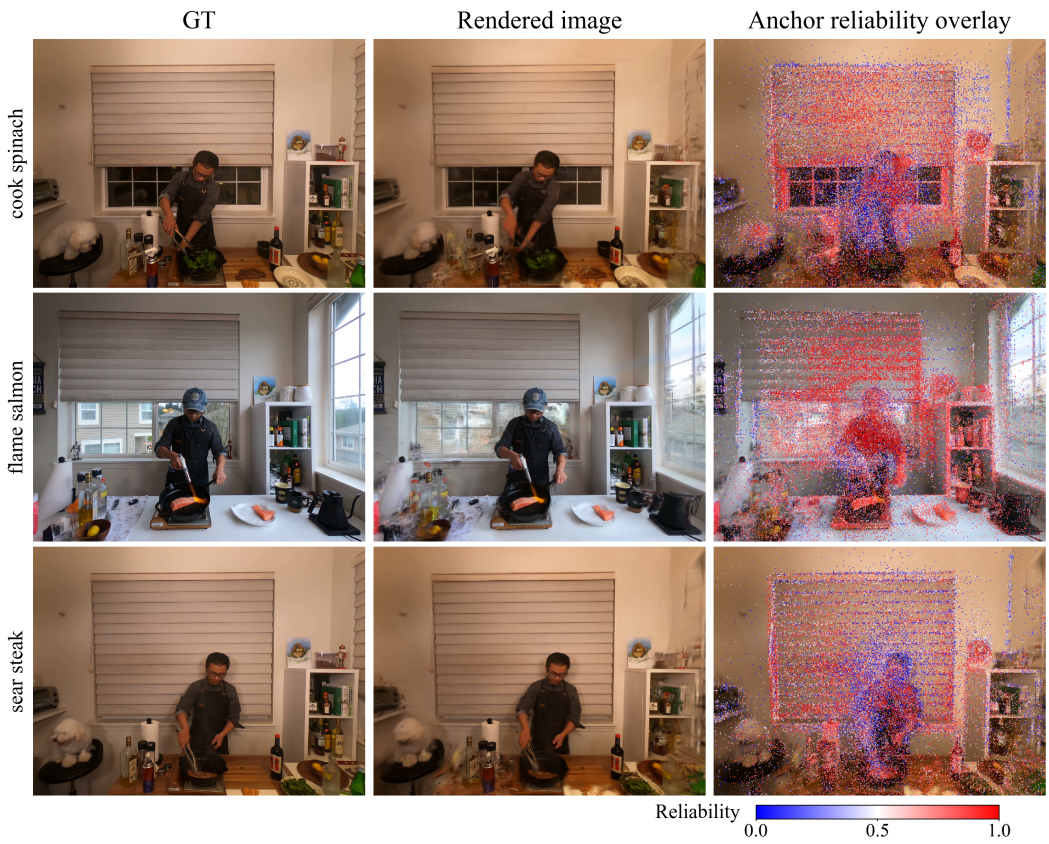

Figure 6: **Visualization of anchor reliability.** Colors from blue to red indicate increasing reliability. High-reliability anchors are concentrated in well-observed and structurally stable regions, while lower reliability appears more frequently in poorly reconstructed or weakly observed regions.

at which this ordering is partitioned. We visualize the bottom $P\%$ low-dynamic anchors and the remaining top $(100 - P)\%$ high-dynamic anchors for $P \in \{60, 70, 80, 90\}$.

As illustrated in Fig. 5, the qualitative separation evolves smoothly as P changes. Across both scenes, the bottom partition consistently preserves most of the static scene structure, while the top partition becomes progressively concentrated on the moving foreground as its size decreases. Importantly, changing P mainly adjusts the extent of the selected regions rather than changing their overall dynamic tendency. This indicates that the proposed dynamicity measure provides a stable relative ordering over anchors, making the subsequent partition less dependent on a particular percentile choice. Accordingly, we use $P = 80\%$ as a representative operating point, because it preserves most static scene content in the low-dynamic partition while retaining a concentrated high-dynamic partition around the moving foreground.

Together, the quantitative sensitivity analysis and qualitative dynamicity visualization show that the proposed statistics primarily capture robust structural trends rather than relying on precisely calibrated thresholds. The default hyperparameters are therefore used as representative operating points, rather than values obtained through exhaustive per-scene tuning.

D Further Analysis of Anchor Reliability

We further analyze whether the estimated anchor reliability reflects meaningful structural properties and rendering importance. Figure 6 visualizes the spatial distribution of anchor reliability, where colors from blue to red indicate increasing reliability. High-reliability anchors are mainly distributed

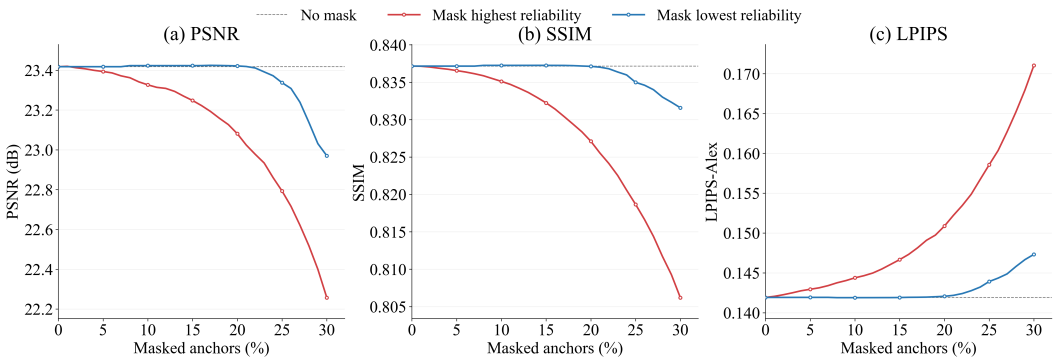

Figure 7: **Rendering quality under reliability-based anchor masking.** Anchors are progressively masked from the highest or lowest reliability values. Removing high-reliability anchors leads to substantially faster degradation in PSNR, SSIM, and LPIPS, whereas removing low-reliability anchors has a smaller initial impact on rendering quality.

over well-observed and stable structures, such as cabinets, tabletops, and window blinds visible from multiple training views. In contrast, regions with poorer reconstruction quality or weaker observation support, such as the side windows, contain noticeably fewer high-reliability anchors. This spatial distribution suggests that the estimated reliability is consistent with the stability and observability of local representations.

To further examine the relationship between reliability and rendering quality, we rank anchors by reliability and progressively mask them from either end of the ranking. As shown in Fig. 7, masking the highest-reliability anchors causes substantially faster degradation in PSNR and SSIM and a more rapid increase in LPIPS. In contrast, masking the lowest-reliability anchors has only a limited effect at small masking ratios. These results indicate that high-reliability anchors contribute more critically to the final rendering, further supporting the effectiveness of the proposed reliability modeling in identifying important local representations.

The relatively small impact of removing low-reliability anchors also suggests potential redundancy among these anchors. This observation motivates future exploration of reliability-based pruning to obtain more compact scene representations with limited degradation in rendering quality.