TAG4D-SLAM: TRACK-ANCHORED 4D GAUSSIAN SLAM WITH FEED-FORWARD PRIORS

Anonymous authors
Paper under double-blind review

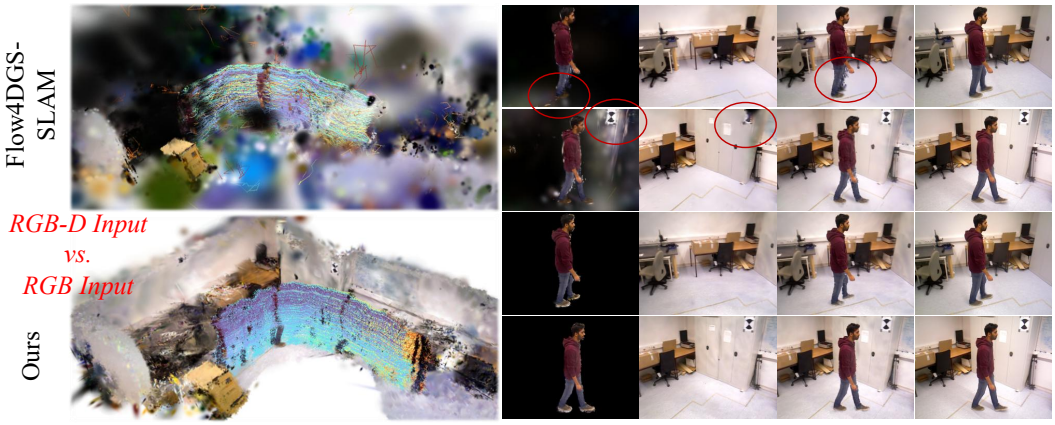

Figure 1: Qualitative results in dynamic scenes. Compared to Flow4DGS-SLAM, our method yields more reasonable appearance and geometry in static regions, and further models dynamic content with more consistent motion. Moreover, it achieves accurate dynamic-static separation and high-fidelity rendering in both regions.

ABSTRACT

Dynamic scenes remain a fundamental challenge for visual SLAM: moving objects violate the static-world assumption, while simply filtering them out discards valuable scene content. To address this, we present **TAG4D-SLAM**, a monocular dynamic Gaussian SLAM framework that jointly performs accurate camera tracking and high-fidelity 4D mapping with priors from the feed-forward reconstruction model. Specifically, we augment the feed-forward model with dynamic prediction, providing pose, depth, confidence, Gaussians, and dynamic probability for SLAM. Going beyond its explicit predictions, we further exploit the latent cross-view geometry encoded in internal attention to perform correspondence-aware patch sampling (CAPS) for camera tracking, turning implicit geometric knowledge into reliable constraints for bundle adjustment. For static mapping, prior Gaussians are compacted through voxelization and incrementally consolidated into a persistent map according to scene coverage. For dynamic content, rather than rejecting moving observations or relying on frame-local optical flow propagation and learned deformation fields, we introduce Track-Anchored Gaussians (TAGs) that anchor canonical dynamic Gaussians to long-lived 3D motion tracks. This decouples persistent scene content from temporal motion and establishes a persistent motion scaffold across frames, enabling more stable temporal association under large motion and occlusion. The resulting system yields accurate dynamic perception, robust camera tracking, and high-quality 4D reconstruction from monocular RGB input. Experiments on challenging dynamic datasets demonstrate our method's state-of-the-art performance in both camera tracking and 4D reconstruction.

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.

1 INTRODUCTION

Recent advances in 3D Gaussian Splatting enable explicit scene representations with high-quality rendering and efficient differentiable rasterization Kerbl et al. (2023), motivating a growing family of Gaussian-based SLAM systems Matsuki et al. (2024); Keetha et al. (2024); Yan et al. (2024a); Sandström et al. (2025). Most of these systems, however, inherit the static-world assumption. Moving objects introduce inconsistent geometric constraints that degrade camera tracking and reconstruction. Dynamic-robust methods therefore often detect and suppress moving observations Xu et al. (2024b); Wen et al. (2025), improving localization at the cost of discarding dynamic scene content.

Recent dynamic SLAM methods instead reconstruct moving content as part of the map. 4DGS-SLAM Li et al. (2025b) represents motion with sparse control points and a learned deformation field, while Flow4DGS-SLAM Wang & Lee (2026) propagates Gaussian centers using flow priors. Although effective, their temporal consistency relies on learned deformation or local motion propagation, which can become unreliable under large motion and occlusion; both also require RGB-D input. This leaves a need for a monocular representation that preserves dynamic content while maintaining stable long-term temporal association.

In parallel, feed-forward reconstruction models provide strong priors for scene geometry Wang et al. (2024); Leroy et al. (2024); Wang et al. (2025a;b). These priors have been integrated into SLAM by recent works, such as Murai et al. (2025); Maggio et al. (2026); Zhang et al. (2026). In dynamic scenes, π^3_{mos}-SLAM Zhong et al. (2026) further predicts masks to reject dynamic measurements during tracking, but does not persistently reconstruct the moving content. Moreover, existing systems mainly consume the explicit outputs of feed-forward models, leaving the cross-view geometric structure encoded in their internal attention largely unused.

To tackle these challenges, we introduce **TAG4D-SLAM**, which estimates camera poses from monocular RGB and maintains a persistent 4D Gaussian map of both static structure and moving objects, guided by feed-forward priors. The feed-forward model predicts camera pose, depth, confidence, Gaussian primitives, and dynamic probabilities, all of which are sufficiently exploited in the SLAM pipeline. For camera tracking, we introduce Correspondence-Aware Patch Sampling (CAPS), which exploits latent geometric knowledge encoded in the feed-forward reconstruction model to prioritize reliable static regions with cross-frame correspondence for patch-based bundle adjustment. For static mapping, excessively redundant prior Gaussians in static regions are first voxelized to compact sets and then adaptively injected into global static Gaussians map based on coverage cues. For dynamic scene content modeling, we introduce Track-Anchored Gaussians (TAGs) that decouple persistent scene content from temporal motion. Canonical dynamic Gaussians are bound to long-lived 3D tracks, whose displacements explicitly drive their motion. By anchoring transient observations to a persistent motion scaffold, this decoupling enables more stable long-term temporal association under large motion and occlusion, without repeated cross-frame Gaussian matching or per-Gaussian motion estimation. In addition, we introduce a temporal Fourier basis that models appearance variation, enabling dynamic Gaussians to capture appearance changes over time. Our contributions are summarized as follows:

- We present **TAG4D-SLAM**, a monocular dynamic Gaussian SLAM framework for joint camera tracking and persistent 4D scene reconstruction from RGB video. It transforms transient feed-forward scene predictions into geometrically consistent tracking constraints and persistent Gaussian representations, enabling accurate camera motion estimation and high-fidelity reconstruction of both static and dynamic scene content.
- We introduce **Correspondence-Aware Patch Sampling (CAPS)**, which exposes the latent cross-view geometric information encoded in the internal attention of a feed-forward reconstruction model and converts it into a sampling prior for bundle adjustment. CAPS guides geometric optimization toward static patches with distinctive and reliable cross-view evidence, providing more informative constraints for camera pose estimation.
- We propose **Track-Anchored Gaussians (TAGs)**, a dynamic representation that separates persistent scene content from its time-varying motion through shared trajectory anchors. By using long-lived 3D tracks to govern the evolution of canonical Gaussians, TAGs establish persistent temporal structure across frames and improve dynamic reconstruction under large motion and partial occlusion without repeated Gaussian matching or independent motion estimation for individual primitives.

2 Related Work

Dynamic-Robust Visual SLAM. Most visual SLAM systems rely on the static-world assumption and suffer from degraded localization and mapping accuracy in the presence of dynamic objects. Early methods such as DynaSLAM Bescos et al. (2018) and ReFusion Palazzolo et al. (2019) detect and suppress dynamic observations via semantic or geometric cues. RoDyn-SLAM Jiang et al. (2024) further utilizes optical-flow information, while DVN-SLAM Wu et al. (2025b) instead uses a local-global neural implicit representation and remains robust without such priors. With 3D Gaussian Splatting, DG-SLAM Xu et al. (2024b) employs motion masking and hybrid pose optimization, whereas Gassidy Wen et al. (2025) detects dynamic instances from temporal variations in rendering losses. More recent methods such as WildGS-SLAM Zheng et al. (2025a) and DROID-W Li et al. (2026) further reduce reliance on explicit motion priors by uncertainty estimation. These methods primarily use dynamic reasoning to protect camera estimation and static reconstruction from moving observations. In contrast, our method retains dynamic observations as persistent scene content.

SLAM with Dynamic Scene Modeling. Beyond treating moving objects as outliers, another line of research explicitly models dynamic content. Co-Fusion Rünz & Agapito (2017), MaskFusion Rünz et al. (2018), and DynaSLAM II Bescos et al. (2021) maintain object-level representations to jointly estimate camera motion, object motion, and scene structure. DynaMoN Schischka et al. (2025) combines motion-aware camera tracking with dynamic neural radiance-field reconstruction. Recent Gaussian-based approaches further enable photorealistic dynamic mapping. CAD-SLAM Wu et al. (2025a) detects dynamics from spatial-temporal inconsistencies and introduces decoupled mapping with a temporal Gaussian model. PG-SLAM Li et al. (2025a) categorizes dynamic objects into rigid and non-rigid categories, using the SMPL model to explicitly capture non-rigid human motion. 4DGS-SLAM Li et al. (2025b) models dynamic motion using sparse control points with an MLP-based deformation field under optical-flow supervision, whereas Flow4DGS-SLAM Wang & Lee (2026) propagates explicit Gaussian centers using scene-flow priors. Despite these advances, existing methods typically maintain temporal consistency through object models, learned deformation fields, or local Gaussian propagation, which can be fragile under large motion and occlusion. In contrast, we use long-lived 3D tracks as temporal anchors to drive canonical Gaussians, avoiding repeated Gaussian matching and per-Gaussian motion estimation.

SLAM with Feed-forward 3D Foundation Models. Recent feed-forward 3D foundation models have provided powerful geometric priors for visual SLAM. DUSt3R Wang et al. (2024) and MASt3R Leroy et al. (2024) directly infer dense 3D geometry from image pairs, while VGGT Wang et al. (2025a) extends this paradigm toward multi-view reconstruction. Building on these priors, MASt3R-SLAM Murai et al. (2025) integrates learned pointmap correspondences with local mapping, loop closure, and global optimization. VGGT-SLAM Maggio et al. (2026) further constructs and aligns VGGT-based submaps for globally consistent tracking. For Gaussian mapping, Flash-Mono Zhang et al. (2026) employs CUT3R Wang et al. (2025b) to stream camera poses and Gaussian primitives, combined with loop closure and global $\mathrm{Sim}(3)$ optimization. Despite significant progress, these approaches primarily target static scenes. Toward dynamics, π^3_{mos}-SLAM Zhong et al. (2026) extends π^3 Wang et al. (2026) with a moving-object segmentation head and couples its depth with patch-wise bundle adjustment in DPVO Teed et al. (2023). However, its segmentation is primarily used to exclude dynamic regions from patch sampling, rather than modeling dynamic content. In contrast, our method goes beyond explicit feed-forward predictions by exploiting internal geometric attention for correspondence-aware tracking, while further organizing transient dynamic observations into a persistent 4D representation.

3 Method

3.1 Overview

Given a monocular RGB sequence $\mathcal{I} = \{I_t\}_{t=1}^{T}$, we jointly estimate the camera trajectory and build a persistent Gaussian representation of the dynamic scene. As illustrated in Fig. 2, the framework proceeds in three stages: feed-forward scene prediction, correspondence-aware camera tracking, and persistent static-dynamic Gaussian mapping. At each incoming frame I_t, a feed-forward reconstruction model consumes a causal window $\mathcal{W}_t$ and predicts camera pose, depth, confidence,

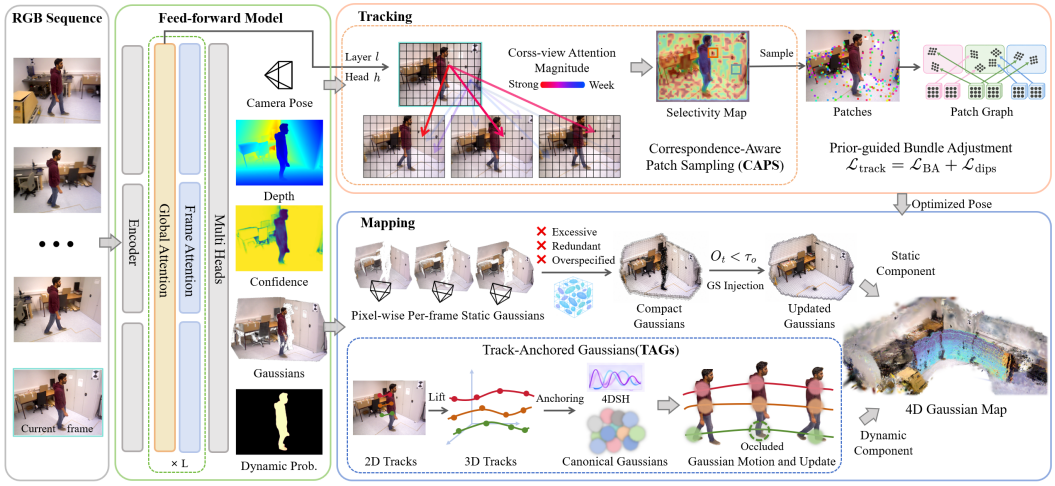

Figure 2: Overview of the proposed TAG4D-SLAM framework. Given a monocular RGB sequence, a feed-forward model predicts geometric priors and dynamic probabilities. Correspondence-Aware Patch Sampling (CAPS) exploits the model's internal cross-view attention to select informative static patches for prior-guided bundle adjustment. For static mapping, prior Gaussians are compacted through voxelization and integrated into a persistent map via coverage-aware insertion. For dynamic mapping, Track-Anchored Gaussians (TAGs) bind canonical Gaussians to persistent 3D motion tracks, decoupling scene content from temporal motion. The static and dynamic components jointly form a persistent 4D Gaussian map.

Gaussian primitives, and dynamic probability. These priors are then scale-aligned and fed into the SLAM pipeline. Camera tracking exploits the model's internal cross-view attention to select informative and correspondence-aware patches for bundle adjustment. For static mapping, we voxelize the prior Gaussians to compact sets and incrementally inject them into the uncovered regions. For dynamic mapping, we anchor canonical dynamic Gaussians to long-lived 3D motion tracks, and continuously update their geometry and appearance. Together, the framework yields accurate poses and high-fidelity 4D mapping of the whole dynamic scene.

3.2 Feed-forward Priors

Building on WorldMirror Liu et al. (2025), our feed-forward model jointly predicts camera poses, depths, and 3D Gaussian primitives. We further train a dynamic head that reads the multi-view input context and infers per-pixel dynamic probabilities. For each incoming frame I_t, we assemble a causal window of w frames $\mathcal{W}_t = \{k_1, \ldots, k_{w-1}, t\}$ for the feed-forward model,

$$\mathcal{F}(\mathcal{W}_t) = \left\{ \hat{\mathbf{T}}_i,\, \hat{\mathbf{D}}_i,\, \hat{\mathbf{C}}_i,\, \hat{\mathcal{G}}_i,\, \mathbf{P}_i^{\text{dyn}} \right\}_{i \in \mathcal{W}_t}, \tag{1}$$

where the terms, from left to right, are the predicted camera pose, depth, confidence, Gaussian primitives, and the per-pixel dynamic probability from the dynamic head. Since these priors are recovered up to an arbitrary scale, we estimate a relative factor s_t by IRLS-based robust alignment and apply it to $\hat{\mathbf{T}}_t$, $\hat{\mathbf{D}}_t$, and $\{\hat{\mathcal{G}}_i\}_{i \in \mathcal{W}_t}$, yielding the scale-aligned quantities denoted as $(\cdot)^a$.

3.3 Correspondence-Aware Patch Sampling for Camera Tracking

Correspondence-Aware Patch Sampling. Feed-forward reconstruction models pretrained on diverse data have demonstrated remarkable geometric capabilities. Beyond their final predictions, these models exhibit an intriguing internal property as reported in Bratulić et al. (2025); You et al. (2026); Kim et al. (2026): certain global attention layers and heads learn highly selective cross-frame interactions, naturally focusing on regions with consistent correspondence across views. This indicates that the learned geometric knowledge is not only encoded in the output, but also emerges as structured patterns within the model itself.

Motivated by this observation, we introduce Correspondence-Aware Patch Sampling (CAPS), which extracts cross-view selectivity from internal attention and uses it as a prior to guide patch selection. By directing the Bundle Adjustment towards regions with high selectivity, the tracker obtains more reliable geometric constraints and improves the overall tracking performance.

For a selected global attention layer ℓ and head h, let $\mathbf{q}_a^{(\ell,h)}$ and $\mathbf{k}_b^{(\ell,h)}$ denote the query and key vectors of tokens a and b, respectively, and let $f(a) \in \mathcal{W}_t$ denote the frame containing token a. The scaled dot-product attention is

$$B_{ab}^{(\ell,h)} = \mathrm{softmax}_b \left(\frac{\mathbf{q}_a^{(\ell,h)\top} \mathbf{k}_b^{(\ell,h)}}{\sqrt{d_h}} \right), \tag{2}$$

where d_h is the attention head dimension. We then aggregate the selected heads as $B_{ab}^{(\ell)} = \frac{1}{|\mathcal{H}|} \sum_{h \in \mathcal{H}} B_{ab}^{(\ell,h)}$. Focusing on cross-frame correspondence, we therefore remove intra-frame interactions and define the set of cross-frame keys for query a as $\mathcal{J}_a = \{b \mid f(b) \neq f(a)\}$. The remaining attention mass is renormalized into a conditional cross-view distribution,

$$p_{ab}^{(\ell)} = \frac{B_{ab}^{(\ell)} \mathbf{1}[b \in \mathcal{J}_a]}{\sum_{b' \in \mathcal{J}_a} B_{ab'}^{(\ell)}}. \tag{3}$$

A query token with a distinctive correspondence in other frames typically concentrates its attention on a small set of locations, whereas ambiguous or weakly corresponding regions yield a more diffuse distribution. We quantify this ambiguity using Shannon entropy,

$$H_a^{(\ell)} = -\sum_{b \in \mathcal{J}_a} p_{ab}^{(\ell)} \log p_{ab}^{(\ell)}, \tag{4}$$

and convert it into a normalized selectivity score $A_a^{(\ell)} = 1 - \frac{H_a^{(\ell)}}{\log |\mathcal{J}_a|}$. We reshape the resulting scores into a patch-level selectivity map $\mathbf{A}_t^{(\ell)}$ for the current frame and average the selectivity maps over the selected layers $\mathbf{A}_t = \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \mathbf{A}_t^{(\ell)}$. A higher value in $\mathbf{A}_t$ indicates that the feed-forward model assigns more concentrated cross-view attention to this region, implying a stronger and less ambiguous correspondence across frames. Therefore, CAPS prioritizes these regions to provide more reliable geometric constraints for bundle adjustment.

We then sample N_p patches in total from the static region $\mathcal{U}_t = \{\mathbf{u} \mid M_t^{\mathrm{dyn}}(\mathbf{u}) = 0\}$, where $M_t^{\mathrm{dyn}} = \mathbf{1}[\mathbf{P}_t^{\mathrm{dyn}} > \varepsilon]$ is the dynamic mask. A fraction ρ of the budget is sampled with probability proportional to $\mathbf{A}_t$,

$$p_t^{\mathrm{caps}}(\mathbf{u}) = \frac{\mathbf{A}_t(\mathbf{u})}{\sum_{\mathbf{v} \in \mathcal{U}_t} \mathbf{A}_t(\mathbf{v})}, \qquad \mathbf{u} \in \mathcal{U}_t. \tag{5}$$

Bundle Adjustment. Let k index a patch, $\mathbf{p}_k$ and d_k denote its center and optimized depth, d_k^a the corresponding sample from $\mathbf{D}_t^a$, and $\mathbf{z}_{ijk}$ the correspondence predicted by the DPVO update network between frames i and j. We optimize

$$\mathcal{L}_{\mathrm{track}} = \sum_{(i,j,k) \in \mathcal{E}} w_{ijk} \left\| \pi(\mathbf{T}_j \mathbf{T}_i^{-1}, \mathbf{p}_k, d_k) - \mathbf{z}_{ijk} \right\|_2^2 + \sum_k \omega_k \left\| d_k - (d_k^a) \right\|_2^2, \tag{6}$$

where w_{ijk} is the correspondence confidence predicted by the DPVO and ω_k is the weight for the depth regularization following π^3_{mos}-SLAM. Camera poses and patch depths are jointly optimized, yielding the refined camera trajectory $\{\mathbf{T}_t\}_{t=1}^T$ used for subsequent Gaussian mapping.

3.4 Voxelized Prior-Initialized Static Mapping

Voxelization and Coverage-Aware Injection. The windowed Gaussians $\{\mathcal{G}_i^a\}_{i \in \mathcal{W}_t}$ are predicted per pixel, so each primitive inherits the confidence and dynamic probability of its supporting pixel. Dynamic observations are first removed, $\mathcal{G}_{\mathcal{W}_t}^s = \bigcup_{i \in \mathcal{W}_t} \{g_{i,\mathbf{u}} \in \mathcal{G}_i^a \mid M_i^{\mathrm{dyn}}(\mathbf{u}) = 0\}$, where $g_{i,\mathbf{u}}$ denotes the Gaussian at pixel $\mathbf{u}$ of frame i. The filtered static Gaussians remain excessive and

redundant. We therefore voxelize $\mathcal{G}^s_{\mathcal{W}_t}$ with grids of size ℓ^a, following AnySplat Jiang et al. (2025), keep the voxels touched by the current frame, and take their aggregated primitives as $\mathcal{G}^a_t$. This compact set is then injected into the static map with a coverage check: a candidate is inserted only if its projection falls where the rendered opacity satisfies $O_t^{static} < \tau_o^s$, indicating that the region is under-represented in the existing map.

Static Map Refinement. After the new Gaussians are injected, we then optimize the static map with photometric and depth supervision: $\mathcal{L}^s_{\text{map}} = \lambda_{\text{rgb}}\mathcal{L}^s_{\text{rgb}} + (1-\lambda_{\text{rgb}})\mathcal{L}^s_{\text{depth}} + \lambda_{\text{size}}\mathcal{L}_{\text{size}}$, where $\mathcal{L}^s_{\text{rgb}}$ combines masked ℓ_1 and SSIM terms, $\mathcal{L}^s_{\text{depth}}$ constrains the rendered depth with $\mathbf{D}^a_t$, and $\mathcal{L}_{\text{size}}$ regularizes overly large Gaussians. Feed-forward initialization removes the need to optimize the static map from scratch and yields high-fidelity reconstruction with lightweight optimization, producing a compact and persistent static map.

3.5 Track-Anchored Dynamic Gaussian Mapping

Prior Gaussians in dynamic regions provide reliable representations at individual instants, but reconstructing them independently frame by frame cannot capture how the scene evolves over time. Naively matching individual Gaussians across frames, on the other hand, is unreliable under occlusion and long-term drift. We therefore introduce Track-Anchored Gaussians (TAGs) that separate *what it is* from *how it moves* with two complementary components: a canonical Gaussian set accumulates persistent scene content, while long-lived 3D tracks capture its motion over time. Anchored to these tracks, canonical Gaussians move with time and maintain a persistent representation, allowing frame-wise observations to be consistently associated and accumulated into a dynamic map.

Incremental 3D Motion Tracks. We employ Track-On-R Aydemir et al. (2026) to maintain 2D trajectories over dynamic regions indicated by M_t^{dyn}. For a tracked point $\mathbf{u}_t$ with its scale-aligned depth $\mathbf{D}^a_t(\mathbf{u}_t)$ and optimized camera pose $\mathbf{T}_t^{\text{c2w}}$, we lift its observation into the world frame as

$$\mathbf{x}_{n,t} = \mathbf{T}_t^{\text{c2w}} \pi^{-1}(\mathbf{u}_{n,t}, \mathbf{D}^a_t(\mathbf{u}_{n,t}))\,, \tag{7}$$

where π^{-1} denotes back-projection and n indexes the track. Collecting these 3D positions over time gives the tracklet $\mathcal{T}_n = \{\mathbf{x}_{n,t}\}_t$, which serves as a persistent 3D motion anchor well aligned with the SLAM coordinate system. Each track is initialized at $\bar{\mathbf{x}}_n$. When dynamic regions appear outside the coverage of existing tracks, we introduce additional tracking queries in these uncovered regions to monitor the newly visible content. These incrementally extended tracks underpin a dynamically maintained motion scaffold, providing accurate and consistent 3D motion.

Track-Anchored Canonical Gaussians. Instead of assigning temporal correspondence to Gaussian primitives themselves, we represent dynamic content with an isotropic canonical Gaussian set $\bar{\mathcal{G}}^{dyn} = \{\bar{g}_j\}$. When Gaussian j is added at birth time b_j, it is first bound to the nearest 3D motion track $n(j) \in \mathcal{T}$ and its canonical center $\bar{\boldsymbol{\mu}}_j$ is initialized by canceling the track's displacement:

$$\bar{\boldsymbol{\mu}}_j = \boldsymbol{\mu}_j(b_j) - \left(\mathbf{x}_{n(j),b_j} - \bar{\mathbf{x}}_{n(j)}\right). \tag{8}$$

At time t, its center $\boldsymbol{\mu}_j(t)$ is recovered by adding the canonical center to the displacement of the track:

$$\boldsymbol{\mu}_j(t) = \bar{\boldsymbol{\mu}}_j + \left(\mathbf{x}_{n(j),t} - \bar{\mathbf{x}}_{n(j)}\right). \tag{9}$$

In this way, canonical Gaussians store persistent geometry, while 3D tracks explicitly provide their time-varying motion. By avoiding repeatedly matching transient Gaussians or learning motion solely from rendering residuals, we achieve consistent and occlusion-robust dynamic modeling.

Temporal Appearance Modeling. The appearance of dynamic scene content may vary over time even when its geometry is consistently tracked. Following StreetGaussians Yan et al. (2024b), We model the appearance of each canonical Gaussian with 4D Spherical Harmonics (4DSH):

$$\mathbf{f}_j(t) = \sum_{k=0}^{K_a-1} \mathbf{a}_{j,k}\, \phi_k(\tau_t), \tag{10}$$

where τ_t is normalized time within the Gaussian's valid interval and ϕ_k denotes the Fourier basis functions. The constant term is initialized from the prior Gaussian at birth time, while the remaining coefficients model temporal appearance variation.

Dynamic Map Update and Refinement. To sustain the dynamic map over time, we update dynamic Gaussians with coverage-aware injection similar to static mapping. Specifically, after deforming the canonical Gaussians to the current time, we render them to obtain the accumulated opacity O_t^{dyn} and insert new candidates from prior Gaussians only at regions where $O_t^{dyn} < \tau_o^d$. Each canonical Gaussian maintains a valid temporal interval and is rendered only when its associated track provides reliable support. Variable visibility of the canonical Gaussians helps to provide clear dynamic rendering results without discarding the canonical parameters when occlusion occurs.

We refine the dynamic map with the same losses as in Sec. 3.4 and only the canonical Gaussian parameters are optimized, keeping the tracks and the Gaussian-to-track associations fixed. Therefore, the rendering loss focuses on refining the canonical content without affecting the dynamic behavior provided by the motion scaffold.

4 Experiments

4.1 Experimental Setup

Datasets. DAVIS-16 and DAVIS-17 Perazzi et al. (2016); Pont-Tuset et al. (2017) are used to evaluate dynamic object segmentation from our dynamic head. For camera tracking and dynamic scene reconstruction, we evaluate on Bonn RGB-D Palazzolo et al. (2019), and Wild-SLAM MoCap Zheng et al. (2025a) datasets without using ground truth depth.

Evaluation Metrics. For moving-object segmentation, we follow the DAVIS protocol and report $\mathcal{J}_{\mathrm{M}}$ and $\mathcal{J}_{\mathrm{R}}$, measuring mean region IoU and the percentage of per-frame $\mathcal{J}$ scores above 0.5, respectively. For camera tracking, we report the root-mean-square absolute trajectory error (ATE RMSE) in centimeters. For rendering quality, we use PSNR, SSIM, and LPIPS. Across all result tables, we highlight the **first**, second, and third best results.

Baselines. For motion segmentation, we compare with feed-forward reconstruction methods including DUSt3R Wang et al. (2024), MonST3R Zhang et al. (2024), DAS3R Xu et al. (2024a), and Easi3R Chen et al. (2025), as well as π^3_{mos}-SLAM Zhong et al. (2026). For tracking, we evaluate against both RGB-D and RGB-only SLAM methods. In particular, WildGS-SLAM Zheng et al. (2025a), DROID-W Li et al. (2026), and π^3_{mos}-SLAM are strong monocular baselines for dynamic scenes. For photometric reconstruction, we compare with WildGS-SLAM Zheng et al. (2025a) for novel view synthesis in the static map. We further compare with RGB-D input dynamic SLAM methods including 4DGS-SLAM Li et al. (2025b) and Flow4DGS-SLAM Wang & Lee (2026) for dynamic Gaussian mapping.

Implementation Details. Our feed-forward reconstruction model is initialized from the pretrained WorldMirror Liu et al. (2025) weights with only the dynamic head being extended and trained. For CAPS, we use global attention block 14 with heads $\{1, 4, 10\}$ inspired by Bratulić et al. (2025) and sample $N_p = 128$ tracking patches each frame. For mapping, the static and dynamic opacity thresholds are set to $\tau_o^s = 0.9$ and $\tau_o^d = 0.85$, respectively. The voxel size is set to $\ell = 0.003$ in feed-forward units. Temporal appearance is represented using $K_a = 9$ harmonic coefficients. All experiments are conducted on a workstation equipped with an Intel Xeon Gold 6530 CPU and an NVIDIA GeForce RTX 4090 GPU. Further implementation details are provided in the Appendix.

4.2 Results

Table 1: Moving-object segmentation on DAVIS-16 and DAVIS-17.

Method	DAVIS-16		DAVIS-17	
	$\mathcal{J}$M↑	$\mathcal{J}$R↑	$\mathcal{J}$M↑	$\mathcal{J}$R↑
DUSt3R Wang et al. (2024)	42.1	45.7	35.2	35.3
MonST3R Zhang et al. (2024)	40.9	42.2	38.6	38.2
DAS3R Xu et al. (2024a)	41.6	39.0	43.5	42.1
Easi3R$_{\mathrm{DUSt3R}}$ Chen et al. (2025)	53.1	60.4	49.0	56.4
Easi3R$_{\mathrm{MonST3R}}$ Chen et al. (2025)	57.7	71.6	56.5	68.6
π^3_{mos}-SLAM Zhong et al. (2026)	68.1	78.3	70.6	81.3
Ours	**80.1**	**91.8**	**78.2**	**89.8**

Moving-Object Segmentation. Table 1 evaluates our dynamic head. Our method consistently outperforms all baselines. It improves $\mathcal{J}_{\mathrm{M}}/\mathcal{J}_{\mathrm{R}}$ over π^3_{mos}-SLAM by $12.0/13.5$ points on DAVIS-16 and $7.6/8.5$ points on DAVIS-17. The strong $\mathcal{J}_{\mathrm{R}}$ results indicate temporally reliable dynamic-region recovery, which not only prevents moving objects from entering BA, but also provides cleaner support for TAGs modeling.

Table 2: **Tracking Performance on the Bonn RGB-D Dataset** (ATE RMSE ↓ [cm]).

Method	Balloon	Balloon2	Crowd	Crowd2	Person	Person2	Moving	Moving2	**Avg.**
RGB-D									
ReFusion Palazzolo et al. (2019)	17.5	25.4	20.4	15.5	28.9	46.3	7.1	17.9	22.38
DynaSLAM (N+G) Bescos et al. (2018)	3.0	2.9	1.6	3.1	6.1	7.8	23.2	3.9	6.45
4DGS-SLAM Li et al. (2025b)	2.7	4.6	141.3	64.8	9.7	11.4	2.8	5.0	30.28
Flow4DGS-SLAM Wang & Lee (2026)	2.9	3.5	2.2	4.1	7.0	11.1	4.4	4.3	4.93
RGB-only									
TTT3R Chen et al. (2026)	21.5	15.4	9.8	7.7	30.0	21.4	33.4	41.2	22.55
MonST3R Zhang et al. (2024)	7.2	6.0	6.6	6.9	9.8	16.1	3.5	6.7	7.85
DynaMoN (MS&SS) Schischka et al. (2025)	2.8	2.7	3.5	2.8	14.8	**2.2**	**1.3**	2.7	4.10
DynaMoN (MS) Schischka et al. (2025)	6.8	3.8	6.1	5.6	**2.4**	3.5	1.4	2.6	4.02
WildGS-SLAM Zheng et al. (2025a)	2.8	**2.4**	1.6	2.2	3.9	3.1	1.7	2.5	2.52
DROID-W Li et al. (2026)	2.6	2.5	1.3	**1.8**	3.3	2.9	1.6	2.3	2.30
π^3_{mos}-SLAM† Zhong et al. (2026)	**2.5**	**2.4**	1.6	2.8	3.5	2.9	**1.3**	**2.0**	2.37
Ours	2.7	2.5	**1.1**	2.6	2.7	2.6	1.4	**2.0**	**2.22**

†The π^3_{mos}-SLAM entries are full-sequence ATE from our reproduction of the official code. They differ from the values reported in the original paper because the released evaluation includes only a subset of frames.

Table 3: **Tracking Performance on Wild-SLAM MoCap Dataset** (ATE RMSE ↓ [cm]).

Method	Person	Racket	Table1	Table2	Umbrella	**Avg.**
RGB-D						
DG-SLAM Xu et al. (2024b)	1.5	1.6	2.0	57.9	1.35	12.87
UP-SLAM Zheng et al. (2025b)	1.1	0.9	0.7	3.6	0.8	1.42
4DGS-SLAM Li et al. (2025b)	6.3	2.7	1.3	33.7	1.0	8.98
Flow4DGS-SLAM Wang & Lee (2026)	0.9	1.2	1.2	3.2	1.2	1.54
RGB-only						
MonST3R-SW Zhang et al. (2024)	7.2	13.2	4.8	33.7	5.5	12.88
MegaSaM Li et al. (2024)	3.2	1.6	1.0	9.4	0.6	3.16
WildGS-SLAM Zheng et al. (2025a)	0.8	**0.4**	0.6	1.3	**0.2**	0.66
DROID-W Li et al. (2026)	0.7	0.5	0.7	2.7	**0.2**	0.96
π^3_{mos}-SLAM† Zhong et al. (2026)	**0.2**	**0.4**	**0.4**	2.1	**0.2**	0.66
Ours	0.6	**0.4**	0.7	**0.9**	**0.2**	**0.56**

Table 4: **Novel View Synthesis Evaluation on Wild-SLAM MoCap Dataset.**

Method	Metric	Person	Racket	Table1	Table2	Umbrella	**Avg.**
Splat-SLAM	PSNR [dB]↑	18.58	16.45	17.54	11.45	16.65	16.13
	SSIM↑	0.754	0.699	0.717	0.458	0.667	0.659
	LPIPS↓	0.298	0.301	0.312	0.650	0.362	0.385
WildGS-SLAM	PSNR [dB]↑	20.31	20.87	20.33	19.16	20.03	20.14
	SSIM↑	0.801	0.785	0.788	0.728	0.766	0.774
	LPIPS↓	0.189	0.186	0.209	0.303	0.210	0.219
4DGS-SLAM	PSNR [dB]↑	19.22	19.20	17.51	14.13	19.65	17.94
	SSIM↑	0.806	0.785	0.740	0.677	0.769	0.755
	LPIPS↓	0.268	0.284	0.340	0.576	0.307	0.355
Flow4DGS-SLAM	PSNR [dB]↑	20.58	20.68	20.49	19.29	20.50	20.31
	SSIM↑	**0.809**	**0.790**	**0.802**	0.742	**0.784**	**0.785**
	LPIPS↓	0.210	0.198	0.210	0.251	0.222	0.218
Ours	PSNR [dB]↑	**20.74**	**20.94**	**20.57**	**19.88**	**20.69**	**20.56**
	SSIM↑	0.804	**0.790**	0.790	**0.794**	0.749	**0.785**
	LPIPS↓	**0.167**	**0.157**	**0.161**	**0.198**	**0.173**	**0.171**

Camera Tracking. As shown in Tables 2 and 3, our method achieves the lowest average ATE of 2.22 cm on Bonn and 0.56 cm on Wild-SLAM. It surpasses recent monocular dynamic SLAM systems, including DROID-W, WildGS-SLAM, and π^3_{mos}-SLAM. This validates the effectiveness of our accurate dynamic perception and CAPS for camera tracking under substantial scene dynamics.

Table 5: **Quantitative rendering results on the Bonn RGB-D Dataset.**

Method	Metric	Balloon	Balloon2	Crowd	Crowd2	Person	Person2	Moving	Moving2	**Avg.**
4DGS-SLAM† Li et al. (2025b)	PSNR [dB]↑	23.81	22.40	16.90	16.85	19.46	17.90	23.93	21.74	20.37
	SSIM↑	0.867	0.843	0.726	0.722	0.827	0.794	0.868	0.866	0.814
	LPIPS↓	0.239	0.268	0.452	0.491	0.305	0.330	0.235	0.252	0.321
Flow4DGS-SLAM† Wang & Lee (2026)	PSNR [dB]↑	26.63	25.90	25.93	25.40	25.85	26.16	26.40	26.63	26.11
	SSIM↑	0.896	0.878	0.876	0.856	0.879	0.882	0.892	0.895	0.881
	LPIPS↓	0.196	0.204	0.209	0.243	0.221	0.206	0.187	0.205	0.209
Ours	PSNR [dB]↑	**28.70**	**27.18**	**26.62**	**26.39**	**27.86**	**26.98**	**27.58**	**26.98**	**27.28**
	SSIM↑	**0.915**	**0.900**	**0.900**	**0.886**	**0.907**	**0.903**	**0.904**	**0.904**	**0.902**
	LPIPS↓	**0.144**	**0.157**	**0.146**	**0.172**	**0.162**	**0.162**	**0.145**	**0.162**	**0.156**

†The 4DGS-SLAM and Flow4DGS-SLAM entries in this table are full-image metrics. They differ from the values reported in the original papers, which are computed only on regions retained by a valid-depth mask.

Photometric Rendering. Table 5 shows that our method outperforms RGB-D baselines on every sequence and all metrics on dynamic scene reconstruction. The superior performance demonstrates the effectiveness of the decoupled design of TAGs and its robustness under occlusion and changing visibility. Figure 3 provides a qualitative comparison on the Bonn RGB-D Dynamic dataset. Compared with 4DGS-SLAM and Flow4DGS-SLAM, our method better preserves the contours and appearance of moving objects, with less blurring and fewer rendering artifacts, while retaining static background details. Table 4 evaluates novel view synthesis on the Wild-SLAM MoCap dataset. Our method achieves the best rendering quality on most scenes and metrics, which validates our method's static mapping quality.



Figure 3: Qualitative rendering results on the Bonn RGB-D Dynamic dataset. Compared with 4DGS-SLAM and Flow4DGS-SLAM, our method produces sharper dynamic objects with fewer rendering artifacts while preserving static background details.

Table 6: Runtime (ms).

	4DGS SLAM	Flow 4DGS	Ours
Feedforward	–	–	416
Dynamic	14	119	60
Tracking	434	850	139
Mapping	109473	9720	1878
FPS	0.04	0.36	**0.94**

Runtime Analysis. Table 6 reports the runtime of different system components. Our method achieves 0.94 FPS, compared with 0.36 FPS for Flow4DGS-SLAM and 0.04 FPS for 4DGS-SLAM. The main efficiency gain comes from mapping, which demonstrates that the feed-forward initialization and track-anchored representation enable substantially more efficient dynamic mapping.

4.3 Ablation Studies

Table 7: Ablation study on the Bonn RGB-D Dynamic dataset.

Method	ATE [cm]↓	PSNR [dB]↑	SSIM↑	LPIPS↓
(a) w/o FFGS	2.22	26.52	0.895	0.179
(b) w/o CAPS	2.59	27.23	0.902	0.158
(c) w/o TAGs	2.22	20.47	0.837	0.243
(d) w/o CAU	2.22	22.24	0.854	0.218
(e) w/o 4DSH	2.22	26.99	0.898	0.159
Full Model (Ours)	**2.22**	**27.28**	**0.902**	**0.156**

Table 7 ablates major components of our method. In (a), replacing the feed-forward Gaussians with predefined counterparts lowers rendering quality, showing that the prior primitives better initialize the map. In (b), removing CAPS and sampling patches uniformly at random drops the tracking accuracy, demonstrating that leveraging the attention selectivity to patchify enables more informative constraints for BA. In (c) and (d), discarding TAGs and reverting to a static Gaussians map, or disabling their coverage-aware updates, reduces rendering metrics significantly, demonstrating the necessity of joint static-dynamic mapping and continuous maintenance of dynamic content. In (e), removing temporal appearance modeling also lowers rendering quality.

5 Conclusion

We presented TAG4D-SLAM, a monocular dynamic Gaussian SLAM framework that leverages feed-forward 3D priors for accurate camera tracking and persistent 4D scene reconstruction. For tracking, CAPS turns the latent cross-view geometry encoded in internal attention into reliable sampling guidance for bundle adjustment. For mapping, voxelized prior initialization and coverage-aware updates efficiently consolidate static content, while TAGs decouple persistent dynamic content from temporal motion by anchoring canonical Gaussians to long-lived 3D motion tracks. Together with temporal appearance modeling, this design establishes a persistent motion scaffold that enables consistent dynamic reconstruction under challenging motion and visibility changes. Extensive experiments demonstrate state-of-the-art camera tracking and 4D reconstruction.

AI USE STATEMENT

We used OpenAI GPT-5.6 to assist with language editing, polishing, and drafting portions of the manuscript based on technical content, experimental results, and ideas provided by the authors. The model was not used to conduct experiments, generate datasets, or establish mathematical claims. All AI-assisted text was critically reviewed, revised, and verified by the authors, who take full responsibility for the technical accuracy, citations, claims, and final content of the paper.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide detailed descriptions of our experimental setup, implementation details, and hyperparameter configurations in Appendix. We will make the source code publicly available upon publication.

REFERENCES

Görkay Aydemir, Fatma Güney, and Weidi Xie. Real-world point tracking with verifier-guided pseudo-labeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Berta Bescos, José M. Fácil, Javier Civera, and José Neira. DynaSLAM: Tracking, Mapping and Inpainting in Dynamic Scenes. *IEEE Robotics and Automation Letters (RA-L)*, 2018.

Berta Bescos, Carlos Campos, Juan D Tardós, and José Neira. Dynaslam ii: Tightly-coupled multi-object tracking and slam. *IEEE robotics and automation letters*, 6(3):5191–5198, 2021.

Jelena Bratulić, Sudhanshu Mittal, Thomas Brox, and Christian Rupprecht. On geometric understanding and learned priors in feed-forward 3d reconstruction models. *arXiv preprint arXiv:2512.11508*, 2025.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Easi3r: Estimating disentangled motion from dust3r without training. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9158–9168. IEEE, 2025.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Ttt3r: 3d reconstruction as test-time training. In *International Conference on Learning Representations*, volume 2026, pp. 50694–50718, 2026.

Lily Goli, Sara Sabour, Mark Matthews, Marcus A Brubaker, Dmitry Lagun, Alec Jacobson, David J Fleet, Saurabh Saxena, and Andrea Tagliasacchi. Romo: Robust motion segmentation improves structure from motion. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6155–6164. IEEE, 2025.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3749–3761, 2022.

Haochen Jiang, Yueming Xu, Kejie Li, Jianfeng Feng, and Li Zhang. Rodyn-slam: Robust dynamic dense rgb-d slam with neural radiance fields. *IEEE Robotics and Automation Letters*, 9(9):7509–7516, 2024.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. AnySplat: Feed-forward 3D gaussian splatting from unconstrained views. *ACM Transactions on Graphics (TOG)*, 44(6):1–16, 2025.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13229–13239. IEEE, 2023.

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 *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

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

Yongsung Kim, Wooseok Song, Jaihyun Lew, Hun Hwangbo, Jaehoon Lee, and Sungroh Yoon. Hess: Head sensitivity score for sparsity redistribution in vggt. *arXiv preprint arXiv:2603.25336*, 2026.

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.

Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE international conference on computer vision*, pp. 2192–2199, 2013.

Haoang Li, Xiangqi Meng, Xingxing Zuo, Zhe Liu, Hesheng Wang, and Daniel Cremers. Pg-slam: Photo-realistic and geometry-aware rgb-d slam in dynamic environments. *IEEE Transactions on Robotics*, 41:6084–6101, 2025a.

Moyang Li, Zihan Zhu, Marc Pollefeys, and Daniel Barath. Droid-slam in the wild. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 36498–36508, 2026.

Yanyan Li, Youxu Fang, Zunjie Zhu, Kunyi Li, Yong Ding, and Federico Tombari. 4d gaussian splatting slam. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25019–25028. IEEE, 2025b.

Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. *arXiv preprint arXiv:2412.04463*, 2024.

Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20981–20990. IEEE, 2022.

Dominic Maggio, Hyungtae Lim, and Luca Carlone. Vggt-slam: Dense rgb slam optimized on the sl (4) manifold. *Advances in Neural Information Processing Systems*, 38:129839–129867, 2026.

Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18039–18048. IEEE, 2024.

Riku Murai, Eric Dexheimer, and Andrew J Davison. Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16695–16705. IEEE, 2025.

Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE transactions on pattern analysis and machine intelligence*, 36(6):1187–1200, 2013.

E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *Proc. IEEE International Conf. on Intelligent Robots and Systems (IROS)*, 2019.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Dehao Qin, Ripon Kumar Saha, Woojeh Chung, Suren Jayasuriya, Jinwei Ye, and Nianyi Li. Unsupervised moving object segmentation with atmospheric turbulence. In *European Conference on Computer Vision*, pp. 18–37. Springer, 2024.

Martin Rünz and Lourdes Agapito. Co-fusion: Real-time segmentation, tracking and fusion of multiple objects. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4471–4478. IEEE, 2017.

Martin Rünz, Maud Buffier, and Lourdes Agapito. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *2018 IEEE international symposium on mixed and augmented reality (ISMAR)*, pp. 10–20. IEEE, 2018.

Erik Sandström, Ganlin Zhang, Keisuke Tateno, Michael Oechsle, Michael Niemeyer, Youmin Zhang, Manthan Patel, Luc Van Gool, Martin Oswald, and Federico Tombari. Splat-slam: Globally optimized rgb-only slam with 3d gaussians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1686–1697, 2025.

Nicolas Schischka, Hannah Schieber, Mert Asim Karaoglu, Melih Görgülü, Florian Grötzner, Alexander Ladikos, Daniel Roth, Nassir Navab, and Benjamin Busam. Dynamon: Motion-aware fast and robust camera localization for dynamic neural radiance fields. *IEEE Robotics and Automation Letters*, 10(1):548–555, 2025. doi: 10.1109/LRA.2024.3511399.

Zhengcheng Shen, Yi Gao, Linh Kästner, and Jens Lambrecht. Habitatdyn dataset: Dynamic object detection to kinematics estimation. *arXiv preprint arXiv:2304.10854*, 2023.

Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. *Advances in neural information processing systems*, 36:39033–39051, 2023.

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.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10510–10522. IEEE, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Y. Wang and G. H. Lee. Flow4dgs-slam: Optical flow-guided 4d gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations*, volume 2026, pp. 10481–10497, 2026.

Long Wen, Shixin Li, Yu Zhang, Yuhong Huang, Jianjie Lin, Fengjunjie Pan, Zhenshan Bing, and Alois Knoll. Gassidy: Gaussian splatting slam in dynamic environments. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8471–8477, 2025. doi: 10.1109/ICRA55743.2025.11127678.

Wenhua Wu, Chenpeng Su, Siting Zhu, Tianchen Deng, Jianhao Jiao, Guangming Wang, Dimitrios Kanoulas, Zhe Liu, and Hesheng Wang. Cad-slam: Consistency-aware dynamic slam with dynamic-static decoupled mapping. *arXiv preprint arXiv:2505.19420*, 2025a.

Wenhua Wu, Guangming Wang, Ting Deng, Sebastian Ægidiu, Stuart Shanks, Valerio Modugno, Dimitrios Kanoulas, and Hesheng Wang. Dvn-slam: Dynamic visual neural slam based on local-global encoding. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14564–14571. IEEE, 2025b.

Kai Xu, Tze Ho Elden Tse, Jizong Peng, and Angela Yao. Das3r: Dynamics-aware gaussian splatting for static scene reconstruction. *arXiv preprint arXiv:2412.19584*, 2024a.

Yueming Xu, Haochen Jiang, Zhongyang Xiao, Jianfeng Feng, and Li Zhang. Dg-slam: Robust dynamic gaussian splatting slam with hybrid pose optimization. *Advances in Neural Information Processing Systems*, 37:51577–51596, 2024b.

Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19595–19604. IEEE, 2024a.

Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *European Conference on Computer Vision*, pp. 156–173. Springer, 2024b.

Jinhao You, Shuo Lyu, Zhuohang Lyu, Tanxuan Li, Zibo Zhao, Jiaxiang Hu, Kai Tang, and Yichen Guo. Regimevggt: Layer-wise spatially preserving redundancy removal for visual geometry grounded transformer. *arXiv preprint arXiv:2606.18439*, 2026.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arxiv:2410.03825*, 2024.

Zicheng Zhang, Ke Wu, Xiangting Meng, Keyu Liu, Jieru Zhao, and Wenchao Ding. Flash-mono: Feed-forward accelerated gaussian splatting monocular slam. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jianhao Zheng, Zihan Zhu, Valentin Bieri, Marc Pollefeys, Songyou Peng, and Iro Armeni. Wildgs-slam: Monocular gaussian splatting slam in dynamic environments. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11461–11471. IEEE, 2025a.

Wancai Zheng, Linlin Ou, Jiajie He, Libo Zhou, Xinyi Yu, and Yan Wei. Up-slam: Adaptively structured gaussian slam with uncertainty prediction in dynamic environments. *arXiv preprint arXiv:2505.22335*, 2025b.

Xingguang Zhong, Liren Jin, Marija Popovic, Jens Behley, and Cyrill Stachniss. Dynamic visual slam using a general 3d prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21805–21815, 2026.

A Additional Implementation Details

A.1 Feed-forward Reconstruction Model Training

Architecture and initialization. We initialize the feed-forward reconstruction model from the pretrained WorldMirror Liu et al. (2025) weights and freeze all backbone parameters. Only the newly initialized dynamic segmentation head is trained. The head uses a DPT decoder with multi-level features from transformer blocks $\{4, 11, 17, 23\}$, a fusion width of 256, and a single-channel sigmoid output at the input image resolution. It consumes the concatenated frame and global features, with input dimension 2048.

Datasets and training. We combine eight sources of RGB sequences and dynamic masks: CasualMotion Goli et al. (2025), DOST Qin et al. (2024), DynamicStereo Karaev et al. (2023), HabitatDyn Shen et al. (2023), HOI4D Liu et al. (2022), MOVi-F Greff et al. (2022), SegTrackV2 Li et al. (2013), and FBMS Ochs et al. (2013). Table 8 summarizes the composition and scene characteristics of the eight training datasets. The dynamic head is trained for 100K steps using an equally weighted combination of BCE and Dice losses. We use AdamW with an initial learning rate of 2×10^{-4}, weight decay 0.1, and cosine decay to 2×10^{-5}.

Table 8: Summary of the datasets used to train the dynamic segmentation head. The image count denotes RGB-mask pairs used for training. "Mixed" indicates that the dataset contains both indoor and outdoor scenes.

Dataset	Indoor/outdoor	Real/synthetic	Images
CasualMotion	Mixed	Real	369
DOST	Outdoor	Real	1,719
DynamicStereo	Indoor	Synthetic	144,900
HabitatDyn	Indoor	Synthetic	26,028
HOI4D	Indoor	Real	76,650
MOVi-F	Mixed	Synthetic	137,688
SegTrackV2	Mixed	Real	785
FBMS	Mixed	Real	5,876

A.2 More Implementation Details

The causal window contains $w = 5$ frames, which are selected with a stride of 4 in the DPVO-maintained keyframe buffer. A pixel is dynamic when its predicted probability exceeds $\varepsilon = 0.55$. Of the N_p patches, a fraction $\rho = 0.4$ is sampled with probability proportional to $\mathbf{A}_t$. Static mapping uses $\lambda_{\text{rgb}} = 0.9$ and $\lambda_{\text{size}} = 0.01$. Within $\mathcal{L}_{\text{rgb}}$, the SSIM term has weight 0.2 and the masked ℓ_1 term has weight 0.8. The scale factor s_t is fit by Huber IRLS on historical patches in the current window. It is initialized at the confidence-weighted median ratio of the feed-forward and DPVO inverse depths, with the Huber threshold at the 70th percentile of the absolute residual. The solver runs for at most five iterations and stops when the relative change falls below 10^{-4}. Each new Gaussian binds only to the nearest valid track, and is left unbound when that track is farther than 0.02. We sample 2D tracking queries on image-space grid with a spacing of 10 pixels. When a dynamic region falls outside the existing tracks, new tracking queries are added every two frames. Online mapping is performed every five frames and runs 80 iterations per keyframe, while WildGS-SLAM Zheng et al. (2025a) uses 5.6 times as many iterations. 4DGS-SLAM Li et al. (2025b) and Flow4DGS-SLAM Wang & Lee (2026) run 200 iterations per keyframe and accumulate the loss over 10 frames at each iteration, so their number of loss evaluations is 25 times ours. After online training, we implement color refinement following these two works.

A.3 More Results and Analysis

Dynamic Segmentation. Fig. 4 compares the dynamic probability estimated by our dynamic head with that of π^3_{mos}-SLAM. Our dynamic head separates dynamic objects with sharp boundaries, whereas π^3_{mos}-SLAM leaves a hazy transition band around dynamic objects and confuses static foreground with dynamic objects.

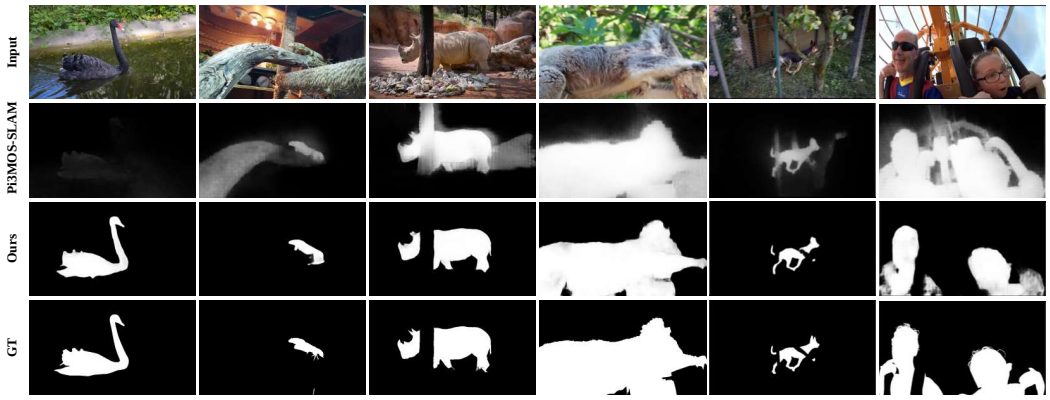

Figure 4: Qualitative Dynamic Segmentation Results.

Visualization and Analysis of CAPS. The left side of Fig. 5 shows the global attention map of the feed-forward reconstruction model. We average heads 1, 4, and 10 at layer 14 on a five-frame input, and the map forms a 5×5 array of frame-to-frame blocks. Off-diagonal blocks collect cross-frame attention, which is of interest. Each off-diagonal block repeats the same diagonal ridge, which implies that a query patch tends to attend to the surrounding co-located patches in other frames for correspondence. Along the top of every block, dark red and dark blue form a sharp contrast band, which indicates that attention there changes significantly across regions. This suggests that the attention is spatially selective, with different query tokens concentrating their attention on distinct regions, consistent with observations in Bratulić et al. (2025); You et al. (2026); Kim et al. (2026).

The middle of Fig. 5 shows the selectivity map of each input frame at the selected layers and heads.

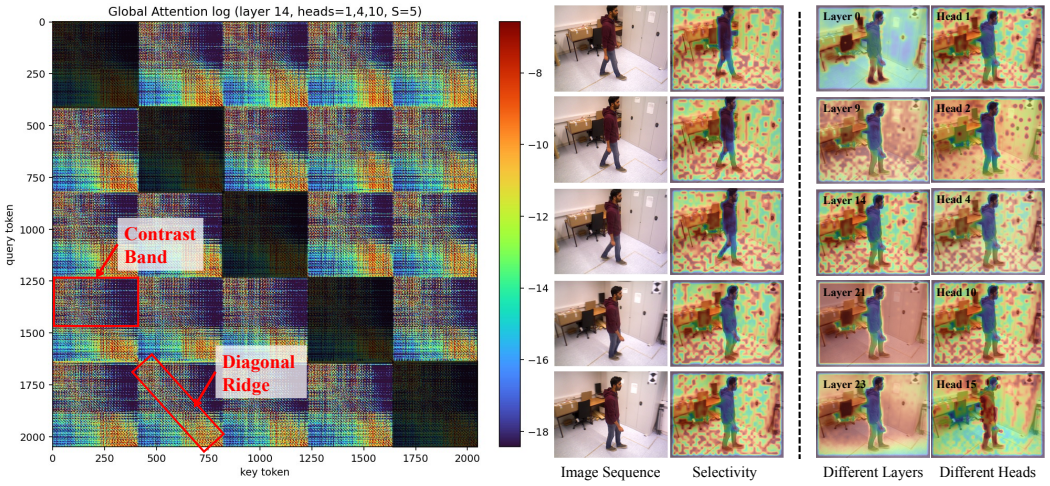

Figure 5: Global attention and selectivity of the feed-forward reconstruction model.

Table 9: Ablation of the selectivity used by CAPS on the Bonn RGB-D Dynamic dataset.

	Ours	All layers	All heads	Gradient only	SIFT match
ATE [cm] ↓	**2.22**	2.29	2.42	10.76	7.18

Regions with strong color variations receive high selectivity like the yellow texture on the floor. Regions with clear cross-frame correspondence also receive high selectivity, including the handle on the cabinet and the white tape on the box. Importantly, these observations suggest that selectivity reflects cross-view geometric evidence rather than local visual saliency alone. Our CAPS exposes this

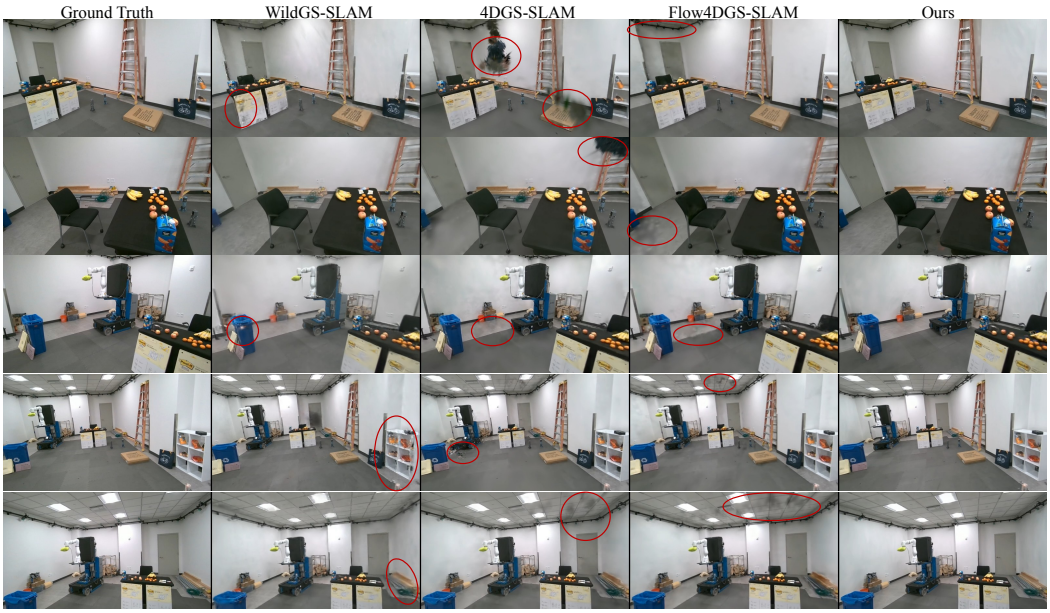

Figure 6: Qualitative novel view synthesis results on the Wild-SLAM MoCap dataset. Red ellipses highlight rendering defects of the baselines.

otherwise implicit prior and converts it into a sampling prior for bundle adjustment, thereby directing camera tracking toward measurements that provide more informative geometric constraints.

The right side of Fig. 5 visualizes the selectivity maps produced by different attention layers and heads. We observe that selectivity behavior is not uniformly distributed across the hierarchy, but emerges more prominently in a subset of layers and heads. We compare the full CAPS with two variants to quantify this observation: averaging selectivity over all layers and averaging it over all heads. As shown in Table 9, the chosen configuration outperforms the other two variants.

To better understand what makes the resulting selectivity effective for tracking, we further compare CAPS with alternative patch-sampling strategies considering two classical visual cues. *Gradient only* samples patches with the highest image gradient magnitude, favoring locally distinctive regions. *SIFT match* uses accumulated cross-frame matching confidence as the sampling score. For each frame to be sampled, sparse SIFT keypoints and descriptors are extracted over the whole image and matched to every other frame in the same temporal window. A match is kept only when the nearest neighbor is clearly closer than the second nearest, that is, Lowe's ratio $d_1/d_2 < 0.75$. Each accepted match adds confidence $1-d_1/d_2$ at that keypoint, and matches to multiple frames add up. This gives a sparse score map substituted for the CAPS selectivity map.

As shown in Table 9, attention-derived selectivity substantially outperforms both handcrafted alternatives. Sampling high-gradient regions alone leads to an ATE of 10.76 cm, indicating that local visual distinctiveness does not necessarily translate into useful geometric constraints for camera tracking. Explicit SIFT-based cross-view matching improves upon gradient-based sampling, but still yields an ATE of 7.18 cm, substantially higher than the 2.22 cm achieved by CAPS. These results indicate that the effectiveness of CAPS cannot be reproduced by simply substituting local saliency or classical sparse correspondence cues. Together with the attention visualizations, they suggest that the internal attention of the feed-forward reconstruction model encodes a geometry-aware notion of patch informativeness that is better aligned with camera pose estimation.

Qualitative Novel View Synthesis Results. Fig. 6 shows qualitative novel view synthesis on static scenes from the Wild-SLAM MoCap dataset. 4DGS-SLAM Li et al. (2025b) and Flow4DGS-SLAM Wang & Lee (2026) produce ghosting in several regions and the former also leaves residual traces of dynamic objects. WildGS-SLAM Zheng et al. (2025a) also yields blur and color artifacts. Our method produces clean, high-fidelity novel views, confirming the quality of the static map.



Figure 7: Qualitative dynamic rendering results on the Bonn RGB-D Dynamic dataset.

Table 10: **Quantitative dynamic rendering results on the Bonn RGB-D Dynamic Dataset.**

Method	Metric	Balloon	Balloon2	Crowd	Crowd2	Person	Person2	Moving	Moving2	**Avg.**
4DGS-SLAM Li et al. (2025b)	PSNR [dB]↑	24.75	16.98	18.22	18.62	19.35	24.23	22.11	17.70	20.24
	SSIM↑	0.795	0.344	0.730	0.716	0.568	0.776	0.596	0.746	0.659
	LPIPS↓	0.174	0.299	0.413	0.363	0.211	0.180	0.252	0.247	0.268
Flow4DGS-SLAM Wang & Lee (2026)	PSNR [dB]↑	5.72	6.14	18.13	15.42	16.18	15.30	18.47	16.09	13.93
	SSIM↑	0.123	0.119	0.331	0.272	0.320	0.280	0.439	0.229	0.264
	LPIPS↓	0.670	0.705	0.476	0.527	0.359	0.365	0.321	0.405	0.479
Ours	PSNR [dB]↑	**27.82**	**27.39**	**25.32**	**25.72**	**28.78**	**30.50**	**22.18**	**22.34**	**26.26**
	SSIM↑	**0.945**	**0.941**	**0.885**	**0.891**	**0.957**	**0.958**	**0.903**	**0.923**	**0.925**
	LPIPS↓	**0.094**	**0.097**	**0.175**	**0.159**	**0.076**	**0.074**	**0.156**	**0.130**	**0.120**

Dynamic Rendering Results. Fig. 7 and Table 10 further evaluate dynamic reconstruction in isolation. Our method consistently produces substantially higher-fidelity dynamic renderings. 4DGS-SLAM exhibits missing or blurred dynamic content under large motion and occlusion, as its learned deformation field must infer temporal motion from sparse control points. Flow4DGS-SLAM is more strongly affected by ghosting and inaccurate object geometry, since frame-local flow propagation can accumulate correspondence and displacement errors, especially around motion boundaries and temporary occlusions. In contrast, TAGs anchor canonical Gaussians to long-lived 3D tracks, providing explicit and persistent temporal correspondence instead of repeatedly inferring motion from local observations. Together with coverage-aware insertion and track-conditioned visibility, this allows newly visible content to be incorporated while suppressing unsupported Gaussians during occlusion. Consequently, our method preserves sharper object boundaries, more complete appearance, and substantially more temporally consistent dynamic renderings. Notably, this advantage is achieved from monocular RGB input, whereas both competing dynamic Gaussian SLAM methods use RGB-D observations.

Tracking Qualitative Results. Following the evaluation protocol of compared baselines, we align each predicted trajectory to the ground truth with a $\mathrm{Sim}(3)$ transformation before computing ATE. Fig. 8 compares estimated camera trajectories with the ground truth on the xy plane. The estimate is drawn as a colored curve and the ground truth as a dashed gray curve. Color encodes the error magnitude, from blue for low error to red for high error. On `Balloon2`, where the motion is rapid, and on `Crowd`, where dynamic occlusion is severe, 4DGS-SLAM and Flow4DGS-SLAM drift substantially. π^3_{mos}-SLAM stays close to our trajectory, but it still produces occasional large deviations and is less robust.

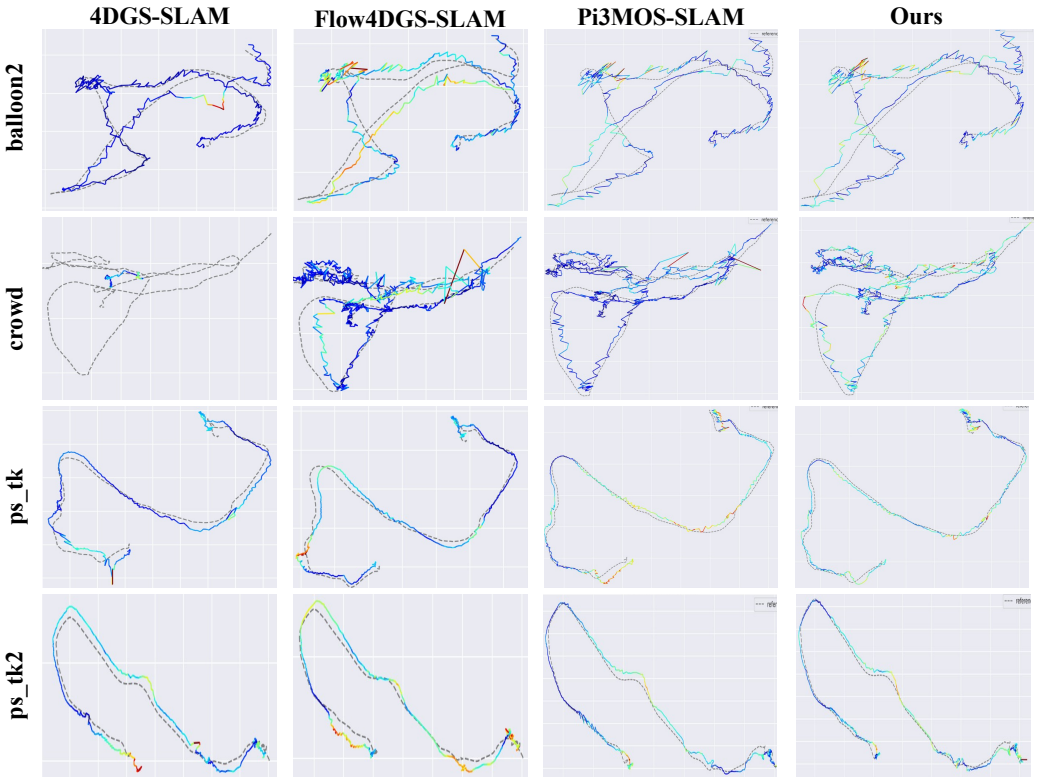

Figure 8: Qualitative comparison of camera trajectories.

A.4 Supplementary Videos

We provide supplementary videos to further demonstrate the temporal behavior of dynamic reconstruction, which cannot be fully captured by frame-based qualitative comparisons. We present side-by-side comparisons with 4DGS-SLAM and Flow4DGS-SLAM on the *Person*, *Balloon*, and *Crowd2* sequences. 4DGS-SLAM often produces incomplete or blurred dynamic content, while Flow4DGS-SLAM is more susceptible to background leakage and temporally accumulated ghosting around moving objects. These artifacts become particularly noticeable under large displacement, occlusion, and changing visibility. The *Crowd2* sequence further involves multiple simultaneously moving people and frequent mutual occlusions, providing a particularly challenging case for maintaining long-term temporal consistency. In contrast, our Track-Anchored Gaussians maintain persistent correspondence by anchoring canonical Gaussians to long-lived 3D motion tracks. This enables the reconstructed dynamic content to follow the observed motion consistently while preserving cleaner object boundaries and more complete appearance throughout the sequence. Overall, the videos demonstrate that our method improves not only per-frame rendering quality, but more importantly the temporal consistency and persistence of dynamic reconstruction.