RayMap3R: Inference-Time RayMap for Dynamic 3D Reconstruction

Feiran Wang[1], Zezhou Shang[1], Gaowen Liu[2], and Yan Yan[1,†]

[1] University of Illinois Chicago, USA
[2] Cisco Research, USA

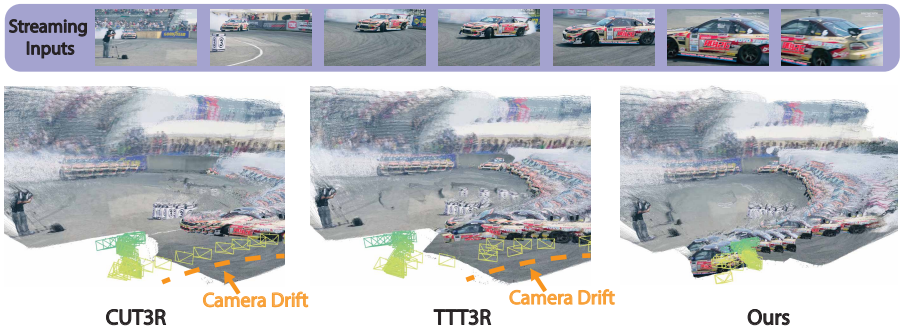

Fig. 1: Streaming 3D Reconstruction for Dynamic Scenes. RayMap3R leverages the static bias of RayMap-only predictions to identify and suppress dynamic regions at inference time, reducing camera drift and improving geometry fidelity.

Abstract. Streaming feed-forward 3D reconstruction enables real-time joint estimation of scene geometry and camera poses from RGB images. However, without explicit dynamic reasoning, streaming models can be affected by moving objects, causing artifacts and drift. In this work, we propose RayMap3R, a training-free streaming framework for dynamic scene reconstruction. We observe that RayMap-based predictions exhibit a static-scene bias, providing an internal cue for dynamic identification. Based on this observation, we construct a dual-branch inference scheme that identifies dynamic regions by contrasting RayMap and image predictions, suppressing their interference during memory updates. We further introduce reset metric alignment and state-aware smoothing to preserve metric consistency and stabilize predicted trajectories. Our method achieves state-of-the-art performance among streaming approaches on dynamic scene reconstruction across multiple benchmarks. The project page and code are available at `https://raymap3r.github.io/`.

Keywords: 3D Reconstruction · Streaming Inference · Dynamic Scenes

1 Introduction

Feed-forward 3D models have achieved remarkable progress in reconstructing 3D structures such as point clouds, depth maps, and camera poses from monocular

† Corresponding author.

images. An influential class of methods [10, 11, 33, 35, 42, 45] adopts an offline reconstruction paradigm that builds pairwise correspondences between all input images, forming dense attention maps that enable precise estimation of geometry and camera poses. However, the offline paradigm requires the complete set of frames before producing output, making it unsuitable for real-time applications. Moreover, both computational and memory costs scale rapidly with sequence length, for instance requiring up to 48 GB for sequences of only 200 frames [19].

To extend feed-forward reconstruction to real-time processing, recent works [7, 32, 34, 37] focus on streaming feed-forward 3D reconstruction, which performs continuous inference from input image streams while maintaining stable computational resources. These methods introduce memory mechanisms that interact directly with incoming images, enabling high-frequency real-time inference with constant memory usage even over thousands of frames. Concretely, approaches include spatial memory banks, implicit recurrent states, and explicit point anchors, offering different trade-offs between efficiency and reconstruction fidelity.

However, streaming feed-forward 3D reconstruction models still face major limitations. First, these models lack explicit mechanisms to identify dynamic objects, partly due to the scarcity of training data with dynamic annotations. Unlike offline methods that leverage dense pairwise matching across all frames, streaming methods process frames sequentially with limited historical context, making them particularly vulnerable to dynamic interference. Second, frame-by-frame prediction may introduce pose noise that accumulates over long sequences.

Existing approaches to dynamic scene reconstruction [5, 16, 29, 42] typically require external modules such as flow estimators, adding overhead and domain-specific dependencies. RayMap [41], a per-pixel camera-ray representation, has been widely adopted in recent feed-forward models [14, 33, 34] to strengthen the link between appearance and geometry. We observe that models trained with RayMap tend to exhibit a static-scene bias, which can be exploited to identify dynamic regions at inference time without additional training or external models.

Specifically, when queried with only RayMap features, the model prioritizes static structure while suppressing dynamic regions. This bias arises because RayMap tokens encode only camera geometry, forcing the model to reconstruct scene content solely from memory. Since static structures are observed consistently across frames, they are recalled reliably, whereas dynamic objects appear transiently and tend to be suppressed. The discrepancy between RayMap-only and image-based predictions thus provides a signal exploitable for training-free dynamic identification. We provide further analysis of this property in Sec. 3.2.

Building on this observation, we propose **RayMap3R**, an inference-time framework for dynamic scene reconstruction. We employ a dual-branch inference scheme: the main branch processes both image and RayMap features for geometry estimation, while the RayMap branch uses only pose-derived RayMap features to produce static-biased predictions. By comparing these predictions, we derive staticness weights that modulate memory state updates, suppressing interference from dynamic regions while preserving static structure. To preserve metric consistency across memory reset boundaries, we introduce reset metric

alignment, estimating corrective transformations from repeated frames to restore alignment between segments. We further propose state-aware smoothing to stabilize predicted trajectories by using the magnitude of internal state changes as an uncertainty signal to adaptively smooth pose predictions online.

We evaluate RayMap3R on benchmarks covering camera pose estimation, depth prediction, and 3D reconstruction across both synthetic and real-world datasets. As shown in Fig. 1, RayMap3R achieves state-of-the-art performance among streaming methods, with particularly strong gains on dynamic scenes under both per-sequence and metric-scale settings, while producing high-quality 3D reconstructions at real-time speeds with constant memory usage.

Our key contributions are:

- We observe that RayMap predictions exhibit a static-scene bias, and exploit this bias to derive staticness weights for dynamic-aware memory updates.
- We introduce reset metric alignment and state-aware smoothing to stabilize trajectory estimation across long sequences.
- Extensive experiments show that RayMap3R achieves leading performance among streaming methods across benchmarks with real-time efficiency.

2 Related Work

Offline Reconstruction Models. Offline reconstruction methods require the complete set of input images before producing outputs [10, 18, 33, 35, 42], employing global optimization or full-sequence attention to prioritize accuracy over real-time performance. DUSt3R [35] pioneered the pointmap representation for scene-level 3D reconstruction, inferring camera poses and aligned point clouds from image pairs. Subsequent approaches [11,18,28,42] extended this framework but require pairwise processing with time-consuming optimization. MASt3R [11] augments DUSt3R with dense local features, while Fast3R [39] achieves reconstruction in a single forward. VGGT [33] employs a DPT backbone for joint pose and geometry estimation, achieving highly accurate results, though computational cost increases rapidly with input count, limiting scalability to long sequences. VGGT-Long [10] introduces chunk-based reconstruction, but memory grows with chunk size, restricting real-time applicability.

Streaming Reconstruction Models. To enable real-time 3D reconstruction, recent works introduce memory mechanisms for continuous inference [7,32,34,37, 45]. Spann3R [32] extends DUSt3R with external spatial memory that retains relevant 3D information across frames, achieving fast reconstruction with efficient memory usage but remaining less robust in dynamic scenes. StreamVGGT [45] distills VGGT and introduces a spatial-temporal decoder for streaming prediction, but memory consumption grows with frame count. Point3R [37] maintains explicit 3D point anchors to preserve and match historical information. CUT3R [34] adopts a recurrent design with implicit memory cache and location dictionary for efficient lookup, achieving continuous real-time reconstruction. TTT3R [7] extends CUT3R with a soft gating mechanism based on cross-attention to mitigate memory forgetting, enabling stable long-range reconstruc-

tion. However, without explicit dynamic supervision during training, streaming models remain susceptible to interference from moving objects.

Structure-from-Motion and Visual SLAM. Traditional SLAM systems [1, 3, 9, 20, 23, 25] rely on feature correspondence and bundle adjustment for camera pose estimation but frequently fail in low-texture or moving object scenarios. Learning-based methods like DROID-SLAM [30] incorporate differentiable optimization frameworks, though they remain vulnerable to dynamic content. AnyCam [36] fuses depth with optical flow for accurate pose estimation; DPVO [31] leverages patch-level features; MASt3R-GA [11] integrates learned features. VGGT-SLAM [19] augments VGGT with SL(4)-based backend refinement, enhancing trajectory consistency and reconstruction quality over extended sequences. However, these methods either require iterative optimization or additional motion estimation modules, incurring substantial overhead.

Dynamic Scene Reconstruction. Existing approaches generally fall into three categories: flow-based methods [16, 42] rely on external flow estimators with hand-tuned thresholds that generalize poorly; segmentation-based methods [24, 29] isolate dynamic objects via learned models yet remain constrained by training categories; and tracking-based methods [5] require known camera intrinsics and iterative optimization. All three introduce additional modules with substantial overhead, limiting generalization. Beyond these, dynamic-aware systems [6, 15, 38, 40, 43, 44] mostly rely on external modules or offline optimization. In contrast, our work reveals that models trained with the RayMap paradigm exhibit a static-scene bias exploitable for training-free dynamic reconstruction.

3 Methods

Our method takes a stream of input images and predicts camera poses, depths, and point clouds. In Sec. 3.1, we introduce RayMap and the memory mechanism for streaming reconstruction. In Sec. 3.2, we show that RayMap-based predictions exhibit a static-scene bias useful for dynamic identification. In Sec. 3.3, we propose a dual-branch inference scheme that leverages this bias to identify dynamic regions by contrasting image-based and RayMap-only predictions. Finally, in Secs. 3.4 and 3.5, we introduce reset metric alignment and state-aware smoothing to stabilize trajectory estimation over long sequences.

3.1 RayMap and Memory Mechanism

RayMap Representation. RayMap is an $H \times W \times 6$ per-pixel tensor encoding the *ray origin* and *unit direction* for each pixel, determined by camera intrinsics $K \in \mathbb{R}^{3\times3}$ and extrinsics $\mathbf{T} = [R \,|\, \boldsymbol{\tau}]$ [12, 34, 41]. For a pixel in homogeneous coordinates $\tilde{\mathbf{u}} = (x, y, 1)^\top$, the RayMap is defined as

$$\mathrm{RayMap}(x, y) = \big[\, \mathbf{c},\, \widehat{\mathbf{d}}(x, y) \,\big] \in \mathbb{R}^6, \tag{1}$$

where the ray origin and unit direction in world coordinates:

$$\mathbf{c} = -R^\top \boldsymbol{\tau}, \qquad \widehat{\mathbf{d}}(x, y) = \frac{R^\top K^{-1} \tilde{\mathbf{u}}}{\|R^\top K^{-1} \tilde{\mathbf{u}}\|_2}. \tag{2}$$

Under the pinhole model, $\mathbf{c}$ remains constant across all pixels as it represents the camera center, while $\widehat{\mathbf{d}}(x, y)$ encodes the viewing direction of each ray.

Memory Mechanism. Streaming 3D reconstruction models maintain a memory cache for continuous interaction with images. Methods such as CUT3R [34] and TTT3R [7] employ an implicit memory cache, enabling stable memory usage and fast inference over thousands of frames. The implicit memory is represented as a latent state s_t, encoding the understanding of the 3D scene at timestep t.

Training and Inference. During training, input image I_t is patchified into image tokens f_t, while the corresponding camera pose is transformed into RayMap and patchified into RayMap tokens r_t. These tokens interact with the state through two strategies: either image tokens f_t combined with RayMap tokens r_t, or RayMap tokens r_t alone. The state updates from s_{t-1} to s_t by integrating information from f_t and r_t, then the query process decodes s_t to predict camera pose, depth, and point clouds. During inference, after a short warmup period, the model can predict 3D information from either images or RayMap alone, enabling geometry prediction from arbitrary viewpoints without image input.

3.2 Static Bias of RayMap Predictions

We observe that streaming 3D reconstruction models trained with the RayMap paradigm [7,34] exhibit a static-scene bias when making predictions from RayMap tokens alone. As described in Sec. 3.1, the model can predict geometry using only RayMap tokens r_t and the scene representation in s_{t-1}. Without appearance information from image features f_t, the model relies solely on geometric consistency in r_t, and tends to prioritize static structure over dynamic content.

As illustrated in Fig. 2 (left), given the same memory state s_{t-1}, the image-based main prediction reconstructs the scene including the dynamic foreground, while the RayMap-based prediction from the same camera pose produces geometry focused on static background, suppressing the influence of moving objects. We attribute this bias to two factors. First, training datasets are dominated by static scenes with limited dynamic annotations, causing models to develop a prior toward static structures. Second, RayMap tokens encode only camera geometry without appearance cues, leading the model to rely on temporally consistent structure stored in memory rather than frame-specific dynamic content.

To examine whether this property holds generally, we compute the per-pixel depth discrepancy between the two branches across 108 sequences from MPI Sintel [2], DAVIS 2017 [22], and TUM RGB-D [27], and measure its overlap with ground-truth dynamic masks via IoU. As shown in Fig. 2 (right), the depth discrepancy correlates positively with the ground-truth dynamic ratio (Spearman $\rho = 0.77$, $p < 10^{-22}$), indicating that the depth discrepancy between branches reflects the presence of dynamic content across diverse scenes. Qualitative examples in Fig. 3 further show that the resulting dynamic map closely aligns with ground-truth masks on both synthetic and real scenes. We leverage this property to identify dynamic regions without additional supervision.

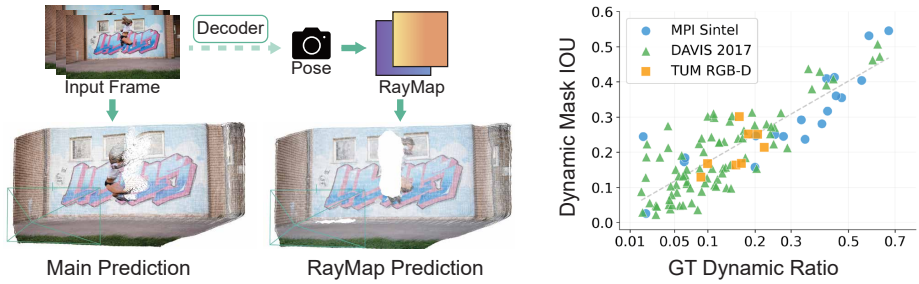

Fig. 2: RayMap Static Bias. Left: The main prediction and RayMap prediction produce differing depth estimates in dynamic regions, suggesting a static bias in RayMap predictions. **Right:** The depth discrepancy correlates with ground-truth dynamic ratio across multiple datasets, suggesting a consistent signal for dynamic identification.

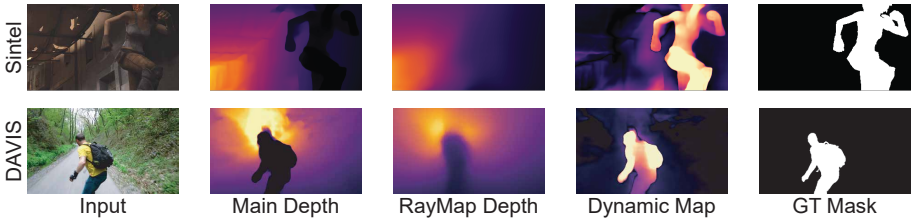

Fig. 3: Dynamic Map Visualization. The dynamic map is the per-pixel depth difference between the main branch and RayMap-only predictions, closely aligning with ground-truth dynamic masks across both synthetic and real scenes.

3.3 Dynamic Identification via RayMap Remap

Building on the static bias observed in Sec. 3.2, we propose a dual-branch inference scheme to identify dynamic regions (Fig. 4). At each timestep, both branches decode from the same frozen state s_{t-1}: the main branch processes image and RayMap features, while the RayMap branch processes only RayMap features constructed from the main branch's predicted pose. The per-pixel depth discrepancy between branches then serves as a signal for dynamic identification.

Specifically, given the main branch's predicted pose $\hat{\mathbf{T}}_t$, we construct a new RayMap $\mathcal{R}(\hat{\mathbf{T}}_t)$ and feed it through the encoder to obtain RayMap tokens r'_t. The decodings for the main and RayMap branches are given respectively by:

$$\hat{\mathbf{y}}_t^{\text{main}} = \text{Dec}(f_t + r_t, s_{t-1}), \quad \hat{\mathbf{y}}_t^{\text{raymap}} = \text{Dec}(r'_t, s_{t-1}), \tag{3}$$

where f_t and r_t are the original image and RayMap tokens. Each prediction $\hat{\mathbf{y}}_t$ contains a depth map z_t and a confidence map.

We identify dynamic regions by comparing the depth and confidence maps from both branches. For each pixel i, we compute the absolute relative depth difference $\delta_i = |z_i^{\text{main}} - z_i^{\text{raymap}}|/|z_i^{\text{main}}|$, where higher values indicate greater likelihood of dynamic content. Since the staticness weights must operate on state tokens rather than pixels, we aggregate δ_i in two steps. Pixel-level scores are

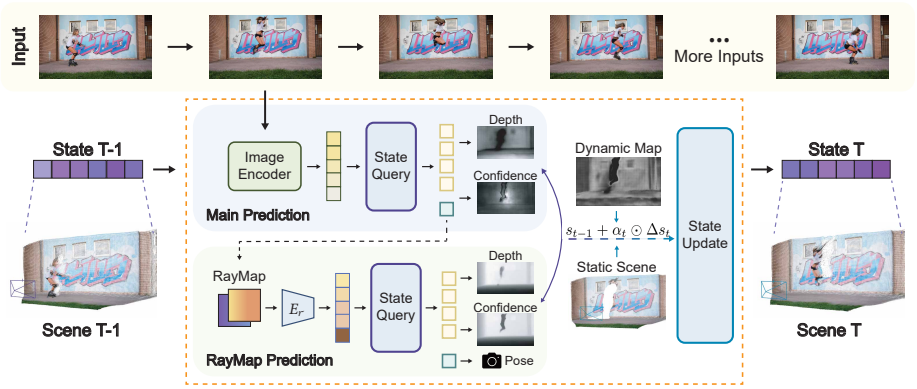

Fig. 4: Method Overview. Our method performs streaming 3D reconstruction via dual-branch inference. At time step t, the **main branch** predicts depth and pose from state s_{t-1} using both image and RayMap features. The predicted pose $\hat{\mathbf{T}}_t$ is remapped into RayMap and encoded into tokens r'_t, then queried against the *frozen* state s_{t-1} by the **RayMap branch** to obtain a static-biased prediction. The depth difference between branches is projected via image-state attention to form staticness weights α_t that suppress dynamic regions during state update ($s_t = s_{t-1} + \alpha_t \odot \Delta s_t$). This dual-branch scheme identifies and suppresses dynamic regions at inference time.

first pooled into image-token-level scores δ_k^{tok} via confidence-weighted averaging within each patch, where the weights are the confidence scores predicted by the main branch. These are then projected onto per-state-token scores δ_j^{state} as a weighted average over δ_k^{tok} using the decoder's cross-attention weights A_{jk} between state token j and image token k. We convert δ^{state} to staticness weights $\alpha_t \in \mathbb{R}^N$, where N is the number of state tokens:

$$\alpha_t = \sigma\left(\gamma \cdot \left(\text{median}(\delta^{\text{state}}) - \delta^{\text{state}}\right) / \text{IQR}(\delta^{\text{state}})\right), \tag{4}$$

where σ is the sigmoid function, γ controls the sensitivity of dynamic-static separation, and IQR is the interquartile range as a robust scale estimate. Tokens with high α_t receive full state updates while those with low α_t are suppressed, effectively filtering dynamic regions from memory.

Finally, we apply these weights to modulate state updates. We accumulate α_t over time via exponential moving average to improve temporal stability, yielding the final weights that gate the state update: $s_t = s_{t-1} + \alpha_t \odot \Delta s_t$, where Δs_t is the state update produced by the decoder and $\odot$ denotes element-wise multiplication. Additionally, we use the pixel-level staticness map to construct a weighted global feature that biases pose retrieval toward static regions during memory updates. After a warmup using only the main branch, this dual-branch scheme operates at each timestep without backpropagation, enabling dynamic-aware streaming reconstruction with constant memory usage.

3.4 Reset Metric Alignment

Streaming reconstruction models maintain a persistent memory to accumulate scene understanding over time. However, the memory mechanism suffers from forgetting over extended sequences, as new observations gradually interfere with earlier context. To mitigate this, memory-based methods typically employ periodic resets: the memory state is cleared and reinitialized using a repeated frame to maintain temporal continuity. We observe that this reset process introduces a notable problem: metric misalignment across segments.

The reset mechanism alters the memory state, causing the model to produce inconsistent camera parameters and geometry for the same repeated frame before and after reset. This discrepancy manifests as scale mismatch between segments, which propagates as systematic drift throughout subsequent frames, accumulating error that degrades reconstruction quality over long sequences.

We address this by exploiting the property that the repeated frame should yield consistent reconstructions before and after reset. Since both observations capture the same physical scene, any discrepancy between their reconstructions directly reflects the metric misalignment introduced by the reset process, providing a measurable signal for correction.

Specifically, we estimate a Sim(3) transformation that aligns the point cloud reconstructions from the repeated frame before and after the reset, as both observations capture the same physical scene. This transformation captures both the scale mismatch and pose offset between segments. The estimated transformation is then applied to all subsequent frames in the new segment, restoring metric alignment and reducing error accumulation.

3.5 State-Aware Smoothing

Streaming methods estimate camera poses sequentially, which may lead to noisy predictions that accumulate over time. While post-hoc optimization [11, 30, 31] can refine trajectories offline, it is incompatible with online processing. To this end, we introduce state-aware smoothing, which derives a per-frame smoothing coefficient from trajectory acceleration and internal state change magnitude to adaptively stabilize pose predictions online.

To derive the per-frame confidence, we define the state change signal sc_t as the mean ℓ_2 norm of Δs_t across all N state tokens, computed prior to the gated update in Sec. 3.3, so that it reflects the full proposed change. We further define the trajectory acceleration $a_t = \|\mathbf{d}_t - \mathbf{d}_{t-1}\|_2$, where $\mathbf{d}_t = \boldsymbol{\tau}_t - \boldsymbol{\tau}_{t-1}$ is the inter-frame camera displacement. The product $a_t \times \mathrm{sc}_t$ captures motion irregularity and model uncertainty, reducing false positives from either signal alone: a high a_t with low sc_t may indicate steady fast motion rather than noise, and vice versa.

To suppress unreliable predictions while preserving stable motion estimates, we convert the product $a_t \times \mathrm{sc}_t$ into a smoothing coefficient $\beta_t \in [0, 1]$ via an inverse mapping and exponentially filter the inter-frame displacements:

$$\hat{\mathbf{d}}_t = \beta_t \, \mathbf{d}_t + (1 - \beta_t) \, \hat{\mathbf{d}}_{t-1}, \quad \beta_t = \frac{1}{1 + \lambda \, |a_t \times \mathrm{sc}_t|}, \tag{5}$$

where $\boldsymbol{\tau}_t \in \mathbb{R}^3$ is the translation component of the predicted pose $\hat{\mathbf{T}}_t$, λ controls the sensitivity of the mapping, and $\hat{\mathbf{d}}_0 = \mathbf{0}$. When both a_t and sc_t are large, β_t approaches zero and the filter relies on the accumulated estimate $\hat{\mathbf{d}}_{t-1}$, attenuating trajectory jitter; when the product is small, β_t approaches one and the raw displacement is preserved. The filtered position $\hat{\boldsymbol{\tau}}_t = \hat{\boldsymbol{\tau}}_{t-1} + \hat{\mathbf{d}}_t$ is computed recursively, producing a smoothed trajectory without explicit history storage. Expanding the recursion yields the closed-form trajectory:

$$\hat{\boldsymbol{\tau}}_t = \boldsymbol{\tau}_0 + \sum_{m=1}^{t} \sum_{k=1}^{m} \beta_k \prod_{l=k+1}^{m} (1 - \beta_l) \cdot \mathbf{d}_k, \tag{6}$$

where $\boldsymbol{\tau}_0$ is the initial translation. The product $\beta_k \prod_{l=k+1}^{m}(1 - \beta_l)$ governs the effective contribution of each past displacement $\mathbf{d}_k$, forming an adaptive decay that concentrates weight on recent frames during stable intervals and broadens the filtering horizon during high-uncertainty intervals. The inverse mapping responds to the absolute magnitude of $a_t \times \mathrm{sc}_t$, so sustained fast motion at constant velocity produces low acceleration and leaves β_t close to one, avoiding excessive smoothing. This causal scheme operates fully online with negligible overhead.

4 Experiments

We evaluate RayMap3R on three core 3D tasks: video depth estimation (Sec. 4.1), camera pose estimation (Sec. 4.2), and 3D reconstruction (Sec. 4.3).

Baselines. We compare RayMap3R against streaming 3D reconstruction methods including Spann3R [32], CUT3R [34], Point3R [37], StreamVGGT [45], and TTT3R [7]. Spann3R extends DUSt3R [35] with spatial memory mechanisms for efficient reconstruction. CUT3R maintains an implicit memory cache for continuous reconstruction. Point3R enhances CUT3R with explicit point anchors. StreamVGGT builds upon VGGT [33] through knowledge distillation to enable streaming reconstruction. TTT3R extends CUT3R by using cross-attention as soft-gate guidance for long-range reconstruction. We also evaluate against offline reconstruction methods [11,33,35,42], which achieve higher accuracy but are not viable for long sequences due to prohibitive memory growth.

Datasets. Following previous work [7,34,42], we evaluate on diverse benchmarks covering both dynamic and static scenes. For dynamic scenes, we use Sintel [2], TUM-Dynamics [27], KITTI [13], and Bonn [21]. For static scenes, we evaluate on ScanNet [8] and 7-Scenes [26]. These benchmarks collectively cover diverse conditions including dynamic and static, synthetic and real-world scenes.

4.1 Video Depth Estimation

Following previous works [7, 34, 42], we evaluate video depth estimation on Sintel [2], KITTI [13], and Bonn [21], covering dynamic and static scenes across indoor and outdoor environments. We use absolute relative error (Abs Rel) and

Table 1: Video Depth Estimation. We report scale-invariant relative depth (per-sequence scale alignment) and metric-scale absolute depth accuracy. RayMap3R achieves leading depth accuracy among streaming methods under both settings. ✗ denotes offline methods; ✓ denotes streaming methods.

			KITTI [13]		BONN [21]		Sintel [2]	
Alignment	**Method**	**Onl.**	Abs Rel ↓	δ<1.25 ↑	Abs Rel ↓	δ<1.25 ↑	Abs Rel ↓	δ<1.25 ↑
Per-sequence scale	DUSt3R [35]	✗	0.144	81.3	0.155	83.3	0.656	45.2
	MonST3R [42]	✗	0.168	74.4	0.067	96.3	0.378	55.8
	VGGT [33]	✗	**0.070**	**96.5**	**0.055**	**97.1**	**0.287**	**66.1**
	Spann3R [32]	✓	0.198	73.7	0.144	81.3	0.622	42.6
	Point3R [37]	✓	0.135	84.0	0.061	96.2	0.451	48.7
	StreamVGGT [45]	✓	0.173	72.1	0.063	97.2	0.323	65.7
	CUT3R [34]	✓	0.118	88.1	0.078	93.7	0.421	47.9
	TTT3R [7]	✓	0.114	90.4	0.068	95.4	0.409	48.8
	Ours	✓	0.098	92.8	0.057	97.4	0.401	50.9
Metric scale	Point3R [37]	✓	0.190	73.9	0.136	94.6	0.778	17.0
	CUT3R [34]	✓	0.122	85.5	0.103	88.5	1.029	23.8
	TTT3R [7]	✓	0.111	88.8	0.089	94.2	0.977	23.2
	Ours	✓	0.104	89.4	0.085	94.8	0.954	24.0

$\delta < 1.25$ as metrics. Following [7,34], we report results under two protocols: per-sequence scale alignment, which evaluates relative depth accuracy, and metric scale without alignment, which measures absolute scale consistency.

As shown in Tab. 1, RayMap3R achieves leading performance among streaming methods under both evaluation protocols. Under per-sequence scale alignment, our method leads on Bonn and KITTI, while on Sintel StreamVGGT achieves lower depth error than ours, albeit with memory that scales with sequence length, among the compared streaming baselines. The improvement on KITTI is pronounced, as explicitly querying static regions via RayMap yields more temporally consistent depth predictions in large-scale outdoor scenes. Under the metric-scale setting, RayMap3R leads on Bonn and KITTI, and achieves competitive accuracy on Sintel, where Point3R obtains lower absolute error but degrades on threshold accuracy. RayMap3R maintains stable performance across both protocols, suggesting that suppressing dynamic regions during state updates helps preserve scale consistency otherwise corrupted by dynamic regions.

4.2 Camera Pose Estimation

Following previous works [4, 34, 42], we evaluate camera pose estimation on Sintel [2] with complex dynamic content, TUM-dynamics [27] with real-world dynamic scenes, and ScanNet [8] for static indoor environments. We report Absolute Translation Error (ATE) for global trajectory accuracy and Relative Pose Error (RPE) for translation and rotation using Sim(3) alignment.

As shown in Tab. 2, RayMap3R achieves leading ATE and translational RPE across all datasets, while maintaining strong rotational RPE. On Sintel, where dynamic objects occupy large portions of the frame, our dual-branch scheme effectively identifies and suppresses their interference, yielding substantial ATE

Table 2: Camera Pose Estimation. RayMap3R achieves leading trajectory accuracy among streaming methods across all datasets, with particularly strong performance on dynamic sequences, while remaining competitive on static scenes.

		Sintel [2]			TUM-dyn [27]			ScanNet [8]		
Method	**Onl.**	ATE ↓	RPE_t ↓	RPE_r ↓	ATE ↓	RPE_t ↓	RPE_r ↓	ATE ↓	RPE_t ↓	RPE_r ↓
DUSt3R [35]	✗	0.417	0.250	5.796	0.083	0.017	3.567	0.081	0.028	0.784
MASt3R [11]	✗	0.185	0.060	1.496	0.038	0.012	0.448	0.078	0.020	0.475
MonST3R [42]	✗	**0.111**	**0.044**	0.869	0.098	0.019	0.935	0.077	0.018	0.529
VGGT [33]	✗	0.172	0.062	**0.471**	**0.012**	**0.010**	**0.310**	**0.035**	**0.015**	**0.377**
Spann3R [32]	✓	0.329	0.110	4.471	0.056	0.021	0.591	0.096	0.023	0.661
Point3R [37]	✓	0.351	0.128	1.822	0.075	0.029	0.642	0.106	0.035	1.946
StreamVGGT [45]	✓	0.251	0.149	1.894	0.061	0.033	3.209	0.161	0.057	3.647
CUT3R [34]	✓	0.208	0.072	0.636	0.031	0.009	0.303	0.098	0.022	0.600
TTT3R [7]	✓	0.210	0.091	0.722	0.019	0.008	0.292	0.065	0.021	0.637
Ours	✓	0.166	0.056	0.720	0.018	0.005	0.287	0.064	0.016	0.635

Table 3: 3D Reconstruction on 7-Scenes [26]. Each metric reports Mean, Median, and Minimum across scenes. RayMap3R achieves leading reconstruction quality among streaming methods across the majority of metrics.

		Acc↓			Comp↓			NC↑			Chamfer↓		
Method	**Onl.**	Mean	Med.	Min	Mean	Med.	Min	Mean	Med.	Min	Mean	Med.	Min
DUSt3R-GA [35]	✗	**0.146**	**0.077**	**0.052**	0.181	**0.067**	**0.042**	**0.736**	**0.839**	**0.865**	**0.327**	**0.144**	**0.094**
MASt3R-GA [11]	✗	0.185	0.081	0.056	**0.180**	0.069	0.047	0.701	0.792	0.821	0.365	0.150	0.103
MonST3R-GA [42]	✗	0.248	0.185	0.142	0.266	0.167	0.121	0.672	0.759	0.783	0.514	0.352	0.263
Spann3R [32]	✓	0.298	0.226	0.170	0.205	0.112	0.078	0.650	0.730	0.754	0.503	0.338	0.248
CUT3R [34]	✓	0.043	0.026	0.015	0.031	0.018	0.009	0.621	0.618	0.523	0.027	0.025	0.013
TTT3R [7]	✓	0.027	0.024	0.012	0.023	0.017	0.005	0.582	0.583	0.545	0.025	0.020	0.011
Ours	✓	0.023	0.023	0.009	0.022	0.016	0.007	0.629	0.626	0.605	0.024	0.021	0.008

improvement over the next-best streaming method. On TUM-dynamics, which contains real-world moving objects with diverse motion patterns, our method similarly achieves leading trajectory accuracy, suggesting that the static bias of RayMap predictions generalizes across different dynamic scenarios. CUT3R achieves comparable rotational RPE on ScanNet yet suffers from accumulated trajectory drift reflected by higher ATE, which our reset metric alignment and state-aware smoothing help to mitigate. On static ScanNet, our method matches the leading streaming method in ATE, suggesting that dynamic filtering via staticness weights does not degrade performance in the absence of moving objects.

4.3 3D Reconstruction

Following [7, 34], we evaluate 3D reconstruction on 7-Scenes [26] using accuracy, completion, normal consistency, and Chamfer distance, running 200 frames per scene. As shown in Tab. 3, RayMap3R achieves leading reconstruction quality among streaming methods across the majority of metrics. In particular, our method achieves the lowest accuracy error across mean, median, and minimum statistics among all streaming approaches. Compared to CUT3R, RayMap3R

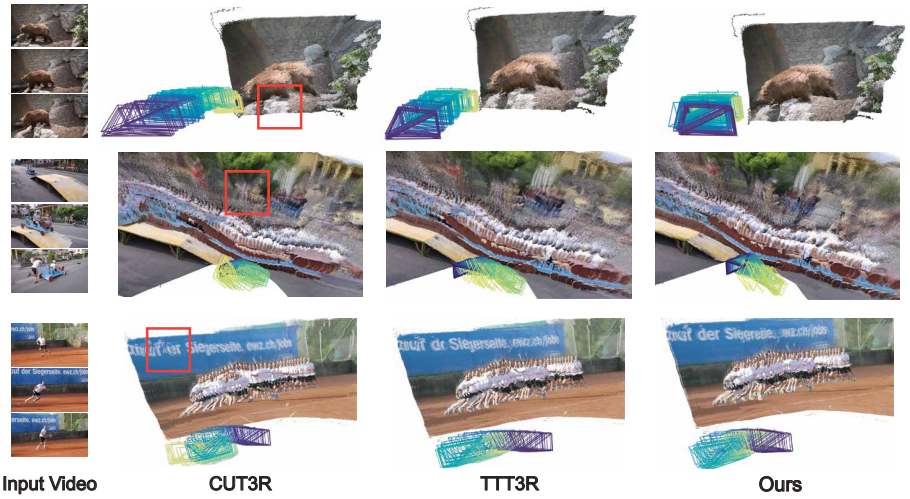

Fig. 5: Qualitative Results on DAVIS [22] Videos. We compare our method with CUT3R [34] and TTT3R [7]. Our method achieves more stable camera pose estimation and produces clearer reconstructions.

yields consistent improvements in both accuracy and completion across all statistics, as filtering dynamic content before state updates prevents corrupted geometry from accumulating in memory. Against TTT3R, RayMap3R achieves lower accuracy across all statistics and better completion on mean and median, with improved Chamfer on mean and minimum and comparable median. Although Spann3R attains higher normal consistency, RayMap3R outperforms it substantially on all other geometric metrics, reflecting the benefit of dynamic-aware memory updates. Furthermore, RayMap3R surpasses MonST3R-GA in accuracy and Chamfer distance, suggesting that selective state updates can partially compensate for the absence of global optimization in streaming reconstruction.

4.4 Qualitative Results

We compare our method with CUT3R [34] and TTT3R [7] on dynamic sequences from the DAVIS dataset [22] as shown in Fig. 5. CUT3R suffers from noticeable camera drift as dynamic content corrupts the memory state, leading to distorted structures and misaligned surfaces. TTT3R improves stability but still drifts when large dynamic regions dominate the scene, as soft gating cannot fully suppress their influence on state updates. In contrast, RayMap3R produces smoother trajectories and more geometrically faithful reconstructions across all sequences, demonstrating consistent robustness to diverse dynamic scenarios. Notably, in the bottom row, the text on the boat surface is clearly legible in our reconstruction, whereas CUT3R collapses the geometry and TTT3R produces blurred results, illustrating that static-region querying preserves fine-grained geometric details that dynamic interference would otherwise corrupt.

Table 4: Inference speed and GPU memory. RayMap3R achieves competitive throughput and reasonable memory.

Method	50 views		1000 views	
	Mem ↓	FPS ↑	Mem ↓	FPS ↑
MonST3R [42]	32.0	0.31	OOM	OOM
VGGT [33]	20.0	21.0	OOM	OOM
Point3R [37]	30.0	5.0	OOM	OOM
CUT3R [34]	6.4	19.7	6.5	19.7
TTT3R [7]	7.6	19.6	7.7	19.6
Ours	9.2	13.8	9.4	13.8

Table 5: Component ablation. R: dual-branch; M: metric alignment; S: state-aware smoothing.

	Base	R	R+M	R+S	**Full**
ATE ↓	Camera Pose				
	0.114	0.113	0.110	0.084	**0.081**
AbsRel ↓	Video Depth				
	0.208	0.186	0.184	0.185	**0.183**
Chamfer ↓	3D Reconstruction				
	0.322	0.245	0.213	0.196	**0.170**

5 Analysis

Inference Speed and Memory Usage. We evaluate inference speed and GPU memory on ScanNet using a single NVIDIA RTX A6000 48GB GPU. As shown in Tab. 4, offline methods [33,42] require all frames before processing and are not viable for streaming scenarios; their memory reaches or exceeds 20 GB at 50 views and grows prohibitively beyond 200 views. Point3R [37] is online but maintains an explicit spatial pointer memory that grows with sequence length, resulting in reduced throughput and out-of-memory beyond 900 views. RayMap3R adopts the implicit-state design, maintaining stable memory across all sequence lengths; the dual-branch design introduces a moderate speed overhead, though accuracy gains are consistent across pose and depth benchmarks (Tabs. 1 and 2).

Component Ablation. We ablate three key components of RayMap3R using CUT3R as the base: dual-branch RayMap identification (R), reset metric alignment (M), and state-aware smoothing (S), with results in Tab. 5. R yields the most substantial improvement in depth and 3D reconstruction, as suppressing dynamic interference reduces per-frame estimation error and prevents corrupted geometry from accumulating in memory, with modest effect on trajectory since filtering operates at the state level. M produces notable Chamfer improvement by correcting scale misalignment at reset boundaries, leading to more globally consistent point clouds. S primarily improves trajectory accuracy by adaptively filtering pose noise based on internal state changes. The full model achieves consistent gains across all tasks, indicating that R, M, and S address complementary aspects of streaming reconstruction in dynamic scenes.

Static Bias Analysis. We quantitatively evaluate the dynamic map across 108 sequences using four metrics summarized in Tab. 6: disc measures the mean discrepancy ratio between dynamic and static regions; Mean AUC casts the signal as a binary classifier, where 0.5 denotes random chance; Spearman ρ measures its sequence-level correlation with ground-truth dynamic ratio; and Mean IoU measures the overlap between the dynamic map and ground-truth masks.

The discrepancy between main and RayMap-only predictions is consistently larger in dynamic regions than static ones across datasets (disc > 1), reflecting structured separation rather than random variation. Mean AUC exceeds 0.5

Table 6: Dynamic Map Evaluation. Evaluation across 108 sequences from three datasets. The depth discrepancy signal shows consistent discriminability and positive correlation with ground-truth dynamics across diverse scenes.

Dataset	Seq	Frames	Mean disc ↑	Mean AUC ↑	Mean IoU ↑	ρ ↑
MPI Sintel [2]	19	746	1.61	0.464	0.298	0.900
DAVIS 2017 [22]	81	5205	1.68	0.538	0.189	0.713
TUM RGB-D [27]	8	680	1.88	0.560	0.206	0.643
All	108	6631	1.69	0.532	0.203	0.771

on real-world datasets, indicating reliable discriminative performance; the lower AUC on Sintel is attributable to its more heterogeneous motion patterns, which make threshold-based classification harder. Spearman ρ further shows that sequences with greater dynamic content yield stronger detection, as illustrated in Fig. 2 (right): despite its lower AUC, Sintel achieves the highest ρ, reflecting that its wide range of dynamic ratios produces a strong and consistent ranking signal. The moderate IoU is expected, as the dynamic map is a continuous signal thresholded against binary ground-truth masks, designed as a soft gating cue rather than a precise segmentation output. Together, these findings suggest that the RayMap static bias generalizes reliably across diverse scene conditions.

Discussion and Scope. RayMap3R relies on the implicit dynamic awareness the model develops during training. This bias arises from the RayMap representation: encoding only camera geometry without appearance cues forces the model to rely on memory, which favors temporally consistent static structures over transient dynamic objects. RayMap3R targets streaming models with recurrent memory and native RayMap-only querying, which enables the training-free cue and defines the method's scope. Since the cue is inherited from the CUT3R's backbone, its reliability may depend on whether the model was trained on dynamic scenes. As RayMap representations spread to offline feed-forward models [14, 17], studying comparable query mechanisms in other RayMap-based architectures is a natural way to extend dynamic reconstruction at broader scales.

6 Conclusion

We presented RayMap3R, a training-free framework for streaming 3D reconstruction in dynamic scenes. By exploiting the static bias in RayMap-only predictions, our dual-branch inference scheme identifies dynamic regions without additional supervision or retraining. Combined with reset metric alignment and state-aware smoothing, RayMap3R suppresses dynamic interference while preserving metric consistency and real-time efficiency. Extensive experiments demonstrate state-of-the-art performance among streaming methods across camera pose estimation, video depth prediction, and 3D reconstruction, with particularly strong improvements on dynamic sequences. We hope this work motivates further exploration of implicit geometric biases in feed-forward models as a resource for dynamic scene understanding and reconstruction.

Acknowledgements

This research is supported by NSF IIS-2525840, CNS-2432534, ECCS-2514574 and Cisco Research unrestricted gift. This article solely reflects opinions and conclusions of authors and not funding agencies.

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building rome in a day. Communications of the ACM **54**(10), 105–112 (2011)
2. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12. pp. 611–625. Springer (2012)
3. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., Tardós, J.D.: Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. IEEE transactions on robotics **37**(6), 1874–1890 (2021)
4. Chen, W., Chen, L., Wang, R., Pollefeys, M.: Leap-vo: Long-term effective any point tracking for visual odometry. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19844–19853 (2024)
5. Chen, W., Zhang, G., Wimbauer, F., Wang, R., Araslanov, N., Vedaldi, A., Cremers, D.: Back on track: Bundle adjustment for dynamic scene reconstruction. arXiv preprint arXiv:2504.14516 (2025)
6. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Easi3r: Estimating disentangled motion from dust3r without training. arXiv preprint arXiv:2503.24391 (2025)
7. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Ttt3r: 3d reconstruction as test-time training. arXiv preprint arXiv:2509.26645 (2025)
8. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
9. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: Monoslam: Real-time single camera slam. IEEE transactions on pattern analysis and machine intelligence **29**(6), 1052–1067 (2007)
10. Deng, K., Ti, Z., Xu, J., Yang, J., Xie, J.: Vggt-long: Chunk it, loop it, align it–pushing vggt's limits on kilometer-scale long rgb sequences. arXiv preprint arXiv:2507.16443 (2025)
11. Duisterhof, B., Zust, L., Weinzaepfel, P., Leroy, V., Cabon, Y., Revaud, J.: Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. arXiv preprint arXiv:2409.19152 (2024)
12. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: Create anything in 3d with multi-view diffusion models. arXiv preprint arXiv:2405.10314 (2024)
13. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The international journal of robotics research **32**(11), 1231–1237 (2013)
14. Keetha, N., Müller, N., Schönberger, J., Porzi, L., Zhang, Y., Fischer, T., Knapitsch, A., Zauss, D., Weber, E., Antunes, N., et al.: Mapanything: Universal feed-forward metric 3d reconstruction. arXiv preprint arXiv:2509.13414 (2025)
15. Kopf, J., Rong, X., Huang, J.B.: Robust consistent video depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1611–1621 (2021)

16. Li, Z., Tucker, R., Cole, F., Wang, Q., Jin, L., Ye, V., Kanazawa, A., Holynski, A., Snavely, N.: Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. arXiv preprint arXiv:2412.04463 (2024)
17. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
18. Lu, J., Huang, T., Li, P., Dou, Z., Lin, C., Cui, Z., Dong, Z., Yeung, S.K., Wang, W., Liu, Y.: Align3r: Aligned monocular depth estimation for dynamic videos. arXiv preprint arXiv:2412.03079 (2024)
19. Maggio, D., Lim, H., Carlone, L.: Vggt-slam: Dense rgb slam optimized on the sl (4) manifold. arXiv preprint arXiv:2505.12549 (2025)
20. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: Orb-slam: A versatile and accurate monocular slam system. IEEE transactions on robotics **31**(5), 1147–1163 (2015)
21. Palazzolo, E., Behley, J., Lottes, P., Giguere, P., Stachniss, C.: Refusion: 3d reconstruction in dynamic environments for rgb-d cameras exploiting residuals. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 7855–7862. IEEE (2019)
22. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 724–732 (2016)
23. Pollefeys, M., Nistér, D., Frahm, J.M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.J., Merrell, P., et al.: Detailed real-time urban 3d reconstruction from video. International Journal of Computer Vision **78**, 143–167 (2008)
24. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024)
25. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
26. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2930–2937 (2013)
27. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: 2012 IEEE/RSJ international conference on intelligent robots and systems. pp. 573–580. IEEE (2012)
28. Sucar, E., Lai, Z., Insafutdinov, E., Vedaldi, A.: Dynamic point maps: A versatile representation for dynamic 3d reconstruction. arXiv preprint arXiv:2503.16318 (2025)
29. Team, A., Zhu, H., Wang, Y., Zhou, J., Chang, W., Zhou, Y., Li, Z., Chen, J., Shen, C., Pang, J., et al.: Aether: Geometric-aware unified world modeling. arXiv preprint arXiv:2503.18945 (2025)
30. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. Advances in neural information processing systems **34**, 16558–16569 (2021)
31. Teed, Z., Lipson, L., Deng, J.: Deep patch visual odometry. Advances in Neural Information Processing Systems **36**, 39033–39051 (2023)
32. Wang, H., Agapito, L.: 3d reconstruction with spatial memory. arXiv preprint arXiv:2408.16061 (2024)

33. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. arXiv preprint arXiv:2503.11651 (2025)
34. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. arXiv preprint arXiv:2501.12387 (2025)
35. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
36. Wimbauer, F., Chen, W., Muhle, D., Rupprecht, C., Cremers, D.: Anycam: Learning to recover camera poses and intrinsics from casual videos. arXiv preprint arXiv:2503.23282 (2025)
37. Wu, Y., Zheng, W., Zhou, J., Lu, J.: Point3r: Streaming 3d reconstruction with explicit spatial pointer memory. arXiv preprint arXiv:2507.02863 (2025)
38. Xu, K., Tse, T.H.E., Peng, J., Yao, A.: Das3r: Dynamics-aware gaussian splatting for static scene reconstruction. arXiv preprint arXiv:2412.19584 (2024)
39. Yang, J., Sax, A., Liang, K.J., Henaff, M., Tang, H., Cao, A., Chai, J., Meier, F., Feiszli, M.: Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21924–21935 (2025)
40. Yu, C., Liu, Z., Liu, X.J., Xie, F., Yang, Y., Wei, Q., Fei, Q.: Ds-slam: A semantic visual slam towards dynamic environments. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1168–1174. IEEE (2018)
41. Zhang, J.Y., Lin, A., Kumar, M., Yang, T.H., Ramanan, D., Tulsiani, S.: Cameras as rays: Pose estimation via ray diffusion. arXiv preprint arXiv:2402.14817 (2024)
42. Zhang, J., Herrmann, C., Hur, J., Jampani, V., Darrell, T., Cole, F., Sun, D., Yang, M.H.: Monst3r: A simple approach for estimating geometry in the presence of motion. arXiv preprint arXiv:2410.03825 (2024)
43. Zhang, Z., Cole, F., Li, Z., Rubinstein, M., Snavely, N., Freeman, W.T.: Structure and motion from casual videos. In: European Conference on Computer Vision. pp. 20–37. Springer (2022)
44. Zhao, W., Liu, S., Guo, H., Wang, W., Liu, Y.J.: Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In: European Conference on Computer Vision. pp. 523–542. Springer (2022)
45. Zhuo, D., Zheng, W., Guo, J., Wu, Y., Zhou, J., Lu, J.: Streaming 4d visual geometry transformer. arXiv preprint arXiv:2507.11539 (2025)

RayMap3R: Inference-Time RayMap for Dynamic 3D Reconstruction

Supplementary Material

Feiran Wang[1], Zezhou Shang[1], Gaowen Liu[2], and Yan Yan[1,†]

[1] University of Illinois Chicago, USA
[2] Cisco Research, USA

The appendix provides implementation details and model backbone descriptions (Secs. A–B), dynamic map visualizations and temporal consistency analysis (Secs. C–D), additional qualitative results (Sec. E), quantitative analysis of the depth discrepancy signal across scene conditions (Sec. F), a detailed description of reset metric alignment (Sec. G) and ablation analysis of state-aware smoothing (Sec. H), cross-dataset component ablation (Sec. I), the computation of the dynamic map (Sec. J), robustness under a different trained setup (Sec. K), generalization and hyperparameter sensitivity (Sec. L), the contribution of static and dynamic regions (Sec. M), relation to recent dynamic reconstruction methods (Sec. N), limitations (Sec. O), and a statement on LLM usage (Sec. P).

A. Implementation Details

Following previous works [3, 17], we use official implementations with default hyperparameters for all baseline methods to facilitate fair comparison. All experiments are conducted on a single NVIDIA RTX A6000 GPU with 48GB memory. RayMap3R builds upon the pre-trained CUT3R backbone [17] without any fine-tuning or additional supervision, preserving the training-free nature of our approach. For streaming reconstruction, the memory state is periodically reset every 50 frames to mitigate state drift over long sequences, with reset metric alignment applied at each reset boundary as described in Sec. G. For camera pose estimation, predicted trajectories are aligned with ground truth via similarity transformation. For depth evaluation, median scaling is applied per sequence to remove scale ambiguity under the per-sequence alignment protocol.

B. Model Backbone

CUT3R [17] is trained via a multi-stage curriculum on 32 diverse datasets spanning synthetic and real-world scenarios, covering static and dynamic scenes, object-centric views, and indoor/outdoor environments, including CO3Dv2 [13], ARKitScenes [5], ScanNet++ [21], TartanAir [18], Waymo [16], MegaDepth [9], DL3DV [10], and DynamicStereo [7]. Stage 1 establishes foundational geometry on static data at lower resolution; Stage 2 incorporates dynamic scenes

† Corresponding author.

with moving objects and partial annotations. As a result, the backbone develops implicit awareness of static-dynamic distinctions through the geometric-only nature of RayMap tokens. RayMap3R makes this implicit awareness explicit and actionable, exploiting it through training-free dual-branch inference to achieve dynamic-aware streaming reconstruction without extra supervision.

C. Dynamic Map Visualization

Fig. 1 visualizes the dynamic map across DAVIS [12] (rows 1–2), MPI Sintel [1] (rows 3–4), and TUM Dynamic [15] (row 5), covering diverse real-world and synthetic scenes with varied dynamic content, object scales, and motion patterns.

The main branch depth incorporates both image and RayMap features and captures full scene geometry including dynamic objects. The RayMap-only depth exhibits a static bias: background structures such as terrain, walls, and floors are reconstructed with reasonable fidelity and depth boundaries, while dynamic objects tend to appear blurred or geometrically inaccurate. This behavior arises because the RayMap branch relies solely on geometric consistency encoded in memory rather than per-frame appearance cues, causing it to favor temporally stable structure over transient dynamic content.

The dynamic map, computed as the per-pixel depth discrepancy between the two branches, highlights regions where the two predictions diverge. On DAVIS, it responds to a small waterbird in motion against a cluttered natural background and to a fast-moving motorcycle with rider, suggesting sensitivity across a range of object scales. On Sintel, moving figures and large foreground objects in cluttered synthetic environments produce clear responses in the dynamic map, while the surrounding static geometry remains relatively suppressed. On TUM, a walking person is highlighted against a static indoor background, with the dynamic map response concentrated on the person's silhouette.

Across all rows, the dynamic map shows reasonable spatial correspondence with the ground-truth masks without any mask supervision, suggesting that the depth discrepancy between branches provides a consistent proxy for dynamic content across real-world and synthetic conditions.

D. Temporal Consistency of Dynamic Maps

Fig. 2 shows the dynamic map across five frames from the DAVIS *longboard* sequence, where multiple subjects move continuously through the scene at varying distances from the camera. Across all frames, the dynamic map consistently produces elevated responses at moving subjects while assigning lower values to static background regions such as the path surface and surrounding vegetation.

In frames 35 and 37, a partially visible person appears at the left edge of the frame, and the dynamic map produces a clear response at that location despite the limited spatial extent of the subject. The response region follows each subject's spatial extent across frames as position and apparent scale change. As the primary subject moves further from the camera between frames 43 and 48,

Fig. 1: Dynamic Map Visualization. We visualize the dynamic map inferred by RayMap3R across DAVIS (rows 1–2), MPI Sintel (rows 3–4), and TUM Dynamic (row 5). The RayMap-only branch suppresses dynamic objects relative to the main branch, and the resulting dynamic map aligns with ground-truth masks.

the response area contracts correspondingly. While some background activations are visible in certain frames, the signal remains predominantly concentrated on the dynamic subjects throughout the sequence. These observations suggest that the depth discrepancy between the main and RayMap-only branches provides a temporally stable proxy for dynamic content.

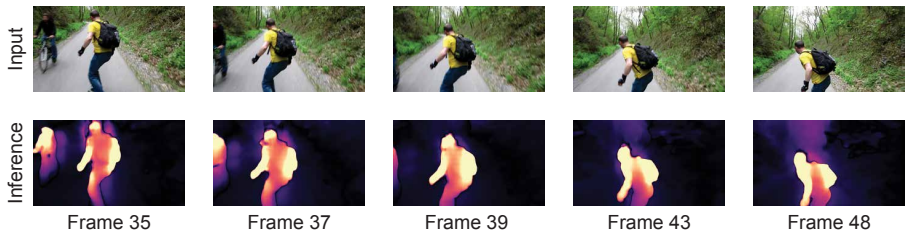

Fig. 2: Temporal Consistency of Dynamic Maps on DAVIS *longboard*. Top: input frames. Bottom: dynamic maps. The dynamic map consistently produces elevated responses at moving subjects, including partially visible persons at the frame boundary.

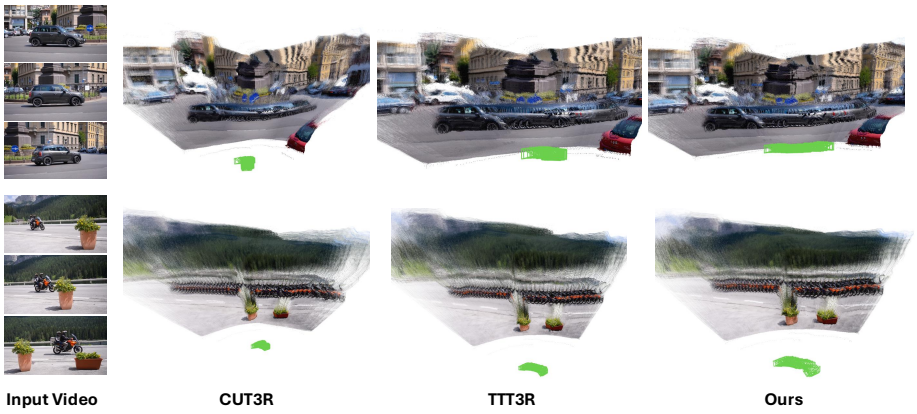

Fig. 3: Additional qualitative comparisons on dynamic scenes. A moving car (top) and motorcycle (bottom); from left to right: input frames, CUT3R [17], TTT3R [3], and Ours. Green denotes the estimated camera poses. RayMap3R yields cleaner static geometry and more stable camera trajectories.

E. Additional Qualitative Results

Fig. 3 shows additional reconstruction comparisons on dynamic sequences—a moving car (top) and a moving motorcycle (bottom)—against CUT3R [17] and TTT3R [3]. The baselines smear the moving object into ghosted geometry, distort the surrounding static structure, and produce drifting camera poses (green), whereas RayMap3R suppresses the dynamic interference, recovering cleaner static reconstructions with more stable camera trajectories.

F. Dynamic Map Analysis across Scene Conditions

To evaluate whether the depth discrepancy signal remains reliable across varying levels of dynamic content, we stratify 131 sequences from MPI Sintel [1], DAVIS 2017 [12], TUM RGB-D [15], and ScanNet [4] by their mean ground-truth dynamic ratio into three groups: Low ($\leq$10%, 66 sequences), Medium (10–30%, 45 sequences), and High (>30%, 20 sequences). This stratification spans the full spectrum from predominantly static scenes to scenes with large dynamic foreground objects, allowing us to assess signal behavior under diverse conditions.

As reported in Tab. 1, the mean δ increases monotonically across groups, consistent with the expectation that overall depth discrepancy grows with dynamic content. The disc metric exceeds 1 across all three groups, indicating that dynamic pixels tend to produce higher discrepancy than static pixels regardless of scene composition. The disc value in the Low group is particularly notable: even when dynamic objects occupy less than 10% of the frame, the discrepancy in dynamic regions is substantially larger than in static ones, suggesting that the separation is not solely driven by scene statistics. The decline in disc at

higher dynamic ratios is consistent with the observation that sustained exposure to dynamic content can gradually corrupt the memory representation of static regions, raising their baseline discrepancy and compressing the ratio, while the absolute separation remains positive throughout. The AUC exceeds 0.5 across all groups, suggesting consistent pixel-level discriminability, with a modest decline at higher dynamic ratios that may reflect increased scene complexity. The IoU increases monotonically with dynamic ratio, which reflect the geometric advantage of larger dynamic regions for threshold-based localization.

Together, these results suggest that the depth discrepancy between branches provides a structurally consistent signal for dynamic identification across diverse scene conditions, remaining discriminative even when dynamic content is sparse.

Table 1: Dynamic Map Evaluation by Scene Dynamic Ratio. We extend the evaluation in Tab. 6 of the main paper by stratifying sequences into three groups based on their mean ground-truth dynamic ratio: Low (≤10%), Medium (10–30%), and High (>30%). The disc metric exceeds 1 across all groups, indicating that the depth discrepancy signal remains discriminative even when dynamic content is sparse.

Dynamic Ratio	Seq	Frames	Mean δ	Mean disc ↑	Mean AUC ↑	Mean IoU ↑
Low (≤10%)	66	4168	0.065	2.24	0.591	0.145
Med (10–30%)	45	2862	0.077	1.93	0.551	0.240
High (>30%)	20	985	0.153	1.36	0.532	0.409

G. Reset Metric Alignment

Memory-based streaming models periodically reset their latent state to mitigate state drift over long sequences [3, 17, 19]. While effective for stability, each reset alters the model's internal scene representation, causing it to produce inconsistent metric scale and pose for the same repeated frame when processed before and after the reset. This discrepancy can manifest as scale mismatch between segments, which may propagate as drift throughout subsequent frames.

Our reset metric alignment, illustrated in Fig. 4, addresses this by using the repeated frame as an anchor for cross-segment correction. Let $\mathbf{P}^-$ and $\mathbf{P}^+$ denote the point clouds predicted from the repeated frame before and after reset, respectively. Since both are predicted from the same input frame at identical pixel positions, correspondences are directly available without extra matching step. To focus the alignment on reliable scene structure, we use the pixel-level staticness map computed at the final pre-reset frame as per-point confidence weights, upweighting stable background regions and downweighting potentially dynamic content. A Sim(3) transformation is then estimated via confidence-weighted SVD, recovering the scale, rotation, and translation discrepancy introduced by the reset. The estimated transformation is applied to all subsequent poses and accumulated point clouds in the new segment, reducing metric inconsistency.

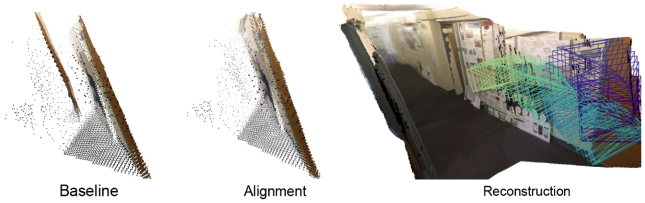

Fig. 4: Effect of Reset Metric Alignment. Baseline (*left*) produces fragmented point clouds across reset boundaries. Our reset metric alignment restores segment consistency (*middle*), yielding coherent full-sequence reconstruction (*right*).

H. State-Aware Smoothing Analysis

We provide additional analysis of state-aware smoothing introduced in Sec. 3.5, which stabilizes trajectory estimation by adaptively filtering inter-frame displacements. Specifically, we evaluate the contribution of each signal component, reporting Mean ATE and Mean RPE_t averaged across Sintel, TUM-dynamics, and ScanNet under the same Sim(3) alignment protocol as in Tab. 2 of the main paper. As state-aware smoothing primarily targets translational displacements, RPE_r shows negligible variation across settings and is not reported.

As shown in Tab. 2, the smoothing coefficient $\beta_t \in [0, 1]$ balances reliance on the current displacement prediction against accumulated history. Fixed-β variants reduce RPE_t over the unsmoothed baseline. However, uniform filtering cannot distinguish stable frames from uncertain ones, yielding limited improvement in ATE relative to the full adaptive scheme. Using a_t alone yields ATE comparable to fixed-β baselines. Trajectory acceleration alone cannot distinguish genuine motion dynamics from prediction noise, making a_t an insufficiently selective signal. Using sc_t alone achieves competitive RPE_t but degrades ATE. State change magnitude tends to be large at transitional intervals such as warmup and reset boundaries, causing over-smoothing that introduces systematic displacement bias and propagates as long-range drift. The product $a_t \times \text{sc}_t$ restricts strong smoothing to frames where both signals are elevated, identifying noisy predictions while preserving reliable motion estimates.

Table 2: Smoothing Signal Ablation. We compare the full adaptive smoothing ($a_t \times \text{sc}_t$) against fixed-coefficient baselines and individual signal components. The full scheme achieves the lowest ATE, as jointly conditioning on both signals avoids under-suppression from fixed filtering and drift from single-signal over-smoothing.

	Baseline	**Fix** β=0.3	**Fix** β=0.5	**Fix** β=0.8	**Only** a_t	**Only** sc_t	**Full** $a_t \times \text{sc}_t$
Mean ATE ↓	0.112	0.091	0.092	0.095	0.091	0.120	0.080
Mean RPE_t ↓	0.036	0.022	0.026	0.033	0.028	0.021	0.021

I. Component Ablation

To complement the ablation in Tab. 5 of the main paper, which samples sequences randomly across datasets for each task, we evaluate each component on complete individual datasets: camera pose estimation on TUM-dynamics [15], video depth on KITTI [6], and 3D reconstruction on 7-Scenes [14]. As shown in Tab. 3, R yields the largest improvement in video depth, as suppressing dynamic interference during state updates directly reduces per-frame depth error. S produces the most substantial gain in camera pose accuracy, as adaptive smoothing attenuates pose noise accumulated over dynamic sequences. M provides a consistent benefit to 3D reconstruction by correcting scale misalignment at reset boundaries, leading to more globally coherent point clouds. The full model achieves the best results across all metrics, suggesting that R, M, and S address complementary aspects of streaming reconstruction.

Table 3: Cross-Dataset Component Ablation. Each task is evaluated on a representative dataset. R: dual-branch RayMap identification; M: reset metric alignment; S: state-aware smoothing. Each component contributes gains across tasks, and the full model achieves the best results across metrics.

	Camera Pose (TUM-dyn)		Video Depth (KITTI)		3D Recon (7-Scenes)	
Setting	ATE ↓	RPE_t ↓	AbsRel ↓	δ<1.25 ↑	Acc ↓	Chamfer ↓
Base	0.033	0.011	0.120	88.3	0.041	0.029
+R	0.032	0.010	0.102	92.2	0.032	0.026
+R+M	0.031	0.008	0.100	92.5	0.028	0.025
+R+S	0.020	0.006	0.101	92.3	0.026	0.025
Full	**0.019**	**0.005**	**0.099**	**92.7**	**0.023**	**0.024**

J. Dynamic Map Computation

We detail the aggregation that converts pixel-level depth discrepancies into the per-state-token staticness scores used for memory gating in Sec. 3.3. For each pixel i, the relative depth discrepancy δ_i between the main and RayMap-only branches is first pooled into an image-token score δ_k^{tok} by confidence-weighted averaging over the pixels covered by token k, and then projected onto state token j using the decoder's cross-attention weights A_{jk}:

$$\delta_k^{\text{tok}} = \frac{\sum_{i \in k} w_i \, \delta_i}{\sum_{i \in k} w_i}, \qquad \delta_j^{\text{state}} = \frac{\sum_k A_{jk} \, \delta_k^{\text{tok}}}{\sum_k A_{jk}}, \tag{1}$$

where $i \in k$ denotes the pixels belonging to image token k, w_i is the confidence predicted by the main branch, and A_{jk} is the decoder cross-attention weight between state token j and image token k. The resulting δ^{state} is mapped to the staticness weights α_t as described in Sec. 3.3, so that state tokens associated with dynamic regions receive attenuated updates. The gating rule is thus fully deterministic: it introduces no learned parameters and reuses the decoder cross-attention already computed during inference.

Table 4: Robustness under a different trained setup. Camera-pose estimation on TUM-Dynamics using an intermediate CUT3R checkpoint trained with a Linear head. Our inference scheme remains effective across the alternative backbone variant. Best in **bold**.

Method	ATE↓	RPE_t↓	RPE_r↓
CUT3R-linear	0.044	0.010	0.344
TTT3R-linear	0.027	0.010	**0.325**
Ours-linear	**0.021**	**0.005**	0.331

K. Robustness under a Different Trained Setup

The dynamic cue in RayMap3R is an emergent static-memory tendency of the pretrained backbone rather than a trained component. To examine whether this property is specific to the final released checkpoint, we evaluate an intermediate CUT3R checkpoint trained with a Linear prediction head and apply our inference scheme on top of it. As shown in Tab. 4, on TUM-Dynamics our method (Ours-linear) still improves camera-pose accuracy over both CUT3R-linear and TTT3R-linear, achieving the lowest ATE and translational RPE. This indicates that the RayMap static bias, and the gains it enables, are not an artifact of one specific training configuration and persist under a different trained setup.

L. Generalization and Hyperparameter Sensitivity

Additional dynamic benchmark. To probe generalization beyond the benchmarks used in the main paper, we evaluate camera-pose estimation on the dynamic Bonn benchmark [11]. As shown in Tab. 5, RayMap3R improves over CUT3R across all three pose metrics, confirming that the static-memory tendency transfers to an additional real-world dynamic dataset.

Hyperparameter sensitivity. RayMap3R uses a single fixed inference setting across all reported evaluations. The two main hyperparameters are γ, which controls the static–dynamic separation of the dual-branch remapping, and λ, which controls the smoothing strength under unreliable predictions. Tab. 6 sweeps each from $0.5\times$ to $2\times$ its default value on Bonn: the ATE varies only marginally, indicating low parameter sensitivity. Because the method is training-free and applied with one fixed configuration, this stability also mitigates the concern that the reported gains might stem from per-dataset tuning on the test set.

M. Contribution of Static and Dynamic Regions

A natural question is whether RayMap3R reconstructs moving objects or simply discards them. To answer this, we run a filter-out test on Sintel in which the RayMap discrepancy cue used for memory gating is restricted to static regions

Table 5: Bonn pose evaluation. RayMap3R improves over CUT3R on the additional dynamic Bonn benchmark. Best in **bold**.

Method	ATE↓	RPE_t↓	RPE_r↓
CUT3R	0.0334	0.0086	0.6764
Ours	**0.0223**	**0.0067**	**0.6533**

Table 6: Sensitivity sweep of γ and λ on Bonn (ATE↓). Performance is stable across settings.

Param	0.5×	Default	2×
γ	0.0223	0.0223	0.0225
λ	0.0226	0.0223	0.0221

Table 7: Static and dynamic region contributions. Filter-out test on Sintel (ATE↓): the gating cue is restricted to static regions, to dynamic regions, or to both. Using both is best (**bold**).

Input	ATE↓
Static only	0.182
Dynamic only	0.220
Static + dynamic	**0.162**

only, to dynamic regions only, or to both. As shown in Tab. 7, using both regions yields the lowest ATE, and removing either the static or the dynamic regions degrades performance. This shows that both background structure and object regions contribute to reconstruction: the dynamic map acts as a soft, continuous gate that down-weights rather than hard-masks dynamic content, which also explains why a moderate mean dynamic-map IoU (around 0.20, Tab. 6 of the main paper) is sufficient to yield consistent downstream gains.

N. Relation to Recent Dynamic Reconstruction Methods

A range of methods address dynamic scenes through dedicated mechanisms. Semantic SLAM systems such as DS-SLAM [22] remove dynamic objects using semantic segmentation priors; trajectory- and optimization-based methods such as ParticleSfM [24], CasualSAM [23], and Robust-CVD [8] recover camera motion and consistent depth from casual dynamic videos via dense point trajectories or test-time optimization; and dynamics-aware reconstruction methods such as DAS3R [20] explicitly model dynamic content to recover the static scene. In contrast, RayMap3R requires no segmentation network, flow estimator, or per-scene optimization: its dynamic cue is derived purely from the discrepancy between the model's own RayMap-only and image-based predictions and is applied online within a streaming memory.

The most closely related approach is Easi3R [2], which repurposes DUSt3R's cross-attention for training-free motion segmentation. RayMap3R differs in two key respects: (i) it operates *online* within a recurrent memory rather than performing offline segmentation over a full sequence, and (ii) its cue is a *geometric* depth discrepancy between two native query modes that gates memory updates,

rather than an attention map used to segment and reweight regions. These differences make our cue directly usable for streaming state gating without any per-sequence optimization.

We further clarify two design choices. The reset metric alignment uses a Sim(3) transformation, which is a standard tool in SLAM and chunk-based reconstruction; our contribution is not the transformation itself but the use of the repeated reset frame as an anchor to correct the metric drift introduced by memory resets while retaining their stabilizing effect (Sec. 3.4). For state-aware smoothing, large state-update magnitudes tend to coincide with abrupt camera motion and unreliable predictions; the product $a_t \times \mathrm{sc}_t$ isolates these cases, so the three components (remapping, reset alignment, and smoothing) all draw on the same internal signals—the RayMap discrepancy and the state-update magnitude—rather than being unrelated heuristics (Sec. 3.5).

O. Limitations

RayMap3R is built on CUT3R's dual-branch design, in which geometry can be queried either from images together with RayMap tokens or from RayMap tokens alone. Our dynamic cue depends directly on this property: it is the discrepancy between these two *native* query modes that exposes dynamic content. Consequently, the approach is tailored to backbones that maintain an explicit recurrent memory state and support a RayMap-only query path, and it does not transfer directly to feed-forward models lacking such a mode. Moreover, because the cue is an emergent static bias of the pretrained model rather than a trained component, its strength is inherited from the training distribution. As studied in Tab. 4, the bias persists under an alternative trained setup, but a backbone trained with substantially less dynamic data could exhibit a weaker bias and reduce the effectiveness of the dual-branch scheme. We view making this implicit bias explicit, and extending it to other RayMap-based architectures including offline feed-forward models, as an important direction for future work.

P. LLM Usage

Large language models were used to assist with language polishing in the preparation of this manuscript. All technical content, experimental design, and conclusions are the work of the authors.

References

1. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12. pp. 611–625. Springer (2012)
2. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Easi3r: Estimating disentangled motion from dust3r without training. arXiv preprint arXiv:2503.24391 (2025)

3. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Ttt3r: 3d reconstruction as test-time training. arXiv preprint arXiv:2509.26645 (2025)
4. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017)
5. Dehghan, A., Baruch, G., Chen, Z., Feigin, Y., Fu, P., Gebauer, T., Kurz, D., Dimry, T., Joffe, B., Schwartz, A., et al.: Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. NeurIPS Datasets and Benchmarks **2**(6), 16 (2021)
6. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. The international journal of robotics research **32**(11), 1231–1237 (2013)
7. Karaev, N., Rocco, I., Graham, B., Neverova, N., Vedaldi, A., Rupprecht, C.: Dynamicstereo: Consistent dynamic depth from stereo videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13229–13239 (2023)
8. Kopf, J., Rong, X., Huang, J.B.: Robust consistent video depth estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1611–1621 (2021)
9. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2041–2050 (2018)
10. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., et al.: Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22160–22169 (2024)
11. Palazzolo, E., Behley, J., Lottes, P., Giguere, P., Stachniss, C.: Refusion: 3d reconstruction in dynamic environments for rgb-d cameras exploiting residuals. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 7855–7862. IEEE (2019)
12. Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 724–732 (2016)
13. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10901–10911 (2021)
14. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2930–2937 (2013)
15. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: 2012 IEEE/RSJ international conference on intelligent robots and systems. pp. 573–580. IEEE (2012)
16. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)
17. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. arXiv preprint arXiv:2501.12387 (2025)

18. Wang, W., Zhu, D., Wang, X., Hu, Y., Qiu, Y., Wang, C., Hu, Y., Kapoor, A., Scherer, S.: Tartanair: A dataset to push the limits of visual slam. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4909–4916 (2020)
19. Wu, Y., Zheng, W., Zhou, J., Lu, J.: Point3r: Streaming 3d reconstruction with explicit spatial pointer memory. arXiv preprint arXiv:2507.02863 (2025)
20. Xu, K., Tse, T.H.E., Peng, J., Yao, A.: Das3r: Dynamics-aware gaussian splatting for static scene reconstruction. arXiv preprint arXiv:2412.19584 (2024)
21. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023)
22. Yu, C., Liu, Z., Liu, X.J., Xie, F., Yang, Y., Wei, Q., Fei, Q.: Ds-slam: A semantic visual slam towards dynamic environments. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1168–1174. IEEE (2018)
23. Zhang, Z., Cole, F., Li, Z., Rubinstein, M., Snavely, N., Freeman, W.T.: Structure and motion from casual videos. In: European Conference on Computer Vision. pp. 20–37. Springer (2022)
24. Zhao, W., Liu, S., Guo, H., Wang, W., Liu, Y.J.: Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In: European Conference on Computer Vision. pp. 523–542. Springer (2022)