MV3R: LONG-SEQUENCE MULTI-VIEW 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

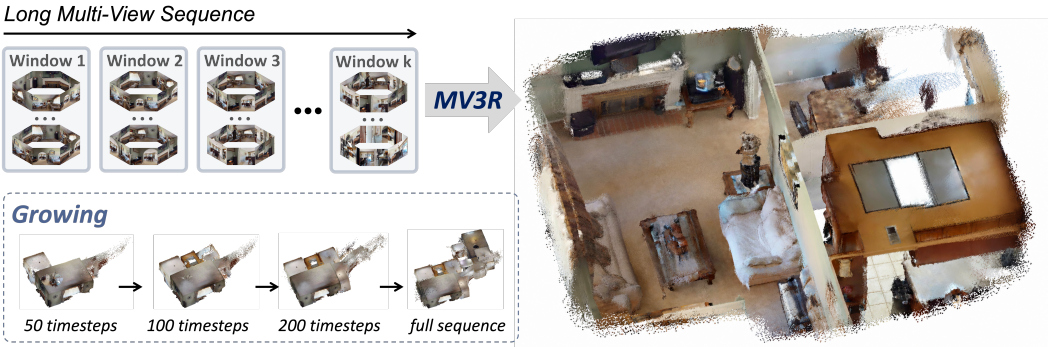

Figure 1: **MV3R is a long-sequence multi-view 3D reconstruction model designed for indoor environments.** It takes a stream of synchronized images from a fixed multi-camera array and progressively reconstructs a globally consistent 3D scene together with the camera-array trajectory. MV3R efficiently scales to thousands of images while suppressing geometric drift over long sequences.

ABSTRACT

Reconstructing 3D scenes from long-horizon multi-camera streams requires processing thousands of images without sacrificing global geometric consistency. Existing feed-forward methods, however, typically process all input views in a single forward pass and therefore scale poorly to long sequences, while streaming methods rely on a bounded state that may discard fine-grained cross-view evidence and accumulate drift over time. To overcome this trade-off, we introduce MV3R, a scalable window-based framework that processes a sequence as overlapping local windows and propagates context through their shared observations, thereby preserving continuity without jointly processing the entire sequence. Building on this windowed design, MV3R incorporates patch-level Plücker ray encodings to expose the spatial configuration of the fixed camera array and improve reconstruction within each window. The shared observations between adjacent windows then serve a second purpose by establishing deterministic correspondences for connecting local reconstructions. To robustly exploit these correspondences, a fixed-capacity scene memory predicts their reliability and guides a two-stage $\mathrm{Sim}(3)$ registration that performs confidence-weighted coarse alignment followed by residual refinement. Together, these designs enable MV3R to scale to long multi-camera sequences with bounded memory while bringing all local reconstructions into a geometrically consistent global coordinate system. To address the lack of benchmarks for indoor long-sequence reconstruction with fixed multi-camera arrays, we further introduce MV3D for evaluating geometric reconstruction quality and long-term trajectory stability. Extensive experiments show that MV3R significantly outperforms existing methods in reconstruction and trajectory estimation across four indoor datasets. Specifically, relative to the strongest competing result for each reconstruction metric on each test subset, MV3R reduces point-cloud Accuracy and Completeness errors by 22.5% and 22.1% on average, respectively. For trajectory estimation, MV3R reduces average ATE from 0.204 for the strongest baseline to 0.095, corresponding to an approximately 53.4% reduction.

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

Robots operating over long periods in indoor environments must continuously maintain a 3D representation of the scene to support localization, navigation, and interaction. To obtain more complete and stable observations, mobile robots are often equipped with multiple rigidly mounted cameras that continuously capture synchronized multi-view images from different directions. As the robot enters new areas, these observations must be promptly converted into 3D geometry and integrated into the existing scene. Consequently, neither reconstructing the entire scene only after a complete trajectory has been collected nor reprocessing all historical observations whenever new measurements arrive is suitable for long-term operation. Streaming reconstruction is therefore a natural formulation in which the system incrementally updates the 3D scene as multi-view observations arrive while keeping the computation and storage required by each update under control. Recent feed-forward 3D reconstruction models that directly predict camera parameters and dense geometry, such as DUSt3R, MASt3R, and VGGT, provide a promising foundation for such continuous multi-view 3D perception (Wang et al., 2024; Leroy et al., 2024; Wang et al., 2025a).

Extending feed-forward reconstruction to long-horizon multi-camera image streams requires satisfying two requirements simultaneously: **(i)** retaining sufficient local observations within a bounded context for reliable geometric reconstruction, and **(ii)** maintaining long-term consistency as observations continue to arrive without allowing resource consumption to grow with the history length. Recent streaming methods preserve historical information through recurrent states, explicit memories, or causal caches, enabling continuous scene updates without repeatedly processing the complete sequence (Wang et al., 2025b; Wu et al., 2025; Chen et al., 2025; Zhuo et al., 2025; Lan et al., 2025; Lu et al., 2026; Zou et al., 2026). However, a fixed-capacity state must continually compress or discard history and may therefore lose early or fine-grained scene information, whereas retaining more history increases computational and memory costs. Synchronized multi-camera input further intensifies this tension because each time step contains multiple correlated views, while local geometry also depends on observations from neighboring time steps. DVGT-2 has begun to explore this setting by fusing the synchronized views of the current time step through cross-view spatial attention and accessing a fixed-window historical cache through temporal causal attention for online multi-camera geometry prediction (Zuo et al., 2026b). Its design, however, is primarily targeted at real-time autonomous-driving perception and adopts a factorized treatment of cross-view and cross-time interactions. In indoor scenes, occlusions, repeated structures, and large viewpoint changes often make local geometry depend jointly on evidence across both cameras and time. Consequently, such step-wise factorization limits the model's ability to reason directly over the complete local spatiotemporal observation set. Increasing the historical cache alone also does not resolve the underlying trade-off between long-term coverage and bounded resources.

This tension motivates us to reconsider the update unit for long-sequence multi-camera reconstruction. We organize synchronized camera observations over a contiguous temporal interval into a fixed-size *spatiotemporal array window*, so that local reconstruction can jointly exploit view and temporal information within a bounded context, rather than requiring the long-term state to carry the full burden of local geometric reasoning. Existing long-sequence methods, including VGGT-Long, SwiftVGGT, and VGGT-Motion, have shown that local chunks or submaps can effectively extend feed-forward reconstruction to much longer sequences (Deng et al., 2025; Lee et al., 2025; Xiong et al., 2026). Once a window becomes the streaming update unit, however, long-term performance depends on reliable local geometry within each window and stable integration of these local reconstructions into a common coordinate system across windows. A fixed camera array provides stable cross-camera geometric relationships that can strengthen local reconstruction, but more accurate local geometry alone cannot prevent registration errors from accumulating along a long sequence. Maintaining global consistency additionally requires historical scene information beyond the current window (Xiong et al., 2026; Zhang et al., 2026a;b). This leads to a natural question: *while preserving incremental updates and fixed resource consumption, can the local spatiotemporal observations of a fixed camera array improve within-window reconstruction, and can long-term scene context improve the calibration from each window to the global scene?*

Based on these observations, we propose MV3R, a window-based streaming 3D reconstruction framework for fixed multi-view camera arrays, built around the principle of *local reconstruction, global calibration*. During **local reconstruction**, synchronized multi-camera observations are organized into spatiotemporal array windows, and patch-level ray encodings constructed from camera

intrinsics and the array layout inject projection and viewing-direction priors, enabling joint prediction of local camera poses and dense geometry. During **global calibration**, shared observations between adjacent windows establish deterministic correspondences. A fixed-capacity scene memory then uses historical context to predict correspondence reliability, which is used to weight the $\mathrm{Sim}(3)$ alignment, and further predicts a residual $\mathrm{Sim}(3)$ correction. In this way, synchronized multi-view observations strengthen reconstruction within each window, while long-term scene history improves global calibration across windows before alignment errors propagate. At the same time, both the local context and the global memory remain fixed in size.

To evaluate both local reconstruction and long-term global consistency, we construct MV3D, an indoor long-sequence 3D reconstruction benchmark for fixed multi-camera arrays, integrating diverse indoor scenes from Replica, Gibson, Matterport3D, and HM3D (Straub et al., 2019; Xia et al., 2018; Chang et al., 2017; Ramakrishnan et al., 2021). At each time step, the benchmark provides synchronized multi-view RGB images and camera intrinsics, together with ground-truth camera poses for supervision and evaluation. The held-out test trajectories contain 331–556 timesteps, corresponding to 1,986–3,336 images. Compared with the strongest baseline, VGGT-Motion (Xiong et al., 2026), MV3R reduces average ATE from 0.204 to 0.095, a 53.4% reduction. These results demonstrate that array-aware local reconstruction and history-conditioned global calibration can effectively suppress accumulated drift over long sequences.

Our main contributions are as follows:

- **Window-based streaming reconstruction for fixed camera arrays.** We jointly model synchronized multi-view and contiguous temporal observations within fixed-size spatiotemporal array windows for streaming reconstruction.
- **Memory-conditioned global calibration.** We use a fixed-capacity scene memory to predict the reliability of cross-window correspondences and a residual $\mathrm{Sim}(3)$ correction.
- **The MV3D indoor long-sequence benchmark.** We construct MV3D for evaluating geometric reconstruction and long-term trajectory consistency on complete test trajectories containing up to 3,336 images.

2 Related Work

2.1 Feed-Forward 3D Reconstruction

With the development of large-scale vision models, 3D reconstruction has increasingly shifted from per-scene optimization to feed-forward prediction. DUSt3R and MASt3R advance cross-view geometric reasoning (Wang et al., 2024; Leroy et al., 2024), while VGGT further unifies camera, depth, and 3D geometry prediction in a feed-forward framework (Wang et al., 2025a). DVGT and DriveVGGT subsequently extend this paradigm to multi-camera sequences in autonomous driving (Zuo et al., 2026a; Jia et al., 2025). However, these methods generally assume bounded-size inputs, with computation and memory costs growing with the number of observations, making them difficult to apply directly to continuous long-sequence streams.

2.2 Long-Sequence Streaming 3D Reconstruction

Existing methods can be broadly divided into two categories. The first continuously updates a scene state as new observations arrive. CUT3R, TTT3R, Point3R, and LONG3R explore this paradigm (Wang et al., 2025b; Chen et al., 2026b; Wu et al., 2025; Chen et al., 2025). StreamVGGT, STream3R, OVGGT, and RetrieveVGGT control the historical context used for online inference through causal processing, caching, or bounded retrieval (Zhuo et al., 2025; Lan et al., 2025; Lu et al., 2026; Zou et al., 2026). DVGT-2 considers synchronized multi-camera input, fusing current multi-view observations with cross-view spatial attention and accessing a fixed-length feature cache through temporal causal attention (Zuo et al., 2026b). These methods avoid repeatedly processing the full history, but the bounded state must both preserve past scene information and support current geometric reasoning. As sequences grow, early and fine-grained information is progressively compressed, while multi-camera input further consumes the limited context capacity. Thus, bounded states control computational cost but do not resolve the tension between long-term information retention and local geometric modeling.

The second category partitions long sequences into local chunks and progressively integrates them into a unified scene. SLAM3R explores this local-to-global strategy (Liu et al., 2025), while VGGT-Long and SwiftVGGT extend chunk-based processing to longer sequences (Deng et al., 2025; Lee et al., 2025). VGGT-Motion, TALO, and VGGT-Align focus on inter-chunk drift and long-term global consistency (Xiong et al., 2026; Zhang et al., 2026a;b). Compared with observation-wise updates, these methods preserve more complete short-term context within local windows and support joint geometric reasoning inside each window. However, their performance still depends on local reconstruction quality and cross-window registration reliability. For fixed multi-camera arrays, synchronized views within a window provide stable cross-camera geometric constraints, while historical scenes outside the window provide longer-term global context. Existing works largely address local-window modeling and long-term global consistency separately. Using both to improve local reconstruction and cross-window calibration for long fixed-array sequences remains underexplored.

3 METHOD

3.1 PROBLEM DEFINITION AND OVERVIEW

Given a long indoor image stream continuously captured by a fixed multi-camera array, the input is

$$\mathcal{I}_{1:T} = \{I_{t,v} \mid t = 1, \dots, T,\ v = 1, \dots, V\}, \tag{1}$$

where T is the number of time steps, V is the number of cameras, and $I_{t,v} \in \mathbb{R}^{H \times W \times 3}$ is the synchronized image captured by camera v at time t. We use a fixed reference camera in the array to represent the motion trajectory of the entire ego and denote its global poses by $\{\mathbf{E}_t\}_{t=1}^{T}$. The model is given camera intrinsics and the relative layout of the cameras within the array. Its objective is to recover dense 3D geometry and the array trajectory while transforming all local observations into a common global coordinate system.

The task presents three primary challenges. First, hundreds of timesteps correspond to thousands of input images, which are difficult for a feed-forward model to process jointly. Second, a fixed array provides a stable cross-camera layout prior, but generic visual features do not explicitly encode this structure. Third, scale, rotation, and translation errors between local windows can accumulate along the sequence, producing long-term trajectory drift.

As illustrated in Figure 2, MV3R addresses these challenges with two modules. An array-aware window-level streaming reconstruction module recovers local geometry through overlapping context propagation and array-geometry encoding. A memory-conditioned global calibration module establishes deterministic correspondences from adjacent-window overlaps, uses scene memory to predict reliability weights for coarse $\mathrm{Sim}(3)$ alignment, and predicts a residual $\mathrm{Sim}(3)$ correction using both memory scales.

3.2 ARRAY-AWARE WINDOW-LEVEL STREAMING RECONSTRUCTION

We partition the long sequence into fixed-length windows with temporal overlap. Each window contains all synchronized camera observations over a contiguous time interval. A modified VGGT backbone jointly predicts a local point map, camera poses, and patch-level features. Because the window length is fixed, the cost of each local reconstruction step is independent of the sequence length.

3.2.1 OVERLAPPING CONTEXT PROPAGATION

Adjacent windows share identical image observations. The current window re-extracts visual features for these images, whereas the previous window has already constructed contextual states for the same patches through multi-frame and multi-camera interaction.

Let $\mathbf{x}_{k,i}^{\mathrm{ov}}$ be the encoder feature of patch i in the overlap region of window k, and let $\mathbf{h}_{k-1,i}^{\mathrm{ov}}$ be the corresponding contextualized feature propagated from the previous window. Since the two features have deterministic position-wise correspondence, we fuse historical context through a residual update:

$$\widetilde{\mathbf{x}}_{k,i}^{\mathrm{ov}} = \mathbf{x}_{k,i}^{\mathrm{ov}} + \mathbf{W}_h \mathbf{h}_{k-1,i}^{\mathrm{ov}}, \tag{2}$$

where $\mathbf{W}_h$ is a learnable linear projection. Newly introduced observations in the current window retain their original encoder features. After window inference, we store the contextualized features,

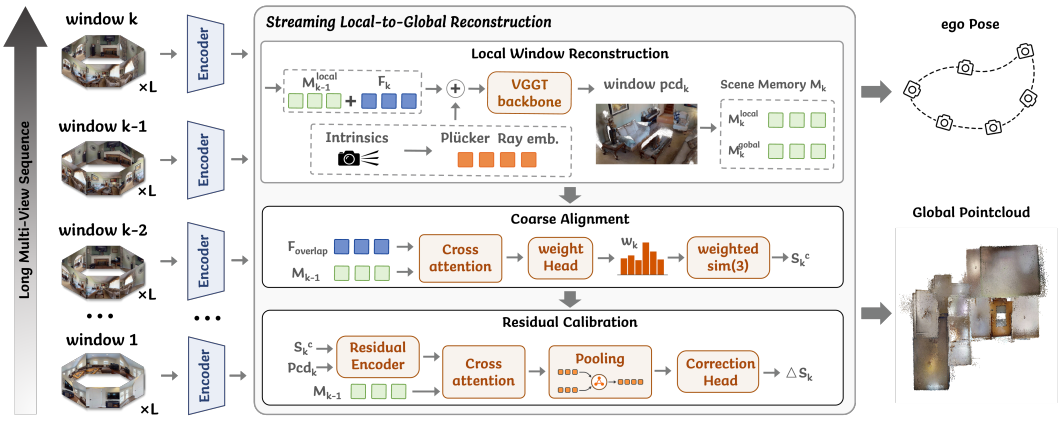

Figure 2: **Overview of MV3R.** MV3R processes a long-horizon multi-view image stream by partitioning it into overlapping local windows. The model recovers window-level geometry through array-aware local reconstruction and performs memory-conditioned global calibration, consisting of correspondence-weighted coarse alignment and memory-conditioned residual correction, to transform all window reconstructions into a shared global coordinate frame, ultimately producing a globally consistent 3D reconstruction.

global 3D positions, and geometric confidence for observations shared with the next window. These states provide deterministic historical feature and geometry correspondences for subsequent inference.

3.2.2 Array-Aware Geometric Encoding

The camera-array layout is represented by the rotation and translation of each camera relative to the fixed reference camera. We normalize all relative translations with a common scale and express every camera ray in the same array reference frame.

For patch i in camera v, let $\mathbf{d}_{v,i}$ be its unit ray direction and $\mathbf{o}_v$ be its normalized camera center. The corresponding Plücker ray and feature injection are

$$\boldsymbol{\ell}_{v,i} = (\mathbf{d}_{v,i},\ \mathbf{o}_v \times \mathbf{d}_{v,i}), \qquad \widehat{\mathbf{x}}_{k,t,v,i} = \mathbf{x}^{\mathrm{ctx}}_{k,t,v,i} + f_{\mathrm{ray}}(\boldsymbol{\ell}_{v,i}), \tag{3}$$

where $\mathbf{x}^{\mathrm{ctx}}_{k,t,v,i}$ denotes the visual feature after optional historical fusion and f_{ray} is a lightweight ray encoder. This encoding allows the VGGT backbone to directly perceive the relative positions and viewing directions of cameras in the fixed array.

The augmented features are processed by the modified VGGT backbone to produce the dense point map $\mathbf{P}_k$, camera poses, and contextualized features $\mathbf{F}_k$ for window k. These outputs support local geometry recovery, inter-window alignment, and subsequent state propagation.

3.3 Memory-Conditioned Global Calibration

Reconstructions from different windows lie in independent local coordinate systems. MV3R first establishes deterministic correspondences from adjacent-window overlaps and uses scene-memory-conditioned reliability weights to estimate a coarse $\mathrm{Sim}(3)$ transformation. It then queries the same scene memory to predict a residual correction for the remaining alignment error.

3.3.1 Fixed-Capacity Scene Memory

Before processing any window, the fixed-capacity scene memory is initialized as $\mathbf{M}_0 = \varnothing$, and its state after processing window k is denoted by $\mathbf{M}_k$. The local coordinate system of the first window is adopted as the global coordinate system and used to build the initial memory $\mathbf{M}_1$.

Each memory entry contains a global 3D position, an aligned contextualized VGGT feature, and geometric confidence. The 3D position is obtained by aggregating valid point-map predictions within the region covered by a patch, and pixel-level VGGT confidence is aggregated to the patch scale.

To balance dense recent geometry with long-term scene coverage, the scene memory is defined as $\mathbf{M}_k = \{\mathbf{M}_k^{\mathrm{loc}}, \mathbf{M}_k^{\mathrm{glob}}\}$. The local-memory branch $\mathbf{M}_k^{\mathrm{loc}}$ retains high-confidence point-feature entries from the most recent windows, whereas the global-memory branch $\mathbf{M}_k^{\mathrm{glob}}$ compresses long-term history through voxel merging and spatial sampling. Each update merges only candidates from the current window with the previous fixed-capacity scene-memory state. Memory size and per-step update cost therefore remain bounded.

3.3.2 Memory-Conditioned Correspondence-Weighted Coarse Alignment

The overlap of adjacent windows contains the same timesteps, cameras, and image patches, and therefore provides deterministic position-wise correspondences. Let $\mathbf{p}_{k,i}^{\mathrm{ov}}$ be a local 3D position in the current window and $\mathbf{q}_{k-1,i}^{\mathrm{ov}}$ be the corresponding global position stored by the previous window.

The scene memory $\mathbf{M}_{k-1}$ is not used to establish or re-search 3D correspondences. The correspondence set is determined exclusively by the shared timesteps, cameras, and patch locations in the adjacent-window overlap. Instead, the local and global branches of $\mathbf{M}_{k-1}$ provide recent and long-term context for predicting the reliability of each deterministic correspondence. A weight-prediction head combines the current-window feature, the corresponding previous-window feature, the retrieved scene-memory context, and VGGT geometric confidence to predict a reliability weight $w_{k,i}$. The coarse window transformation is obtained by weighted $\mathrm{Sim}(3)$ alignment:

$$\mathbf{S}_k^c = \underset{\mathbf{S}\in\mathrm{Sim}(3)}{\arg\min} \sum_i w_{k,i} \left\| \mathbf{q}_{k-1,i}^{\mathrm{ov}} - \mathbf{S}\mathbf{p}_{k,i}^{\mathrm{ov}} \right\|_2^2 . \tag{4}$$

VGGT-Long performs overlap-based chunk alignment with $\mathrm{Sim}(3)$, where correspondence weights are determined from VGGT confidence and Huber-based residuals through IRLS (Deng et al., 2025). In our setting, the camera-array layout likewise does not provide an accurate metric scale, and local window predictions may retain scale discrepancies. We therefore also adopt $\mathrm{Sim}(3)$ to jointly correct scale, rotation, and translation, but predict correspondence reliability from the current- and previous-window features, geometric confidence, and recent and long-term scene-memory context. The resulting weighted $\mathrm{Sim}(3)$ objective is solved in closed form using a differentiable weighted variant of the Umeyama alignment algorithm (Umeyama, 1991). This reliability weighting reduces the influence of occlusion, weak texture, repetitive structures, point-map outliers, and locally degenerate geometry.

3.3.3 Memory-Conditioned Residual Correction

Although the correspondence set used for coarse alignment is determined exclusively by adjacent-window overlap, the reliability weights are conditioned on scene memory. Nevertheless, the resulting coarse transformation may retain residual error when the overlap is small, noisy, or geometrically degenerate. To further exploit recent and long-term scene context at the window level, we introduce a memory-conditioned residual correction module.

For each valid correspondence, a residual extractor jointly encodes the current feature, the corresponding previous-window feature, the coordinate residual after coarse alignment, and the reliability weight. A context-fusion module then uses these residual tokens to query the local and global branches of the scene memory $\mathbf{M}_{k-1}$ and obtains a window-level representation through reliability-weighted pooling. A residual prediction head regresses a seven-dimensional $\mathrm{Sim}(3)$ Lie-algebra parameter $\Delta\boldsymbol{\xi}_k$, describing rotational, translational, and log-scale corrections.

Under a column-vector, left-multiplication convention, the final window transformation is

$$\Delta\mathbf{S}_k = \mathrm{Exp}_{\mathrm{Sim}(3)}\left(\Delta\boldsymbol{\xi}_k\right), \qquad \mathbf{S}_k = \Delta\mathbf{S}_k\mathbf{S}_k^c. \tag{5}$$

The final transformation is applied to the current point map and camera poses and is then used to update the propagated overlap state and the scene memory $\mathbf{M}_k$, including its local and global branches. For time steps that appear in multiple windows, we retain the prediction farthest from a window boundary to reduce boundary effects.

When the window size and memory capacity are fixed, both local reconstruction and memory interaction have per-window cost independent of the total sequence length, supporting long-sequence streaming inference with fixed window latency.

3.4 Training Objectives

We initialize the model from pretrained VGGT and jointly optimize the modified global-attention layers, prediction heads, and the newly introduced MV3R modules. Local reconstruction follows the standard VGGT geometric supervision for point maps, depth, and camera poses. For global calibration, we supervise correspondence reliability using the geometric consistency of overlapping predictions and directly optimize the final $\mathrm{Sim}(3)$ transformation against the ground-truth inter-window transformation. Historical contextual states and scene memory are detached between consecutive windows to avoid backpropagation through the full sequence. Detailed target construction, loss formulations, and the sequential training strategy are provided in Appendix B.

4 MV3D Benchmark

We introduce MV3D to evaluate long-horizon indoor 3D reconstruction from synchronized multi-view streams. Existing continuous indoor datasets such as 7-Scenes and Neural RGB-D are primarily monocular (Glocker et al., 2013; Azinović et al., 2022), while multi-view datasets such as Matterport3D are typically captured at discrete viewpoints without continuous camera trajectories (Chang et al., 2017). Simulation-based datasets including TartanAir and TartanGround partially satisfy both requirements, but are not designed for long-horizon indoor reconstruction with a fixed camera array (Wang et al., 2020; Patel et al., 2025). MV3D therefore combines synchronized multi-view observations, continuous camera motion, and dense metric geometry in a unified benchmark.

All sequences are rendered with Habitat-Sim (Savva et al., 2019). Detailed trajectory generation, camera configuration, annotation format, and data splits are provided in Appendix A.

5 Experiments

We evaluate MV3R on eight held-out test trajectories of MV3D, each from a distinct test scene, with two trajectories from each source dataset. We compare against offline models (VGGT, Pi3X (Wang et al., 2025c), DA3 (Lin et al., 2025), and DVGT), streaming models(StreamVGGT, LingBot-Map (Chen et al., 2026a), CUT3R, TTT3R (Chen et al., 2026b), Point3R, and InfiniteVGGT (Yuan et al., 2026)), and chunk-wise methods (TALO, VGGT-Align, VGGT-Motion, and VGGT-Long). All methods are evaluated using the same images and calibration protocol. **Trainable baselines are fine-tuned for two epochs on the same training split for fair comparison.** We report Acc., Comp., and NC for reconstruction, ATE for camera motion, and AUC@30 for ablations. Reconstruction uses 50 time steps (300 images), while long-sequence evaluation covers 331–556 time steps (1,986–3,336 images).

Table 1: 3D reconstruction results on four MV3D test subsets. MP3D denotes Matterport3D. Acc., Comp., and NC denote accuracy, completeness, and normal consistency, respectively. All methods use complete six-view observations over 50 time steps.

Type	Method	Replica			HM3D			MP3D			Gibson		
		Acc.↓	Comp.↓	NC↑	Acc.↓	Comp.↓	NC↑	Acc.↓	Comp.↓	NC↑	Acc.↓	Comp.↓	NC↑
Offline	VGGT	0.034	0.012	0.759	0.060	0.023	**0.704**	0.031	0.012	0.748	0.034	0.014	0.735
	Pi3X	0.037	0.018	0.727	0.081	0.028	0.640	0.038	0.022	0.698	0.043	0.027	0.688
	DA3	0.040	0.023	0.719	0.068	0.034	0.662	0.040	0.024	0.704	0.045	0.026	0.681
	DVGT	0.112	0.060	0.646	0.132	0.071	0.576	0.098	0.051	0.641	0.111	0.057	0.633
Streaming	StreamVGGT	0.048	0.019	0.748	0.072	0.029	0.686	0.034	0.016	0.732	0.037	0.018	0.728
	LingBot-Map	0.106	0.048	0.673	0.145	0.054	0.632	0.091	0.044	0.657	0.106	0.049	0.649
	CUT3R	0.057	0.021	0.688	0.092	0.040	0.668	0.052	0.025	0.676	0.053	0.034	0.661
	TTT3R	0.054	0.022	0.694	0.089	0.038	0.664	0.053	0.023	0.681	0.051	0.035	0.665
	Point3R	0.049	0.026	0.701	0.076	0.032	0.650	0.047	0.028	0.689	0.054	0.030	0.674
Chunk-wise	TALO	0.030	0.014	0.751	0.057	0.018	0.692	0.028	0.011	0.749	0.031	0.012	0.734
	VGGT-Align	0.034	0.016	0.747	0.063	0.021	0.687	0.032	0.013	0.744	0.034	0.014	0.731
	VGGT-Motion	0.036	0.017	0.744	0.062	0.022	0.688	0.030	0.014	0.746	0.037	0.015	0.727
	VGGT-Long	0.041	0.020	0.738	0.069	0.025	0.679	0.034	0.016	0.740	0.036	0.018	0.729
	MV3R	**0.022**	**0.009**	**0.784**	**0.048**	**0.016**	0.697	**0.021**	**0.008**	**0.757**	**0.024**	**0.009**	**0.738**

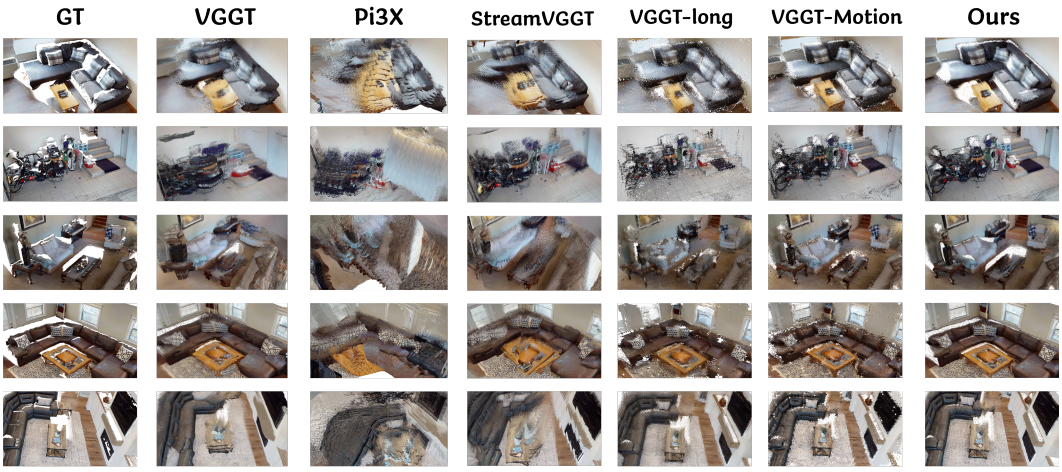

Figure 3: Qualitative 3D reconstruction comparison. From left to right: ground truth, VGGT, Pi3X, StreamVGGT, VGGT-Long, VGGT-Motion, and MV3R.

5.1 3D Reconstruction

We evaluate the first 50 time steps with six synchronized views (300 images) from each held-out trajectory, averaging results over two trajectories per source dataset.

As shown in Table 1, MV3R achieves the best result in 11 of the 12 metrics across the four test subsets and ranks second to VGGT only on HM3D NC. Relative to the strongest competing result for each metric on each test subset, MV3R reduces Acc. and Comp. errors by 22.5% and 22.1% on average, respectively, while increasing NC by approximately 0.9% on average. These consistent gains across Replica, HM3D, Matterport3D, and Gibson demonstrate improved geometric accuracy and completeness without sacrificing surface-normal quality.

Figure 3 shows that MV3R better preserves scene layout and object structure on the 300-image sequences. Competing methods exhibit more noise, missing surfaces, and geometric deformation.

5.2 Long-Sequence Camera-Motion Estimation

We next evaluate complete trajectories containing 331–556 time steps, or 1,986–3,336 images. Since offline global-attention models cannot process these sequences in a single forward pass under the available memory budget, we compare methods capable of completing full-sequence inference.

Table 2: Absolute trajectory error (ATE↓) on the eight held-out long-sequence test trajectories. Each header reports the number of time steps T and the number of synchronized cameras V. Best results are shown in bold.

Method	**S0** $T = 400$ $V = 6$	**S1** $T = 411$ $V = 6$	**S2** $T = 556$ $V = 6$	**S3** $T = 337$ $V = 6$	**S4** $T = 449$ $V = 6$	**S5** $T = 331$ $V = 6$	**S6** $T = 448$ $V = 6$	**S7** $T = 385$ $V = 6$	**Avg.↓**
CUT3R	0.844	3.442	1.816	1.620	2.349	2.329	0.137	0.962	1.687
TTT3R	0.731	3.486	1.589	1.443	2.071	2.357	0.131	0.844	1.582
Point3R	1.783	3.572	2.918	2.064	3.050	4.878	3.354	3.197	3.102
InfiniteVGGT	0.746	3.431	1.615	1.406	2.040	2.404	0.127	0.837	1.576
LingBot-Map	0.941	3.578	2.521	1.921	2.563	5.108	2.991	3.051	2.834
VGGT-Motion	0.086	0.269	0.372	0.146	0.229	0.151	0.196	0.181	0.204
TALO	0.101	0.254	0.472	0.183	0.241	0.143	0.229	0.184	0.226
VGGT-Align	0.112	0.326	0.438	0.221	0.221	0.221	0.257	0.176	0.247
VGGT-Long	0.124	0.361	0.615	0.267	0.348	0.216	0.281	0.224	0.305
MV3R	**0.038**	**0.120**	**0.178**	**0.051**	**0.082**	**0.090**	**0.125**	**0.078**	**0.095**

Table 2 shows that MV3R achieves the lowest ATE on all eight held-out trajectories, averaging 0.095 versus 0.204 for the strongest baseline, VGGT-Motion, a 53.4% reduction.

As illustrated in Figure 4, MV3R remains closely aligned with the ground-truth trajectories, whereas the baselines exhibit larger local offsets and cumulative deformation.

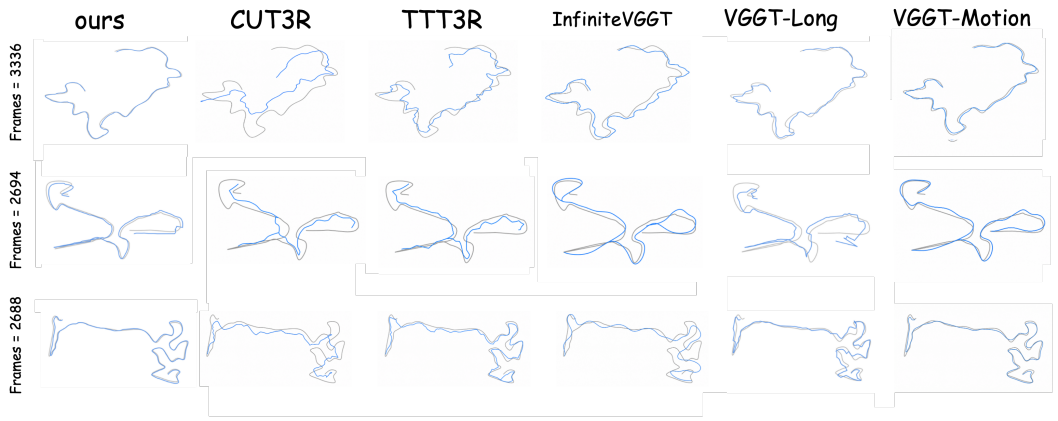

Figure 4: Long-sequence camera trajectories. Gray denotes ground truth and blue denotes predictions.

5.3 Ablation Studies

We evaluate the two main components of MV3R on Replica under settings designed to isolate their respective effects. The array-aware encoding ablation uses $T = 100$ and $V = 6$ to evaluate local multi-view reconstruction, whereas the scene-memory ablation uses $T = 300$ and $V = 6$ to emphasize long-range calibration. Since the two experiments use different sequence lengths, their absolute values are not directly comparable.

Table 3: Ablation studies of the proposed components.

(a) Array Encoding

Setting	Acc.↓	Comp.↓	NC↑	ATE↓	AUC@30↑
w/o	0.074	0.023	0.655	0.099	0.901
w/	**0.052**	**0.014**	**0.694**	**0.076**	**0.931**

(b) Scene Memory

Setting	Acc.↓	Comp.↓	NC↑	ATE↓	AUC@30↑
w/o	0.071	0.018	0.605	0.224	0.749
w/	**0.061**	**0.014**	**0.661**	**0.108**	**0.935**

Array-aware geometric encoding. Removing the Plücker-ray encoding worsens all metrics in Table 3(a). Acc. increases from 0.052 to 0.074 and Comp. from 0.014 to 0.023, while camera-motion estimation also degrades. Thus, fixed-array geometry improves local cross-view consistency.

Scene memory. Without scene memory, inter-window calibration relies only on adjacent-window overlaps. In Table 3(b), scene memory reduces ATE from 0.224 to 0.108 and raises AUC@30 from 0.749 to 0.935, while also improving reconstruction. This confirms that long-term scene context stabilizes global calibration and reduces accumulated drift.

6 Conclusion

We presented MV3R, a window-based streaming model for long-horizon indoor multi-view reconstruction. By combining overlap-feature propagation, fixed-array Plücker-ray encoding, and memory-conditioned $\mathrm{Sim}(3)$ calibration, MV3R scales to thousands of images while maintaining global consistency. We also introduced MV3D, a synchronized six-camera benchmark with 200 trajectories across 138 indoor scenes. Experiments show strong reconstruction and trajectory-estimation gains, particularly in reducing cumulative drift. Future work will improve efficiency and robustness to calibration, synchronization, and dynamic-scene variations.

Reproducibility Statement

The main paper provides the complete definition of MV3R, including array-aware local reconstruction and memory-conditioned global calibration. Detailed training objectives, target construction, and the sequential training strategy are provided in Appendix B. Detailed benchmark construction, trajectory generation, and camera configuration are provided in Appendix A, while evaluation metrics and principal experimental settings are described in Section 5.

Ethics Statement

The benchmark is derived from reconstructed indoor environments distributed under source-specific licenses and usage conditions. Any release of MV3D assets must comply with the licenses and redistribution terms of Replica, Gibson, Matterport3D, and HM3D. Indoor reconstruction can expose sensitive spatial layouts. Released data and models should therefore exclude personal information and clearly document privacy and permitted-use constraints.

AI Use Statement

A large language model was used to assist with translating the original Chinese draft into English and with manuscript organization, language editing, and LaTeX formatting. The authors reviewed and verified all AI-assisted content and take responsibility for the final manuscript, including all scientific claims, equations, citations, experimental results, and conclusions.

References

Dejan Azinović, Ricardo Martin-Brualla, Dan B. Goldman, Matthias Nießner, and Justus Thies. Neural RGB-D Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *International Conference on 3D Vision*, 2017.

Lin-Zhuo Chen, Jian Gao, Yihang Chen, Ka Leong Cheng, Yipengjing Sun, Liangxiao Hu, Nan Xue, Xing Zhu, Yujun Shen, Yao Yao, and Yinghao Xu. Geometric Context Transformer for Streaming 3D Reconstruction. arXiv preprint arXiv:2604.14141, 2026a.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. TTT3R: 3D Reconstruction as Test-Time Training. In *International Conference on Learning Representations*, 2026b.

Zhuoguang Chen, Minghui Qin, Tianyuan Yuan, Zhe Liu, and Hang Zhao. LONG3R: Long Sequence Streaming 3D Reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Kai Deng, Zexin Ti, Jiawei Xu, Jian Yang, and Jin Xie. VGGT-Long: Chunk it, Loop it, Align it—Pushing VGGT's Limits on Kilometer-Scale Long RGB Sequences. arXiv preprint arXiv:2507.16443, 2025.

Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-Time RGB-D Camera Relocalization. In *IEEE International Symposium on Mixed and Augmented Reality*, pp. 173–179, 2013.

Xiaosong Jia, Yanhao Liu, Junqi You, Renqiu Xia, Yu Hong, and Junchi Yan. DriveVGGT: Visual Geometry Transformer for Autonomous Driving. arXiv preprint arXiv:2511.22264, 2025.

Yushi Lan, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Shuai Yang, Bo Dai, Chen Change Loy, and Xingang Pan. STream3R: Scalable Sequential 3D Reconstruction with Causal Transformer. arXiv preprint arXiv:2508.10893, 2025.

Jungho Lee, Minhyeok Lee, Sunghun Yang, Minseok Kang, and Sangyoun Lee. SwiftVGGT: A Scalable Visual Geometry Grounded Transformer for Large-Scale Scenes. arXiv preprint arXiv:2511.18290, 2025.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding Image Matching in 3D with MASt3R. In *European Conference on Computer Vision*, 2024.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth Anything 3: Recovering the Visual Space from Any Views. arXiv preprint arXiv:2511.10647, 2025.

Yuzheng Liu, Siyan Dong, Shuzhe Wang, Yingda Yin, Yanchao Yang, Qingnan Fan, and Baoquan Chen. SLAM3R: Real-Time Dense Scene Reconstruction from Monocular RGB Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16651–16662, 2025.

Si-Yu Lu, Po-Ting Chen, Hui-Che Hsu, Sin-Ye Jhong, Wen-Huang Cheng, and Yung-Yao Chen. OVGGT: $O(1)$ Constant-Cost Streaming Visual Geometry Transformer. arXiv preprint arXiv:2603.05959, 2026.

Manthan Patel, Fan Yang, Yuheng Qiu, Cesar Cadena, Sebastian Scherer, Marco Hutter, and Wenshan Wang. TartanGround: A Large-Scale Dataset for Ground Robot Perception and Navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2025.

Santhosh K. Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alex Clegg, John Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X. Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-Matterport 3D Dataset (HM3D): 1000 Large-Scale 3D Environments for Embodied AI. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The Replica Dataset: A Digital Replica of Indoor Spaces. arXiv preprint arXiv:1906.05797, 2019.

Shinji Umeyama. Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, 1991.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual Geometry Grounded Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3D Perception Model with Persistent State. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jérôme Revaud. DUSt3R: Geometric 3D Vision Made Easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. TartanAir: A Dataset to Push the Limits of Visual SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Scalable Permutation-Equivariant Visual Geometry Learning. arXiv preprint arXiv:2507.13347; Pi3X model release, 2025c.

Yuqi Wu, Wenzhao Zheng, Jie Zhou, and Jiwen Lu. Point3R: Streaming 3D Reconstruction with Explicit Spatial Pointer Memory. arXiv preprint arXiv:2507.02863, 2025.

Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson Env: Real-World Perception for Embodied Agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Zhuang Xiong, Chen Zhang, Qingshan Xu, and Wenbing Tao. VGGT-Motion: Motion-Aware Calibration-Free Monocular SLAM for Long-Range Consistency. arXiv preprint arXiv:2602.05508, 2026.

Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. InfiniteVGGT: Visual Geometry Grounded Transformer for Endless Streams. arXiv preprint arXiv:2601.02281, 2026.

Fengyi Zhang, Tianjun Zhang, Kasra Khosoussi, Zheng Zhang, Zi Huang, and Yadan Luo. TALO: Pushing 3D Vision Foundation Models Towards Globally Consistent Online Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a.

Wei Zhang, Yihang Wu, Songhua Li, and Qi Wang. VGGT-Align: Bridging Local Reconstruction and Global Consistency for Long-Sequence 3D Reconstruction. arXiv preprint arXiv:2608.15260, 2026b.

Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Streaming 4D Visual Geometry Transformer. arXiv preprint arXiv:2507.11539, 2025.

Zichen Zou, Xiaosong Jia, Zuxuan Wu, and Yu-Gang Jiang. Attention Itself Could Retrieve: RetrieveVGGT: Training-Free Long Context Streaming 3D Reconstruction via Query-Key Similarity Retrieval. arXiv preprint arXiv:2605.09644, 2026.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Shengyin Jiang, Long Chen, Zhi-Xin Yang, and Jiwen Lu. DVGT: Driving Visual Geometry Transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Hanbing Li, Long Chen, Zhi-Xin Yang, and Jiwen Lu. DVGT-2: Vision-Geometry-Action Model for Autonomous Driving at Scale. arXiv preprint arXiv:2604.00813, 2026b.

Supplementary Material

MV3R: Long-Sequence Multi-View 3D Reconstruction

In this supplementary material, we provide additional details on benchmark construction and training, together with extra ablation results that support the main paper. The contents are organized as follows:

- **Section A (Benchmark Construction Details).** We describe the scene sources, trajectory generation procedure, camera-array configuration, annotations, and dataset splits of MV3D.
- **Section B (Training Details).** We present the training objectives and the sequential training strategy.
- **Section C (Additional Ablation Studies).** We provide an additional ablation on window length.

A Benchmark Construction Details

A.1 Scene Sources and Trajectory Generation

We render continuous RGB-D sequences with Habitat-Sim (Savva et al., 2019) using scenes from Replica, Gibson, Matterport3D, and HM3D (Straub et al., 2019; Xia et al., 2018; Chang et al., 2017; Ramakrishnan et al., 2021). For each scene, we sample spatially distributed waypoints on the navigation mesh, connect consecutive waypoints with shortest paths, and apply curve smoothing and uniform resampling. This procedure produces continuous trajectories that broadly cover traversable regions while maintaining approximately constant motion speed.

A.2 Camera Array and Annotations

At each time step, a rigid six-camera array captures synchronized observations. All cameras are attached to the same six-degree-of-freedom agent pose with fixed relative extrinsics. One camera faces forward, while the remaining five are distributed uniformly in the horizontal plane. We use a pinhole camera model with a resolution of 1280×960 and a horizontal field of view of 100°. The benchmark provides RGB images, metric depth maps, camera poses represented by translation and quaternion, and camera intrinsics (f_x, f_y, c_x, c_y).

A.3 Dataset Statistics and Splits

MV3D contains 200 continuous trajectories covering 138 indoor scenes. The benchmark includes 192 training trajectories and eight held-out test trajectories. Each test trajectory comes from a distinct scene, with two test trajectories from each source dataset. Training and test data are separated at the scene level to prevent information leakage.

Table 4: Summary of the MV3D benchmark.

Property	Configuration
Scene sources	Replica, Gibson, Matterport3D, HM3D
Scale / split	138 scenes, 200 trajectories; 192 train / 8 test
Camera input	Six synchronized fixed-extrinsic cameras; 1280×960, 100° FoV
Annotations	RGB, metric depth, 6-DoF poses, intrinsics

B Training Details

B.1 Training Objectives

The model is initialized from pretrained VGGT. We train the modified global-attention layers, prediction heads, and the new MV3R modules. Local reconstruction uses the standard VGGT geometric supervision, including point-map, depth, and camera-pose losses.

For each window, a fixed reference view defines the local coordinate system. The ground-truth transformation $\mathbf{S}_k^*$ from the window coordinate system to the global coordinate system is constructed from the ground-truth relative pose between that reference view and the reference view of the first window, with $\mathbf{S}_1^* = \mathbf{I}$.

Although identical patches in adjacent windows have deterministic correspondence, their geometric predictions are not necessarily equally reliable. We therefore supervise correspondence reliability with a soft target based on the consistency of local points transformed into the ground-truth global coordinate system:

$$y_{k,i} = \exp\left(-\alpha_{\text{align}} \left\| \mathbf{S}_{k-1}^* \,\text{sg}\left(\mathbf{p}_{k-1,i}^{\text{ov}}\right) - \mathbf{S}_k^* \,\text{sg}\left(\mathbf{p}_{k,i}^{\text{ov}}\right) \right\|_2\right), \tag{6}$$

where $\text{sg}(\cdot)$ denotes stop-gradient and α_{align} is a temperature coefficient. Correspondences with higher geometric consistency receive larger target weights. Given logits from the weight-prediction head, we use binary cross-entropy over valid correspondences.

For global calibration, we directly supervise the final window transformation after residual correction:

$$\mathcal{L}_{\text{calib}} = \left\| \text{Log}_{\text{Sim}(3)} \left[(\mathbf{S}_k^*)^{-1} \Delta\mathbf{S}_k \mathbf{S}_k^c \right] \right\|_{1,\boldsymbol{\lambda}}, \tag{7}$$

where $\|\cdot\|_{1,\boldsymbol{\lambda}}$ is a weighted L_1 norm over rotation, translation, and log-scale components. Because the weighted Umeyama solver is differentiable, this objective jointly supervises correspondence-reliability prediction and residual transformation correction.

B.2 Sequential Training Strategy

During sequential training, the contextual states and scene memory $\mathbf{M}_{k-1}$ inherited from the previous window are treated as stop-gradient variables, avoiding backpropagation through the complete sequence. We adopt curriculum-based memory updates. Early in training, the scene memory is updated using ground-truth transformations. Predicted transformations are then progressively introduced, and the final stage relies exclusively on predicted transformations. This strategy improves training stability while reducing the discrepancy between training and streaming inference.

C Additional Ablation Studies

C.1 Window Length

We vary the number of time steps per window while holding all other settings fixed.

Figure 5 shows that larger windows generally improve trajectory estimation by providing richer temporal context. Acc. and Comp. are best at 20 time steps, whereas further increasing the window length raises memory consumption and slightly degrades reconstruction. We therefore use 20 time steps in the main experiments.

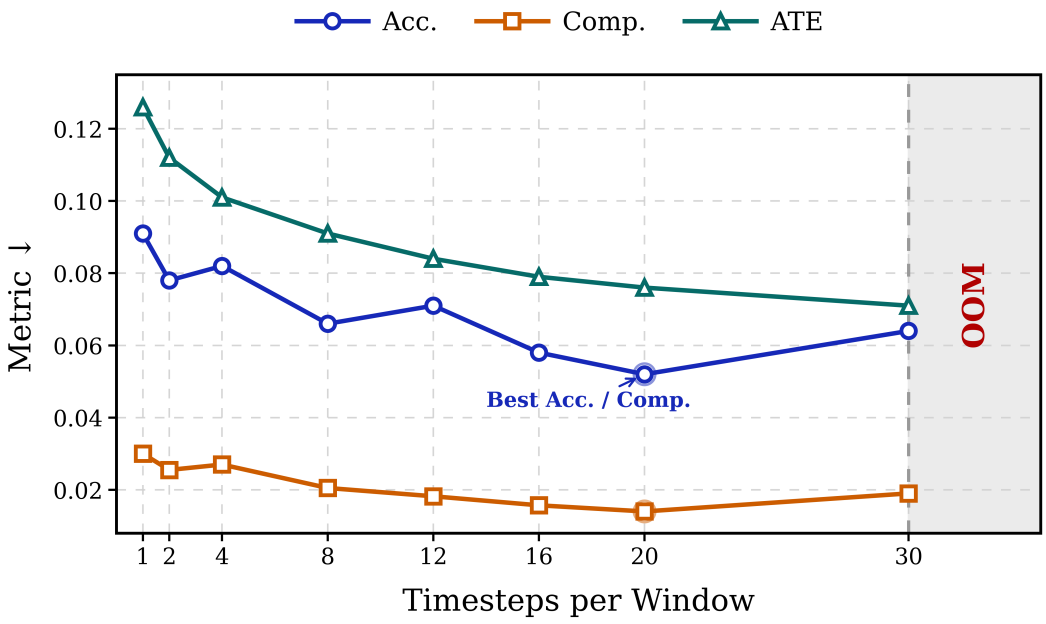

Figure 5: Acc., Comp., and ATE under different window lengths. The horizontal axis shows time steps per window. The shaded region marks settings that exceed available GPU memory.