MEMORY MATTERS: A DIAGNOSTIC MEMORY BENCHMARK FOR ONLINE 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

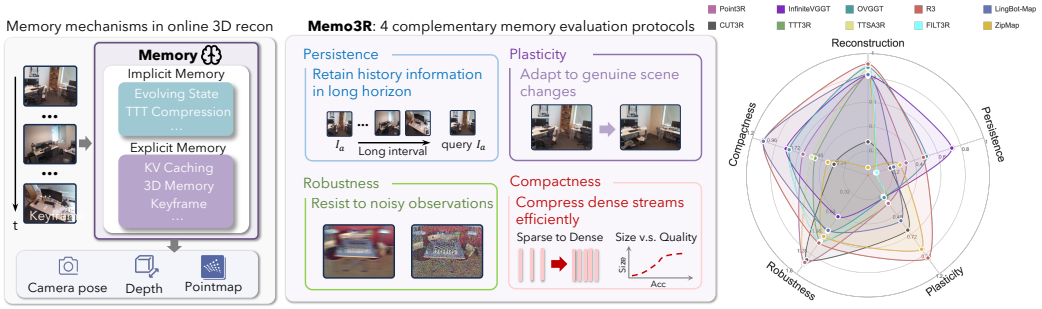


Figure 1: We present Memo3R, a diagnostic benchmark for feed-forward online 3D reconstruction. We revisit representative online 3D reconstruction methods from a memory-centric view, and conceptualize them into four technical categories: evolving state, test-time-training compression, KV caching, 3D memory and keyframes. Memo3R further formulates four fundamental dimensions of memory as the core of its evaluation: persistence, plasticity, robustness, and compactness.

ABSTRACT

Feed-forward online 3D reconstruction is a challenging long-context reasoning problem, where large geometry models rely on memory mechanisms to retain historical information and preserve reconstruction accuracy. However, existing benchmarks mainly measure final reconstruction quality, offering limited insight into what memory retains, when it updates, how it handles unreliable inputs, and how efficiently it compresses scene history. We introduce Memo3R, a diagnostic benchmark that evaluates memory in online 3D reconstruction along four dimensions: persistence, plasticity, robustness, and compactness. For each dimension, we design controlled causal protocols and diagnostic metrics that isolate memory behavior. Benchmarking representative implicit- and explicit-memory methods across indoor and outdoor scenes reveals memory-specific trade-offs beyond conventional reconstruction metrics, providing practical guidance for future long-context 3D reconstruction models. We will release our code and data.

1 INTRODUCTION

Recent large geometry models (Wang et al., 2025a;b; 2026c;a) have advanced feed-forward 3D reconstruction by jointly predicting camera parameters, depth maps, and point maps from visual observations. While they provide efficient alternatives to traditional optimization-based reconstruction pipelines (Schönberger & Frahm, 2016), deploying them in online settings introduces a fundamentally different challenge. Offline reconstruction can jointly reason over a complete set of input views, whereas online reconstruction processes observations sequentially and must update its representation as new frames arrive, without repeatedly revisiting the full observation history. Consequently, online reconstruction is inherently memory-dependent: accurate predictions require not only the current observation, but also geometric, scale, and pose information accumulated from the past.

Recent methods address this challenge by augmenting feed-forward reconstruction models with memory mechanisms. As illustrated in Fig. 1 (left), these mechanisms can be broadly organized into two implementation categories: *Implicit memories* encode historical information in evolving

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.

latent states (Wang et al., 2025b; Chen et al., 2026b; Zheng et al., 2026; Jin & Ye, 2026) or model parameters adapted through test-time training (TTT) compression (Jin et al., 2026; Zhang et al., 2026). *Explicit memories* retain accessible historical information, such as keyframes (Xu et al., 2026), key-value tokens from previous observations for future retrieval (Zhuo et al., 2026; Yushi et al., 2026; Yuan et al., 2026; Lu et al., 2026), and spatial representations including point clouds, voxels, or spatial pointers (Wang & Agapito, 2025; Wu et al., 2025; Wang & Agapito, 2026). Despite their architectural differences, all aim to preserve useful information from past observations for future geometric prediction. This raises a common question that existing evaluations do not directly answer: *how well does the memory itself work?*

Memory has long been studied as a mechanism for retaining and retrieving information over extended contexts, including memory-augmented neural networks (Weston et al., 2014; Graves et al., 2014; 2016), continual learning (Kirkpatrick et al., 2017; De Lange et al., 2021), object permanence (Shamsian et al., 2020; Tokmakov et al., 2021; Plizzari et al., 2025; Patraucean et al., 2023), and long-horizon reasoning benchmarks (Grauman et al., 2022; Khanna et al., 2024; Hu et al., 2025). Online 3D reconstruction is a geometric instance of this broader problem: memory must preserve metric geometry, camera context, and spatial consistency over causal streams. However, existing 3D reconstruction benchmarks primarily evaluate task-level outputs, such as camera pose estimation (Sturm et al., 2012; Dai et al., 2017; Zhou et al., 2018; Reizenstein et al., 2021), video depth prediction (Palazzolo et al., 2019; Geiger et al., 2013; Bozic et al., 2021), and point-map estimation (Shotton et al., 2013; Azinović et al., 2022; Yuan et al., 2026; Zhang et al., 2026). Such metrics reveal whether a reconstruction is accurate, but provide limited insight into *why* an online model succeeds or fails as a memory system: what information survives over long horizons, whether valid new evidence can be incorporated without erasing useful history, how unreliable observations affect the accumulated state, and how memory requirements scale with sequence length. Moreover, existing evaluations often permit chunk-wise processing, post-hoc refinement, or state resets that improve final reconstruction quality but confound the assessment of memory. These practices may expose earlier predictions to later observations, mix memory performance with additional optimization, or shorten the effective retention horizon. Therefore, the resulting performance may not faithfully reflect the model's ability to maintain and use its accumulated state during continuous, causal inference.

In this paper, we present Memo3R, the first systematic benchmark for studying online 3D reconstruction through the lens of memory. Rather than treating memory as an implementation detail, we evaluate it as a core capability that determines whether an online model can preserve long-range scene information, adapt to newly observed geometry, respond to unreliable inputs, and remain computationally efficient. To this end, as shown in Fig. 1 (middle), we formulate four fundamental benchmarking dimensions for memory in online 3D reconstruction: **(i) persistence**, retaining important historical information over long sequences; **(ii) plasticity**, adapting to genuine scene changes or newly observed geometry; **(iii) robustness**, resisting noisy, ambiguous, or unreliable inputs; and **(iv) compactness**, compressing accumulated observations into an efficient memory representation. For each dimension, we develop controlled protocols and diagnostic measures that isolate the corresponding memory behavior under a unified causal evaluation setting.

Rather than collecting a new dataset, Memo3R reorganizes existing 3D reconstruction datasets into targeted online evaluation protocols, allowing diverse memory architectures to be compared under matched observation histories and resource constraints. These protocols reveal strengths and limitations of representative memory methods beyond conventional reconstruction accuracy. Explicit memories, including KV caching, spatial memory and keyframe-based memory, generally provide stronger long-term retention and corruption resistance through directly accessible historical storage, whereas evolving-state and TTT-compression methods achieve more compact memory representations but are more vulnerable to long-horizon forgetting and unreliable updates (see Fig. 1 (right)). Moreover, increasing memory capacity strengthens retention but does not consistently improve adaptation to scene changes. Memo3R therefore complements conventional reconstruction metrics with a reusable framework for diagnosing how online 3D reconstruction systems remember, update, resist corruption, and compress their observation history.

2 Related Work

We focus here on feed-forward online 3D reconstruction and their memory mechanisms. Additional background on general reconstruction pipelines and offline methods are discussed in Supp. Sec. B

Memory in Online 3D Reconstruction. To tackle long-sequence reconstruction, online feedforward models introduce memory mechanisms to retain and propagate scene information over streaming inputs. We group existing designs into two broad categories: implicit memories, including evolving state and TTT compression; and explicit memories, including keyframes, KV caching, and 3D memory. Evolving state methods (Wang et al., 2025b; Chen et al., 2026b; Zheng et al., 2026; Jin & Ye, 2026) maintain persistent latent states across frames; TTT compression methods (Jin et al., 2026; Zhang et al., 2026) store history information in test-time optimizable fast weights; keyframe methods (Xu et al., 2026; Chen et al., 2026a) retain important keyframes; KV caching methods (Zhuo et al., 2026; Yushi et al., 2026; Yuan et al., 2026; Cheng et al., 2026) reuse cached tokens from previous observations; and 3D memory methods (Wang & Agapito, 2025; Wu et al., 2025; Chen et al., 2025) maintain explicit spatial structures. These methods all aim to summarize past observations for future reconstruction, but may exhibit different trade-offs in retention, adaptation, robustness, and efficiency. We provide a more detailed discussion of memory types in Sec. 3.

Memory-centric Benchmarking. Memory-centric benchmarks evaluate whether models can retain, retrieve, and update information over long contexts. Prior work has studied such capabilities in long-horizon reasoning (Zhang et al., 2025b; Goletto et al., 2024), continual learning (De Lange et al., 2021; Tang et al., 2024), embodied perception (Hu et al., 2025; Yang et al., 2025) and object permanence (Shamsian et al., 2020; Tokmakov et al., 2021; Plizzari et al., 2025; Patraucean et al., 2023). For example, 3DLLM-Mem (Hu et al., 2025) evaluates long-term spatial-temporal memory for embodied 3D reasoning, while ViLCo-Bench (Tang et al., 2024) studies episodic memory in continual video-language tasks. In contrast, online 3D reconstruction requires memory to preserve metric geometry, camera context, and spatial consistency under causal streaming. Existing reconstruction evaluations mainly measure final output quality and provide limited diagnostic insight into how memory retains, updates, resists corruption, or compresses scene history. Memo3R addresses this gap by introducing memory-targeted diagnostic protocols for feed-forward online 3D reconstruction.

3 Benchmarking Setup

This section defines the common causal evaluation setting used throughout Memo3R. We first formulate online reconstruction with memory, then categorize the memory implementations evaluated in our benchmark, and finally describe the datasets, metrics, and model selection criteria.

Formulation. Given an image stream $\mathcal{I}_{1:T} = \{I_1, \ldots, I_T\}$, online 3D reconstruction aims to causally estimate scene geometry and camera trajectory as observations arrive. At each time step t, the model receives the current image I_t together with a memory state $\mathcal{M}_{t-1}$ that summarizes previous observations. It then predicts camera parameters $\hat{\Pi}_t$, a depth map $\hat{D}_t$, and a point map $\hat{P}_t$, while updating its memory:

$$(\hat{Y}_t, \mathcal{M}_t) = g_\theta(I_t, \mathcal{M}_{t-1}), \tag{1}$$

where $\hat{Y}_t = \{\hat{\Pi}_t, \hat{D}_t, \hat{P}_t\}$ denotes the predicted 3D scene representation at time t, and g_θ is the streaming 3D reconstruction model parameterized by θ. The memory state $\mathcal{M}_t$ abstracts any information carried forward from previous observations, regardless of its implementation. It may correspond to explicit spatial structures, cached tokens, latent hidden states, or adapted model parameters.

Memory Types. We group existing methods into two broad categories according to how they represent historical information: explicit and implicit memory.

Explicit Memory Methods retain historical observations or features as individually accessible entries. These include *keyframe-based methods* (Xu et al., 2026), which store selected past frames and their associated features; *KV caching methods* (Yushi et al., 2026; Zhuo et al., 2026; Yuan et al., 2026; Lu et al., 2026), which cache keys and values from previous image tokens for subsequent attention; and *3D memory methods* (Wu et al., 2025; Chen et al., 2025; Wang et al., 2026b; Wang & Agapito, 2025), which organize historical information in spatial structures, such as voxels or spatial pointers.

These representations support direct retrieval of historical visual or geometric context, but require selection, fusion, pruning, or eviction strategies to reduce redundancy and control memory growth.

Implicit Memory Methods compress historical information into a persistent latent state or model parameters. *Evolving state methods* (Wang et al., 2025b; Chen et al., 2026b; Zheng et al., 2026; Jin & Ye, 2026) retrieve and update a latent state as new observations arrive, while *test-time training compression methods* (Jin et al., 2026) encode sequence-specific information in a small set of fast weights updated online. Both approaches provide compact memory representations that naturally support causal processing, although individual historical observations are no longer directly accessible.

Evaluated Methods. We evaluate representative methods with publicly available implementations across two memory categories. *Implicit memories* include evolving-state methods CUT3R (Wang et al., 2025b), TTT3R (Chen et al., 2026b), TTSA3R (Zheng et al., 2026), and FILT3R (Jin & Ye, 2026), and online version of ZipMap (Jin et al., 2026) with TTT compression. *Explicit memories* include Point3R (Wu et al., 2025) with pointer-based 3D memory, InfiniteVGGT (Yuan et al., 2026) and OVGGT (Lu et al., 2026) with KV caching, and R^3 (Xu et al., 2026) and LingBot-Map (Chen et al., 2026a) with keyframe-based memory. Memory components are detailed in Supp. Table A1. All methods are evaluated under a strictly causal online protocol without chunk-wise processing, state resets, or post-processing. Because the evaluated methods may differ in training data, sequence length, and optimization settings, their absolute reconstruction quality is not directly comparable in all cases. We therefore focus primarily on **relative performance changes** under each diagnostic protocol, while also reporting absolute reconstruction metrics for completeness.

Datasets. Following prior work (Chen et al., 2026b; Xie et al., 2026; Jin & Ye, 2026), we build our benchmark on three 3D reconstruction datasets: 7-Scenes (Shotton et al., 2013) for static indoor scenes with varied camera trajectories; Bonn (Palazzolo et al., 2019) for dynamic indoor scenes; and Virtual KITTI 2 (VKITTI2) (Cabon et al., 2020) for large-scale dynamic outdoor scenes with forward-driving motion. We sample up to 1000 frames from each sequence using a stride of 1.

Base Metrics. Following previous work (Wang et al., 2025b; Chen et al., 2026b; Jin & Ye, 2026), we use Chamfer Distance (CD) as the metric for geometric accuracy. CD is computed as the average of accuracy and completeness: accuracy is the nearest-neighbor Euclidean distance from each reconstructed point to the ground truth, while completeness is computed in the reverse direction.

Although our main experiments use 3D reconstruction datasets, Memo3R is designed to be dataset- and representation-agnostic. Its diagnostic protocols operate on causal image streams and can be applied to other datasets and outputs, such as depth maps and camera trajectories, whenever suitable ground truth or evaluation metrics are available.

4 MEMO3R: A MEMORY-CENTRIC DIAGNOSTIC BENCHMARK

Memo3R evaluates memory for feed-forward online 3D reconstruction along four axes: persistence, plasticity, robustness, and compactness. Each diagnostic is designed to isolate one memory capability under a strictly causal protocol.

4.1 PERSISTENCE

Persistence measures the ability of an online reconstruction system to retain previously observed static scene content while continuing to process new frames. Unlike short-term temporal smoothing, which only requires stability across adjacent frames, persistence requires memory to preserve geometry, scale, and pose context over long intervals, even after the corresponding region has left the current field of view. We evaluate this property with a temporal retention test.

Objective. The test measures whether information from an anchor frame remains recoverable after the model has processed later observations. A persistent memory should allow the model to reconstruct the geometry visible in the anchor frame with similar quality even as the retention interval increases; forgetting appears as degraded reconstruction accuracy over longer intervals.

Protocol. Let I_a denote an anchor frame, and let R_a denote the visible scene region with available ground-truth geometry at time a. As shown in Fig. 2 (Left), we first run the model causally on the

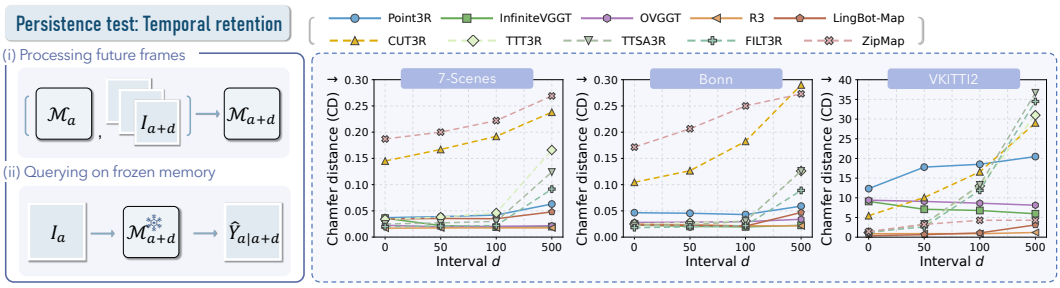


Figure 2: Left: workflow of persistence test. The memory state $\mathcal{M}_a$ at anchor frame I_a is updated with future frames, yielding $\mathcal{M}_{a+d}$. $\mathcal{M}_{a+d}$ is then frozen and queried using anchor frame I_a to evaluate whether anchor-frame information is retained over interval d. Right: persistence results.

stream until time $a+d$, where d is the retention interval. We then freeze the resulting memory state, denoted by $\mathcal{M}^*_{a+d}$. Finally, we query the model with the anchor image, $\hat{Y}_{a|a+d} = g_\theta(I_a, \mathcal{M}^*_{a+d})$, and evaluate the predicted geometry against the ground-truth point cloud of R_a.

Key Variables. We vary $d \in \{50, 100, 500\}$ to measure how reconstruction quality changes as the model processes increasingly many frames after the anchor. A persistent memory should produce stable depth maps, point maps, and camera estimates as d increases, whereas a memory that forgets or is overwritten by later frames should show increasing reconstruction error.

Results. Fig. 2 (right) reports the temporal retention curves on 7-Scenes, Bonn, and Virtual KITTI 2. For each dataset, we evaluate on anchor frames shared across all intervals. Qualitative examples are provided in the supplementary material.

Implicit memories suffer from long-horizon forgetting. Across all three datasets, evolving-state methods exhibit increasing reconstruction error as the retention interval grows: On 7-Scenes and Bonn, TTT3R, TTSA3R, and FILT3R maintain relatively low errors at $d \leq 100$, but degrade substantially at $d = 500$. On VKITTI2, their initial advantage over CUT3R disappears at $d = 500$, suggesting that fixed-size compressed states remain vulnerable to long-horizon information loss. ZipMap also degrades on all datasets, but exhibits a different pattern on VKITTI2: its reconstruction error rises sharply from $d = 0$ to $d = 50$, then largely stabilizes after $d = 100$. We hypothesize that the temporally smooth trajectories and relatively regular outdoor geometry in VKITTI2 better suit its TTT-based compression: following initial adaptation, subsequent observations provide relatively consistent structural information and cause limited additional interference. In contrast, the fine-grained geometry and greater viewpoint variation in indoor scenes may impose more competing updates on the same compact state, resulting in cumulative degradation.

Explicit memories provide stronger temporal retention. In contrast, the evaluated explicit-memory methods generally exhibit significantly more stable performance across different retention intervals. These methods retain historical information in directly accessible entries: Point3R uses spatial pointers, InfiniteVGGT and OVGGT cache KV features, LingBot-Map and R^3 retain keyframe-based memory. Such representations allow selective memory updates based on spatial or semantic proximity, without necessarily rewriting a shared compressed state, and potentially reducing interference with previously stored information. Interestingly, InfiniteVGGT, OVGGT and R^3 even improve with longer retention intervals on 7-Scenes, Virtual KITTI 2 and Bonn respectively, suggesting that the selectively accumulated context can support more accurate reconstruction when querying previously observed regions, similar to an in-context learning process.

4.2 Plasticity

Plasticity measures whether memory can adapt when the scene itself changes. It is complementary to persistence: a useful memory should retain stable geometry over long horizons, but it should not preserve outdated content once later observations provide consistent evidence of a scene change. In online 3D reconstruction, plasticity therefore requires selective updates: the memory should revise changed regions while preserving unchanged geometry, scale, and camera context.

Objective. We evaluate plasticity with controlled object-change test. The goal is to measure whether a model can update outdated geometry after objects are added, while preserving the surrounding

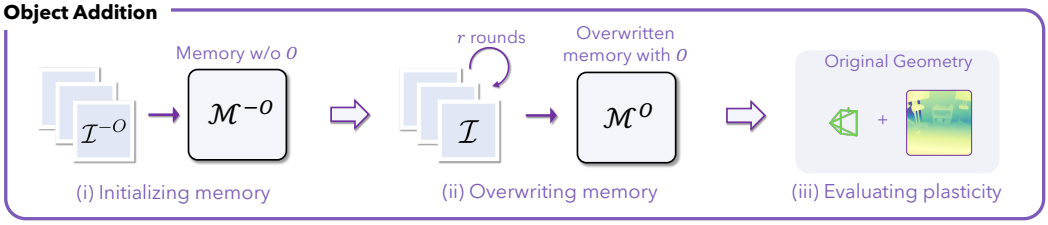


Figure 3: Workflow of the plasticity test. The memory is initialized from the object-removed sequence $\mathcal{I}^{-O}$, then updated with the clean sequence $\mathcal{I}$ containing object O for r rounds. Final-round predictions are evaluated against the original scene geometry.

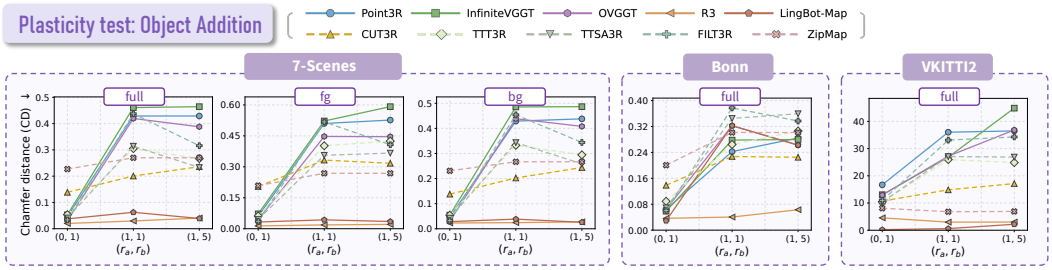


Figure 4: Results of plasticity test under three datasets. fg: changed foreground. bg: unchanged background.

static scene. This setting exposes two failure modes: overly rigid memories continue to reconstruct stale geometry, whereas overly plastic memories update too aggressively and corrupt unchanged regions.

Protocol. Given a clean sequence $\mathcal{I}$ in which an object O is visible, we construct an edited sequence $\mathcal{I}^{-O}$ by removing O and inpainting the exposed background, while leaving the rest of the image unchanged. We construct each evaluation stream in two stages. First, the model processes the edited sequence $\mathcal{I}^{-O}$ for r_a repeated passes, which initializes or reinforces the memory with the edited scene state. Then, the model processes the original sequence $\mathcal{I}$ for r_b repeated passes, which tests whether the memory can adapt to the changed scene. Formally, the resulting stream is

$$\mathcal{S}_{\mathcal{I}^{-O}\to\mathcal{I},r} = (\mathcal{I}^{-O})^{r_a} \oplus \mathrm{rev}(\mathcal{I})^{r_b},$$

where $\oplus$ denotes sequence concatenation, $\mathrm{rev}(\cdot)$ denotes sequence reversal, and $r = (r_a, r_b)$ specifies the number of repeated passes over the edited and original sequences. We reverse the original sequence so that the camera trajectory remains temporally continuous at the transition between $\mathcal{I}^{-O}$ and $\mathcal{I}$. The stream $\mathcal{S}_{\mathcal{I}^{-O}\to\mathcal{I},r}$ therefore evaluates object addition. Fig. 3 depicts the workflow of object-addition evaluation. We evaluate predictions from the final round of the original sequence $\mathcal{I}$, ensuring that the evaluation frames are free of synthesis artifacts. The detailed data generation procedure and quality analyses are provided in Supp. Sec. D. We also provide additional object-removal evaluation in Supp. Sec. E.

The repeated-pass design tests two aspects of plasticity. Increasing r_b measures whether memory can progressively incorporate valid post-change evidence over repeated observations. Increasing r_a measures memory inertia, i.e., whether repeatedly reinforcing the pre-change scene makes the memory harder to update after the scene changes.

Metrics. Given the object-changed region R_O and the unchanged background region R_B, we report reconstruction quality on the last round of the original sequence, both for the full scene and separately for R_O and R_B. Errors in R_O measure residual stale geometry after object addition, while error on R_B measures collateral damage to unchanged scene content. We also report adaptation curves over different rounds of original sequences. The number of rounds required for the error on R_O to stabilize measures adaptation speed, while the error on R_B indicates whether the update remains spatially selective.

Key Variables. We vary the number of repeated rounds $(r_a, r_b) \in \{(0,1),(1,1),(1,5)\}$, which respectively evaluates (i) single-pass reconstruction performance on the original sequence without prior memory accumulation; (ii) whether the memory can react to true scene changes after observing the edited sequence once; (iii) update capacity: an effective plastic memory should reduce error

in the changed region R_O as r_b increases, while maintaining stable reconstruction quality in the unchanged background R_B. Together, these variations diagnose whether memory can adapt to valid new evidence without globally disrupting stable scene content.

Results. Fig. 4 shows the 3D reconstruction error in different repetition settings. An effectively plastic memory should exhibit only minor degradation from $(0, 1)$ to $(1, 1)$, indicating that outdated information can be overwritten after the scene change, and should achieve better performance under $(1, 5)$ as repeated observations of the changed scene provide stronger evidence for updating.

Compressed memories adapt more readily to genuine scene changes. Overall, compressed memories, including evolving state and TTT compression, exhibit stronger plasticity to scene changes. As r_b increases, these methods more effectively incorporate the post-change observations, reducing residual errors in the changed region R_O while better preserving reconstruction quality in the unchanged background R_B. In contrast, explicit memories such as KV caching and 3D memory tend to preserve previously stored scene information more directly. While this improves persistence, it also makes outdated content harder to overwrite after a true scene change. As a result, these methods are more likely to retain residual geometry in the changed region R_O or require more repeated post-change evidence before adapting.

Selective state updates do not improve plasticity. We find that though TTT3R and TTSA3R improve persistence over CUT3R via more selective weight update strategies, they exhibit weaker plasticity than the original CUT3R model. One possible reason is that evolving-state methods often use bounded update gates, which help suppress unreliable observations but may also limit the update magnitude when strong evidence of a genuine scene change appears. This suggests a tension between filtering harmful updates and rapidly incorporating valid changes. TTT-compression methods could in principle address this tension through adaptive test-time learning rates. However, ZipMap only shows superior plasticity on VKITTI2, possibly due to its tuning data distribution and limited training sequence length.

Our memory-budget analysis in Supp. Sec. C further shows the trade-off of persistence and plasticity: increasing explicit-memory capacity strengthens retention, while plasticity saturates or degrades, suggesting that when memory capacity is abundant, plasticity becomes the main bottleneck.

4.3 Robustness

Robustness measures whether memory remains stable when the input stream contains unreliable observations. Unlike plasticity, where later frames provide valid evidence of a scene change, corruptions should not substantially alter the stored scene representation. A robust memory should therefore preserve clean historical geometry while filtering, down-weighting, or otherwise resisting low-quality updates.

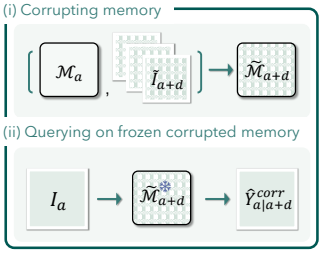


Figure 5: Workflow of robustness test. Memory $\mathcal{M}_a$, from clean frames through the anchor, is updated with corrupted future frames, then frozen and queried with the clean anchor to measure degradation.

Objective. We test whether a memory initialized from clean observations can survive a later corrupted segment. The diagnostic asks how much previously stored geometry is lost when the model continues to update from frames affected by noise, blur, exposure shifts, or rolling-shutter artifacts.

Protocol. This protocol follows the temporal retention test in Sec. 4.1, but perturbs the frames processed after the anchor frame I_a, which are denoted as $\tilde{\mathcal{I}}_{(a+1):(a+d)}$. As illustrated in Fig. 5, we first process clean frames up to the anchor time a to obtain a clean memory state $\mathcal{M}_a$. We then continue the stream with corrupted frames $\tilde{I}_{a+1}, \ldots, \tilde{I}_{a+d}$ and obtain a corrupted memory $\tilde{\mathcal{M}}_{a+d}$. After the corrupted segment, we freeze the memory and query the original clean anchor frame:

$$\hat{Y}^{\text{corr}}_{a|a+d} = g_\theta(I_a, \tilde{\mathcal{M}}^*_{a+d}).$$

As a clean-memory reference, we also query the anchor frame using the memory state before corruption:

$$\hat{Y}^{\text{clean}}_{a} = g_\theta(I_a, \mathcal{M}^*_a).$$

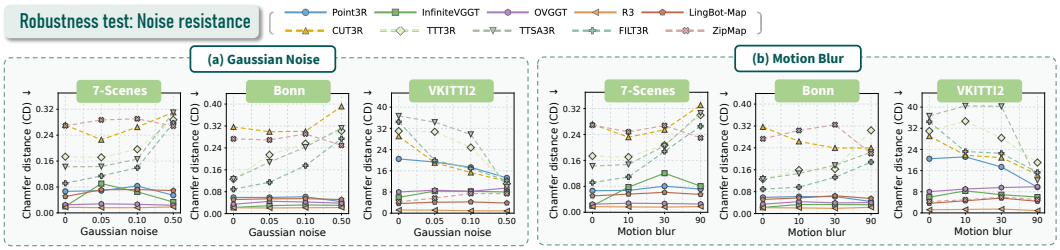


Figure 6: Results of robustness test under increasing Gaussian noise and motion blur.

The difference between these two reconstructions measures how much the corrupted segment has damaged the memory of the anchor region.

Metrics. We evaluate both $\hat{Y}_a^{\text{clean}}$ and $\hat{Y}_{a|a+d}^{\text{corr}}$ against the ground-truth geometry visible in the anchor frame. Let $E(\cdot)$ denote a reconstruction error metric such as Chamfer Distance. A smaller gap between $E(\hat{Y}_a^{\text{corr}})$ and $E(\hat{Y}_a^{\text{clean}})$ indicates that useful historical information remains recoverable despite unreliable inputs, while a larger gap indicates that corrupted observations overwrite or degrade the stored memory.

Key Variables. We vary the corruption type and severity. In our experiments, the corrupted interval length is fixed to 500 frames before revisiting the anchor frame. Additional implementation details are provided in Supp. Sec. D.

Results. Fig. 6 reports the robustness results under Gaussian noise with increasing noise levels on the left, and motion blur with increasing blur severity on the right. Additional results under lighting change are provided in Supp. Sec. E

Explicit memories are more robust to corrupted observations. Explicit-memory methods generally show stronger robustness, particularly on the indoor datasets, which demonstrate that retaining accessible historical context can help preserve reconstruction quality despite corrupted inputs. Spatial 3D memory can limit interference through spatially localized updates, while KV-caching and keyframe methods employ selective update strategies to preserve informative cached context that remains useful even when later inputs are degraded.

Selective state updates reduce, but do not eliminate, corruption sensitivity. Among evolving state methods, TTT3R, TTSA3R and FILT3R generally improve over CUT3R, especially under mild and moderate corruptions, indicating that more selective or adaptive state update strategies can slow down the corruption process with lower updating weights. However, their performance still degrades under severe corruption. This is because each update in evolving state always modifies a compact global state. Once corrupted observations are absorbed into the state, the damage can affect reconstruction globally rather than being localized to specific memory entries. On VKITTI2, evolving-state methods perform better at the highest Gaussian noise level than under clean updates. One possible explanation is that severe corruption weakens subsequent updates, reducing interference with the stored anchor representation.

4.4 Compactness

Compactness measures whether memory can effectively compress an input stream while preserving reconstruction-relevant scene information. We evaluate compactness using information-density test.

Objective. The information-density test compares streams that depict the same scene but contain different levels of redundancy. The goal is to measure how reconstruction quality on a fixed set of evaluation frames changes as the model processes denser or sparser streams. A compact memory should benefit from denser observations, or at least preserve reconstruction accuracy, without being degraded by redundant inputs or requiring unbounded memory growth.

Protocol. As shown in Fig. 7 (left), given a full sequence $\mathcal{I}_{1:T}$ depicting a scene, we construct evaluation streams by subsampling the original sequence with different temporal strides: $\mathcal{I}^{(s)} = \{I_1, I_{1+s}, I_{1+2s}, \ldots\}$, where s denotes the sampling stride. Smaller strides produce denser streams with more overlapping observations and higher redundancy, while larger strides produce sparser streams with fewer observations and lower redundancy.

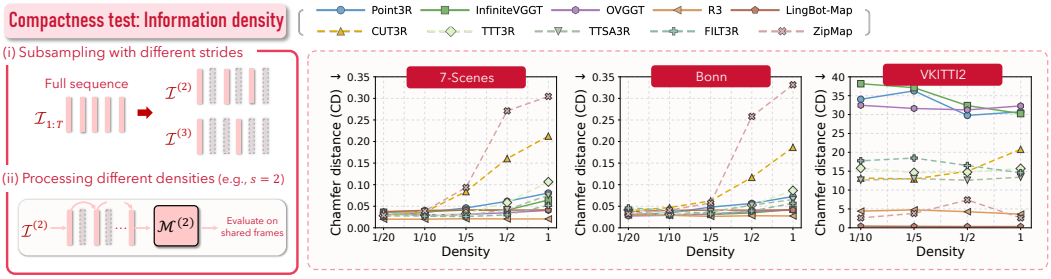


Figure 7: Left: workflow of compactness test. We construct evaluation sequences with different information densities. Right: results of compactness tests with different information densities.

Metrics. We evaluate reconstruction only on frames shared by all subsampled sequences within each dataset, and report reconstruction metric on the shared evaluation set.

Key Variables. We use $s \in \{1, 2, 5, 10, 20\}$ for 7-Scenes and Bonn, and $s \in \{1, 2, 5, 10\}$ for VKITTI2. As s decreases, the input stream becomes denser and contains more redundant observations. A compact memory should maintain reconstruction accuracy under these denser streams, indicating that it can absorb additional information without being overwhelmed by redundant observations.

Results. In Fig. 7 (right), we report the information density curves. As input density increases, a compact memory should degrade less from the redundant information, *i.e.*, the slope over $\frac{1}{s}$ is more gradual.

Compactness does not directly follow from persistence. Although TTSA3R and FILT3R are less persistent than Point3R and InfiniteVGGT in the temporal retention test (Sec. 4.1), they show competitive compactness under denser input streams, illustrating that fixed-size evolving states are effective at absorbing redundant observations into a compact representation. Conversely, R^3 performs well in both diagnostics, and can even benefit from denser observations on Bonn and VKITTI2. A plausible explanation is that denser sampling provides more candidates for keyframe selection, while selective retention limits redundant memory entries. These results show that retaining historical information and compressing redundant observations are distinct capabilities: a method can handle dense streams effectively while vulnerable to long-horizon forgetting.

Compactness test reveals a trade-off between compression and retention. Evolving-state and TTT-compression methods are compact by construction because they utilize fixed-size states or fast weights, but this compression can lose details needed for long-horizon persistence. In contrast, 3D-memory and KV-caching methods preserve richer historical information, which often improves persistence and robustness, but they require explicit mechanisms such as spatial fusion, pointer merging, token pruning, or cache selection to control redundancy and memory growth. The consistent performances of R^3 and LingBot-Map over increasing density suggest that selective memory management can filter out the redundant observations successfully. These results suggest that compactness should not be assessed by memory size or final reconstruction accuracy alone, but by how effectively a memory retains useful information as input redundancy increases.

5 Conclusion

In this work, we introduce Memo3R, a diagnostic benchmark for evaluating memory in feed-forward online 3D reconstruction. Rather than treating memory as an implementation detail or evaluating models only by final reconstruction quality, Memo3R studies memory as a core capability through four complementary dimensions: persistence, plasticity, robustness, and compactness. For each dimension, we design controlled online protocols that isolate a specific memory behavior under a strictly causal setting. Our experiments reveal that existing online reconstruction models exhibit substantially different memory behaviors beyond what conventional reconstruction metrics can capture. By making memory behavior measurable across multiple axes, Memo3R provides a reusable framework for diagnosing current memory mechanisms and guiding the development of future online 3D reconstruction models.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing and assist in writing code for repetitive tasks. We reviewed all AI-assisted work, revising the suggested text for clarity and accuracy, and verifying that the generated code performed the intended operations correctly. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Memo3R may have positive societal impact by supporting the development of more reliable online 3D reconstruction systems for robotics, embodied AI, AR/VR, and spatial perception. By diagnosing memory retention, adaptation, robustness, and compactness, the benchmark can help identify failure modes that may otherwise be hidden by conventional reconstruction metrics. Potential negative impacts are indirect, but improved online 3D reconstruction could be misused for privacy-invasive spatial mapping, surveillance, or tracking. In addition, unreliable reconstructions may pose safety risks if deployed in downstream robotic or embodied systems. Memo3R is a benchmark rather than a deployed system, but these risks should be considered when using it to guide future reconstruction models.

REFERENCES

Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. 2022.

Aljaz Bozic, Pablo Palafox, Justus Thies, Angela Dai, and Matthias Nießner. Transformerfusion: Monocular rgb scene reconstruction using transformers. *NeurIPS*, 2021.

Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020.

Lin-Zhuo Chen, Jian Gao, Yihang Chen, Ka Leong Cheng, Yipengjing Sun, Liangxiao Hu, Nan Xue, Xing Zhu, Yujun Shen, Yao Yao, et al. 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 *ICLR*, 2026b.

Zhuoguang Chen, Minghui Qin, Tianyuan Yuan, Zhe Liu, and Hang Zhao. Long3r: Long sequence streaming 3d reconstruction. In *ICCV*, 2025.

Chong Cheng, Xianda Chen, Tao Xie, Wei Yin, Weiqiang Ren, Qian Zhang, Xiaoyang Guo, and Hao Wang. Longstream: Long-sequence streaming autoregressive visual geometry. *arXiv preprint arXiv:2602.13172*, 2026.

Dongkyu Cho, Taesup Moon, Rumi Chunara, Kyunghyun Cho, and Sungmin Cha. Forget forgetting: Continual learning in a world of abundant memory. In *ICLR*, 2026.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. 44(7):3366–3385, 2021.

Sven Elflein, Ruilong Li, Sérgio Agostinho, Zan Gojcic, Laura Leal-Taixé, Qunjie Zhou, and Aljosa Osep. Vgg-t^3: Offline feed-forward 3d reconstruction at scale. 2026.

Yang Fu, Yike Zheng, Ziyun Dai, and Henghui Ding. Effecterase: Joint video object removal and insertion for high-quality effect erasing. 2026.

Yasutaka Furukawa and Carlos Hernández. Multi-view stereo: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The international journal of robotics research*, 32(11):1231–1237, 2013.

Gabriele Goletto, Tushar Nagarajan, Giuseppe Averta, and Dima Damen. Amego: Active memory from long egocentric videos. In *ECCV*, 2024.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. 2022.

Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538 (7626):471–476, 2016.

Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model. 2025.

Haian Jin, Rundi Wu, Tianyuan Zhang, Ruiqi Gao, Jonathan T Barron, Noah Snavely, and Aleksander Holynski. Zipmap: Linear-time stateful 3d reconstruction with test-time training. 2026.

Seonghyun Jin and Jong Chul Ye. Filt3r: Latent state adaptive kalman filter for streaming 3d reconstruction. *arXiv preprint arXiv:2603.18493*, 2026.

Bingxin Ke, Qunjie Zhou, Jiahui Huang, Xuanchi Ren, Tianchang Shen, Konrad Schindler, Laura Leal-Taixé, and Shengyu Huang. Depth completion as parameter-efficient test-time adaptation. *arXiv preprint arXiv:2602.14751*, 2026.

Nikhil Varma Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. In *3DV*, 2026.

Mukul Khanna, Ram Ramrakhya, Gunjan Chhablani, Sriram Yenamandra, Theophile Gervet, Matthew Chang, Zsolt Kira, Devendra Singh Chaplot, Dhruv Batra, and Roozbeh Mottaghi. Goat-bench: A benchmark for multi-modal lifelong navigation. 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwińska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. pp. 1611–1621, 2021.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024.

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. In *ECCV*, 2026.

Emanuele Palazzolo, Jens Behley, Philipp Lottes, Philippe Giguere, and Cyrill Stachniss. Refusion: 3d reconstruction in dynamic environments for rgb-d cameras exploiting residuals. In *IROS*, 2019.

Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception test: A diagnostic benchmark for multimodal video models. 2023.

Chiara Plizzari, Shubham Goel, Toby Perrett, Jacob Chalk, Angjoo Kanazawa, and Dima Damen. Spatial cognition from egocentric video: Out of sight, not out of mind. In *3DV*, 2025.

Marc Pollefeys, Reinhard Koch, and Luc Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. 32(1):7–25, 1999.

Marc Pollefeys, Luc Van Gool, Maarten Vergauwen, Frank Verbiest, Kurt Cornelis, Jan Tops, and Reinhard Koch. Visual modeling with a hand-held camera. 59(3):207–232, 2004.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. 2025.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. 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.

Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. 2016.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. 2016.

Aviv Shamsian, Ofri Kleinfeld, Amir Globerson, and Gal Chechik. Learning object permanence from video. 2020.

Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. 2013.

Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *IROS*, 2012.

Tianqi Tang, Shohreh Deldari, Hao Xue, Celso De Melo, and Flora Salim. Vilco-bench: Video language continual learning benchmark. 2024.

Pavel Tokmakov, Jie Li, Wolfram Burgard, and Adrien Gaidon. Learning to track with object permanence. 2021.

Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. In *3DV*, 2025.

Hengyi Wang and Lourdes Agapito. Amb3r: Accurate feed-forward metric-scale 3d reconstruction with backend. 2026.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. 2025a.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-omega. *arXiv preprint arXiv:2605.15195*, 2026a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. 2025b.

Runze Wang, Yuxuan Song, Youcheng Cai, and Ligang Liu. Stac: Plug-and-play spatio-temporal aware cache compression for streaming 3d reconstruction. 2026b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. 2024.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *ICLR*, 2026c.

Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.

Yuqi Wu, Wenzhao Zheng, Jie Zhou, and Jiwen Lu. Point3r: Streaming 3d reconstruction with explicit spatial pointer memory. In *NeurIPS*, 2025.

Tao Xie, Peishan Yang, Yudong Jin, Yingfeng Cai, Wei Yin, Weiqiang Ren, Qian Zhang, Wei Hua, Sida Peng, Xiaoyang Guo, et al. Scal3r: Scalable test-time training for large-scale 3d reconstruction. 2026.

Congrong Xu, Huachen Gao, Xingyu Chen, Yuliang Xiu, Jun Gao, and Anpei Chen. r^3: 3d reconstruction via relative regression. *arXiv preprint arXiv:2605.26519*, 2026.

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *CVPR*, 2025.

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.

Lan Yushi, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Bo Dai, Shuai Yang, Chen Change Loy, and Xingang Pan. Stream3r: Scalable sequential 3d reconstruction with causal transformer. In *ICLR*, 2026.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Chen Sun, Ming-Hsuan Yang, Forrester Cole, Trevor Darrell, and Deqing Sun. Loger: Long-context geometric reconstruction with hybrid memory. *arXiv preprint arXiv:2603.03269*, 2026.

Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. 2025a.

Shiduo Zhang, Zhe Xu, Peiju Liu, Xiaopeng Yu, Yuan Li, Qinghui Gao, Zhaoye Fei, Zhangyue Yin, Zuxuan Wu, Yu-Gang Jiang, et al. Vlabench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks. In *ICCV*, 2025b.

Zhijie Zheng, Xinhao Xiang, and Jiawei Zhang. Ttsa3r: Training-free temporal-spatial adaptive persistent state for streaming 3d reconstruction. *arXiv preprint arXiv:2601.22615*, 2026.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Streaming 4d visual geometry transformer. In *ICLR*, 2026.

Appendix

A Limitations

Memo3R focuses on feed-forward online reconstruction models with accessible implementations, and our main experiments use both indoor and outdoor RGB-D reconstruction datasets. Although results on three datasets show similar trends, broader validation on embodied navigation settings remains future work. Our plasticity protocol relies on generated object-removal sequences and depth completion, which enables controlled scene changes but may introduce artifacts on changed areas. Finally, memory footprint is not perfectly comparable across heterogeneous implementations, therefore we report implementation-specific memory components alongside reconstruction quality.

B Additional Related Works

Feed-forward 3D Reconstruction. Traditional 3D reconstruction pipelines include Structure from Motion (Agarwal et al., 2011; Kopf et al., 2021; Pollefeys et al., 1999; 2004; Schönberger & Frahm, 2016) and Multi-View Stereo (Furukawa & Hernández, 2015; Schönberger et al., 2016). In contrast, recent geometric foundation models (Wang et al., 2024; Leroy et al., 2024) enable feedforward 3D reconstruction from pairwise image input, directly predicting canonical 3D pointmaps. Follow-up works extend this paradigm to an arbitrary number of input views (Wang et al., 2025a; 2026c; Zhang et al., 2025a; Keetha et al., 2026). However, the bidirectional-attention-based frameworks cause quadratic computation cost and memory consumption when the number of input frames increases. VGG-T^3 (Elflein et al., 2026) linearizes the complexity by replacing global attention layers (Wang et al., 2025a) with MLP layers via test-time training. Despite these advances, the offline methods are not designed for incremental sequential input, motivating online reconstruction models that maintain memory across streams.

Detailed Memory Categorizations.

Evolving State Methods (Wang et al., 2025b; Chen et al., 2026b; Zheng et al., 2026; Jin & Ye, 2026) compress historical information into a persistent latent state. At each time step, the state is retrieved together with the current image and then updated to incorporate new observations. This memory is compact and naturally supports causal processing.

Test-time Training Compression Methods (Jin et al., 2026) store sequence-specific information in test-time-trainable layers. Instead of explicitly keeping past frames or geometric structures, the model updates a small set of fast weights online using the incoming observations, offering a highly compact memory representation.

KV Caching Methods (Yushi et al., 2026; Zhuo et al., 2026; Yuan et al., 2026; Lu et al., 2026) maintain a cache of keys and values computed from previous image tokens. Future frames can attend to cached tokens to retrieve historical visual context. This memory is directly accessible at the token level and can preserve detailed information, but its size grows with sequence length unless token pruning or eviction is applied.

Keyframe Methods (Xu et al., 2026; Chen et al., 2026a) retain selected historical frames and their associated representations to support future predictions. This memory mechanism resembles KV caching but operates at frame-level granularity, with memory updates and eviction performed on entire keyframes.

3D Memory Methods (Wu et al., 2025; Chen et al., 2025; Wang et al., 2026b; Wang & Agapito, 2025) store historical information in explicit spatial structures, such as voxels or spatial pointers. These memories are geometrically interpretable and can be queried or updated according to spatial proximity, thereby effectively preserving scene structure, but require effective fusion, merging, and pruning strategies to avoid redundancy and control memory growth.

C Effect of Memory Size on Persistence-Plasticity Trade-off

Table A1: Memory components retained during inference, including buffers used for memory updates.

Method	Memory components
CUT3R	Persistent state, pose memory
TTT3R	Persistent state, pose memory, cross-attention maps
TTSA3R	Persistent state, pose memory, cross-attention maps, previous candidate state, previous image features
FILT3R	Persistent state, pose memory, previous candidate state, per-token variance, drift-normalization EMA
ZipMap	Scene-adapted fast weights in all TTT layers
Point3R	Pointer positions and features, pose memory
InfiniteVGGT	Backbone KV cache, camera-head KV cache, layer-wise cache-allocation statistics
OVGGT	Backbone KV cache including protected anchors, camera-head KV cache, anchor-management state including anchor depth and pose for coverage-based selection
LingBot-Map	Backbone KV caches for anchor frames, the local sliding window, and trajectory tokens, camera-head KV caches
R^3	Backbone KV cache, retained camera and relative-pose features, keyframe-selection features, reference poses, keyframe scores and metadata

In this section, we investigate how memory size shapes memory characteristics. Since equal byte counts do not imply equal representational capacity across heterogeneous memory forms, a strictly equal-budget comparison across all methods is not directly achievable. Instead, we vary the memory budget in explicit-memory methods, as they are directly accessible, and obtain results of persistence and plasticity results under different memory sizes. For a fair comparison between two tests, regarding plasticity test, we sample original sequences to match the length of the evaluated sequences in the persistence test, then prepend their edited counterparts to construct the plasticity test streams. This ensures that the evaluations of two tests are on the same frames.

Method	**Memory (MiB)**
CUT3R	3.75
TTT3R	436.31
TTSA3R	441.56
FILT3R	12.75
ZipMap	576.47
Point3R	8.15
InfiniteVGGT	7031.2
OVGGT	1171.9
R^3	9875.8
LingBot-Map	10988.5

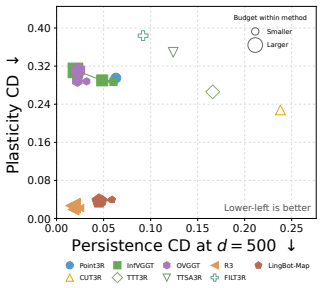


Figure A1: Left: Memory size in MiB. Right: Persistence–plasticity trade-off under varying memory budgets.

Memory Components. In Table A1, we list the memory components of each method used to calculate the memory-specific sizes shown in Fig. A1 (left).

Adaptation becomes the main bottleneck in larger memory. Fig. A1 (Right) compares explicit-memory methods on the 7-Scenes persistence and plasticity tests. Larger memories improve retention, but plasticity saturates or even deteriorates as memory size increases. For example, plasticity CD increases from 0.288 to 0.311 for OVGGT and from 0.019 to 0.027 for R^3, indicating worse adaptation. We hypothesize that larger memories retain more stale information, delaying adaptation to changed regions. With abundant memory, effective adaptation therefore becomes a central challenge, consistent with observations in continual learning (Cho et al., 2026).

Persistence under different memory budgets. We further investigate how memory capacity influences persistence, examining whether the superior persistence of explicit memories stems from their larger memory budgets or the nature of their representations. To this end, we evaluate explicit-memory methods on the large-scale outdoor dataset VKITTI2 across multiple memory budgets, approximately matched to those of implicit-memory methods.

We show the results in Table A2. With budgets of 640k and 1.2M tokens, InfiniteVGGT consistently improves as the retention interval d increases, matching the trend observed on 7-Scenes. At

Table A2: Persistence results at different memory sizes. Relative changes between intervals are shown alongside the absolute Chamfer Distance.

Method	Memory size	$d = 0$	$d = 50$	$d = 100$	$d = 500$
CUT3R	CUT3R	5.457	10.068 (-84.50%)	16.596 (-64.85%)	29.028 (-74.91%)
Point3R	$\approx$ CUT3R	12.328	17.795 (-44.35%)	18.518 (-4.06%)	20.475 (-10.57%)
Point3R	$\approx 2\times$ CUT3R	11.942	13.973 (-17.01%)	14.931 (-6.86%)	15.447 (-3.46%)
Point3R	$\approx 4\times$ CUT3R	11.324	13.061 (-15.34%)	13.435 (-2.86%)	17.203 (-28.06%)
FILT3R	$\approx 4\times$ CUT3R	1.317	2.604 (-97.72%)	11.894 (-356.76%)	34.471 (-189.82%)
TTT3R	$\approx 100\times$ CUT3R	1.174	3.139 (-167.50%)	13.003 (-314.30%)	30.980 (-138.25%)
TTSA3R	$\approx$ TTT3R	1.000	3.344 (-234.40%)	13.270 (-296.83%)	36.684 (-176.44%)
ZipMap	$\approx 1.3\times$ TTT3R	1.446	3.238 (-123.89%)	4.285 (-32.33%)	4.294 (-0.20%)
InfiniteVGGT	80k $\approx$ TTT3R	7.668	8.802 (-14.79%)	9.477 (-7.67%)	10.323 (-8.93%)
InfiniteVGGT	640k $\approx 8\times$ TTT3R	9.948	8.592 ($+13.63\%$)	8.562 ($+0.34\%$)	6.906 ($+19.35\%$)
InfiniteVGGT	1.2M $\approx 15\times$ TTT3R	9.076	7.086 ($+21.93\%$)	6.776 ($+4.37\%$)	5.946 ($+12.25\%$)
R^3	$\approx 23\times$ TTT3R	0.875	0.890 (-1.71%)	0.879 ($+1.24\%$)	1.184 (-34.70%)
LingBot-Map	$\approx 25\times$ TTT3R	0.363	0.634 (-74.66%)	1.072 (-69.09%)	3.110 (-190.11%)

the smallest budget of 80k tokens, however, its cache retains little context beyond the anchor frame, and reconstruction quality instead deteriorates with increasing d. This reversal suggests that InfiniteVGGT's ability to benefit from intervening observations depends on sufficient cache capacity. Nevertheless, both Point3R and InfiniteVGGT exhibit smaller relative increases in reconstruction error from $d = 0$ to $d = 500$ at their minimum evaluated budgets than the implicit-memory methods, including evolving-state and TTT-compression approaches. These results suggest that persistence depends on both memory capacity and memory design: explicit memories can provide stronger temporal retention even under constrained budgets, although this does not necessarily translate into lower absolute reconstruction error.

D More Implementation Details

We evaluate the proposed protocols on NVIDIA RTX A6000 GPUs. A single evaluation of persistence, plasticity, or robustness can take several hours on 4 GPUs, while the compactness test takes approximately 1 hour.

Test score calculation. For the four diagnostic dimensions in Fig. 1, we compute relative scores from full-scene Chamfer Distance (CD) as $E_{\text{ref}}/E_{\text{test}}$, where higher values indicate better performance relative to the reference. For persistence, we use $d = 0$ as the reference and $d = 500$ as the test, and divide the resulting ratio by two for visualization. For plasticity, we compare $(r_a, r_b) = (1, 5)$ against the reference $(0, 1)$, averaging the ratios across object removal and addition where available; VKITTI2 includes only object addition. For robustness, we use the clean-sequence result as the reference and average the ratios for the strongest Gaussian noise ($n = 0.50$) and motion blur ($m = 90$). For compactness, we use the sparse sequence $\mathcal{I}^{(10)}$ as the reference and the dense sequence $\mathcal{I}^{(1)}$ as the test. We compute ratios before averaging over settings or directions within each dataset, then average the three dataset scores with equal weight. Unchanged error corresponds to a score of 1, or 0.5 for persistence; improvements beyond these values are retained.

Persistence test. The evaluation is implemented by selecting every frame that satisfies $a + d \leq T$ as the anchor frame. For fair comparison across different retention intervals, we evaluate all intervals on the shared valid anchor image range, defined by the largest retention interval. Since the maximum interval is $d = 500$, we use the first 500 frames of each 1000-frame sequence as anchor frames.

Plasticity test. For the indoor datasets 7-Scenes and Bonn, we synthesize images and geometry by removing objects, as detailed below together with our quality assessment procedure. For the outdoor dataset VKITTI2, reliable inpainting is more challenging: removing vehicles can expose large, previously occluded regions, while street geometry and depth variations make it difficult to maintain consistency between the synthesized appearance and geometry across frames. We therefore use the provided instance masks in VKITTI2 to select foreground objects across video frames and replace their image regions with black pixels, creating a proxy for post-change observations. Our results suggest that this masking-based setting also supports plasticity evaluation without requiring

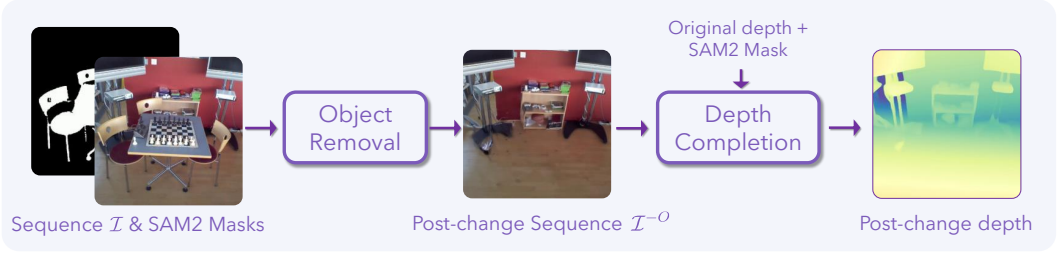


Figure A2: Workflow for generating the post-change geometry ground truth.

synthesizing and inpainting. This broadens the applicability of our protocol to datasets with ground-truth geometry and temporally consistent object masks.

Data generation. Fig. A2 illustrates the workflow for generating post-change geometry. For each video sequence, we first select and segment target foreground objects using SAM2 (Ravi et al., 2025). We then remove these objects using a video object-removal diffusion model (Fu et al., 2026). Finally, we apply a depth completion model (Ke et al., 2026) to estimate depth in the inpainted regions, yielding post-change geometry for evaluation.

Quality control and validation. To control data quality, we manually select prompt points for SAM2, visually verify the temporal and spatial consistency of resulting video object masks, and inspect the mask-based inpainting results to exclude samples with artifacts. We further assess depth-inpainting accuracy on the original 7Scenes dataset by masking ground-truth depth maps with the same SAM2 masks, recovering the masked depth with the same depth-inpainting model and comparing the predictions against the corresponding ground truth. The MAE and AbsRel results can be as low as 0.0095 and 0.0127, respectively. To constrain the influence of generated geometry on plasticity evaluation, we propose two evaluation settings, in which the object-addition setting uses the original ground-truth depth as its evaluation target, same for the unchanged-background evaluation in the object-removal setting. Qualitative examples of the generated images and geometry on are provided in Fig. A3.

Robustness test. We vary both the corruption type and level. As Sec. 4.3, we consider three common image perturbations: Gaussian noise, motion blur and lighting change. For Gaussian noise, we perturb each input image by adding zero-mean i.i.d. noise to the normalized RGB values:

$$\tilde{I} = \text{clip}(I + \epsilon, 0, 1), \qquad \epsilon \sim \mathcal{N}(0, \sigma^2),$$

where $\sigma \in \{0.05, 0.1, 0.5\}$ controls the noise level. For motion blur, we convolve each image with a linear motion-blur kernel. The kernel is initialized as a horizontal line, randomly rotated by an angle sampled uniformly from $[0°, 180°]$, and normalized to sum to one. We vary the kernel size over $\{10, 30, 90\}$, where larger kernels correspond to stronger blur. For lighting changes, we apply a global affine transformation to the normalized RGB values:

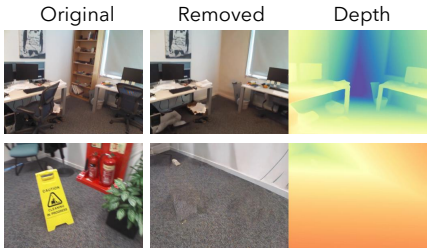


Figure A3: More examples of inpainted images and geometry.

$$\tilde{I} = \text{clip}\big((1 + 0.35s)I + 0.05s,\ 0,\ 1\big)$$

where $s \in \{-0.5, -1.0, -2.0\}$ controls the perturbation severity. The same transformation is applied uniformly across all pixels and RGB channels, with more negative values of s producing stronger darkening effects.

Evaluation interval. Because repeatedly processing perturbed frames by selecting every frame as anchor frame can cause large computational cost, we set the interval of anchor frame as every five frames, and only evaluate reconstruction quality on the selected anchor frames.

Method configurations. We use the default settings provided in each method's public implementation. For R^3, we use the R^3-long checkpoint. LingBot-Map's online inference requires a buffer of eight images to be accumulated and processed jointly, introducing a short delay that violates our strictly online, frame-by-frame evaluation protocol.

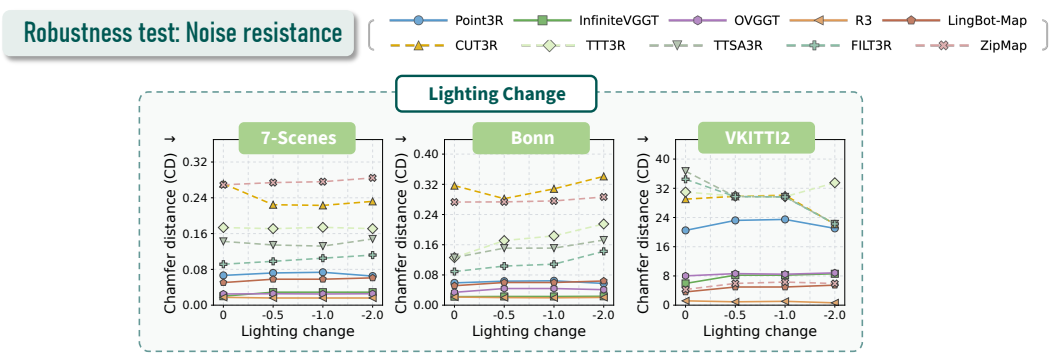


Figure A4: Results of robustness test under increasing lighting change.

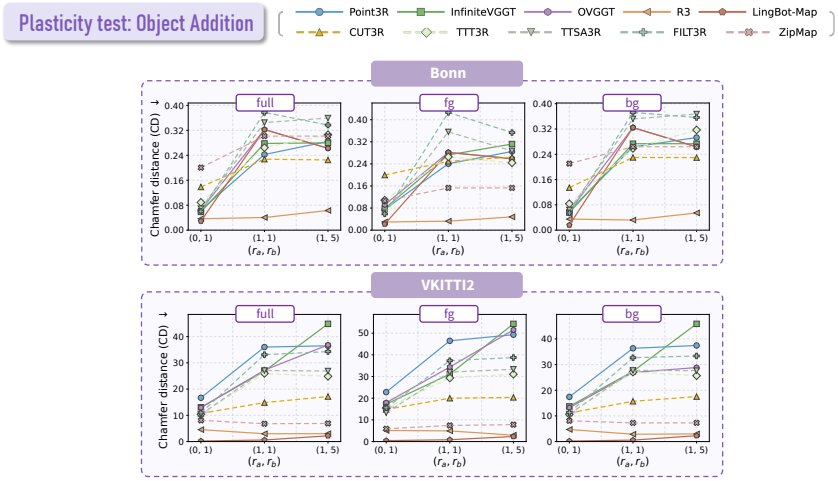


Figure A5: Full results of object-addition plasticity test on Bonn and VKITTI2. fg: changed foreground. bg: unchanged background.

E Additional Results

We provide additional plasticity and robustness results, and a persistence evaluation under varying memory budgets.

Plasticity tests. In addition to the full 7-Scenes results in Sec. 4.2, we present full object-addition results for Bonn and VKITTI2 in Fig. A5. We also report object-removal results in Fig. A6, evaluated against synthesized post-change geometry. These additional results support our findings in Sec. 4.2 that strong persistence does not necessarily imply strong plasticity: KV-caching and spatial-memory methods often exhibit substantial degradation after scene changes, and repeated post-change observations do not consistently restore reconstruction quality. However, strong results from R^3 indicate that this trade-off depends on the memory design, potentially on memory update strategies, rather than applying uniformly to all explicit memories. For object removal setting, foreground evaluation uses inpainted depth as ground truth and may therefore be affected by synthesis artifacts, whereas background evaluation uses the original ground-truth geometry, avoiding uncertainty from synthesized evaluation targets.

Robustness tests. In addition to the Gaussian noise and motion blur results in Sec. 4.3, Fig. A4 presents results under lighting changes. The evaluated methods generally exhibit limited sensitivity to darkening, maintaining relatively stable reconstruction performance across the tested severity levels.

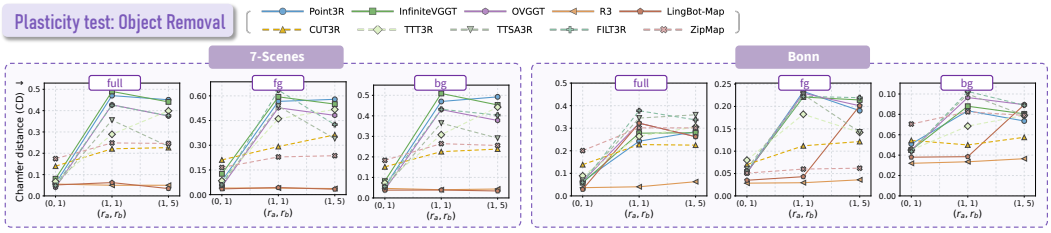


Figure A6: Results of plasticity test under 7-Scenes and Bonn on object removal. fg: changed foreground. bg: unchanged background.

F Qualitative Results

We present qualitative results of the persistence test in Figs. A7 to A12. Explicit-memory methods preserve reconstruction quality even at a retention interval of $d = 500$. Implicit-memory methods exhibit severe degradation at large intervals. For example, though FILT3R improves over CUT3R, TTT3R, and TTSA3R at $d = 50$ and $d = 100$, it degrades significantly at $d = 500$.

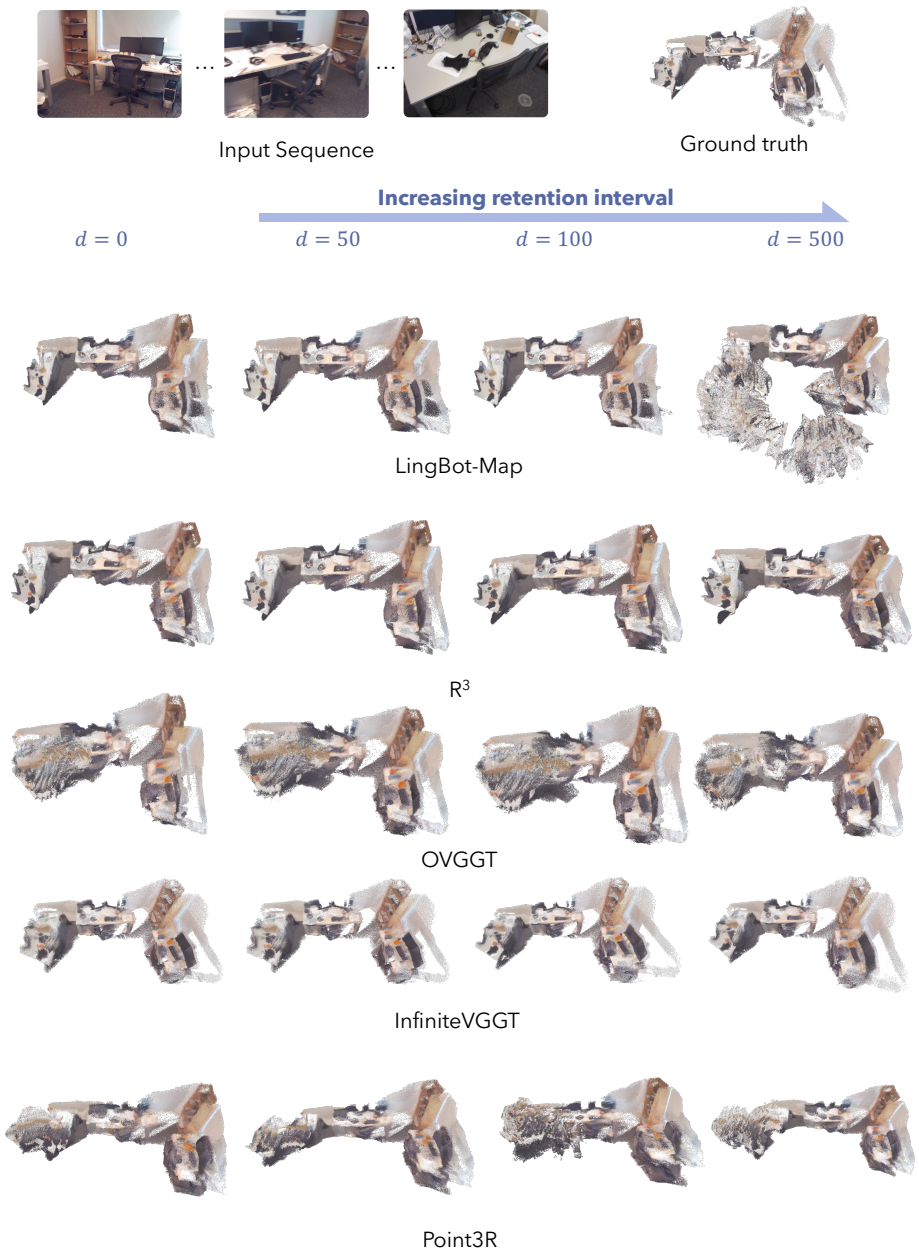


Figure A7: Qualitative examples of explicit-memory methods on 7-Scenes persistence test.

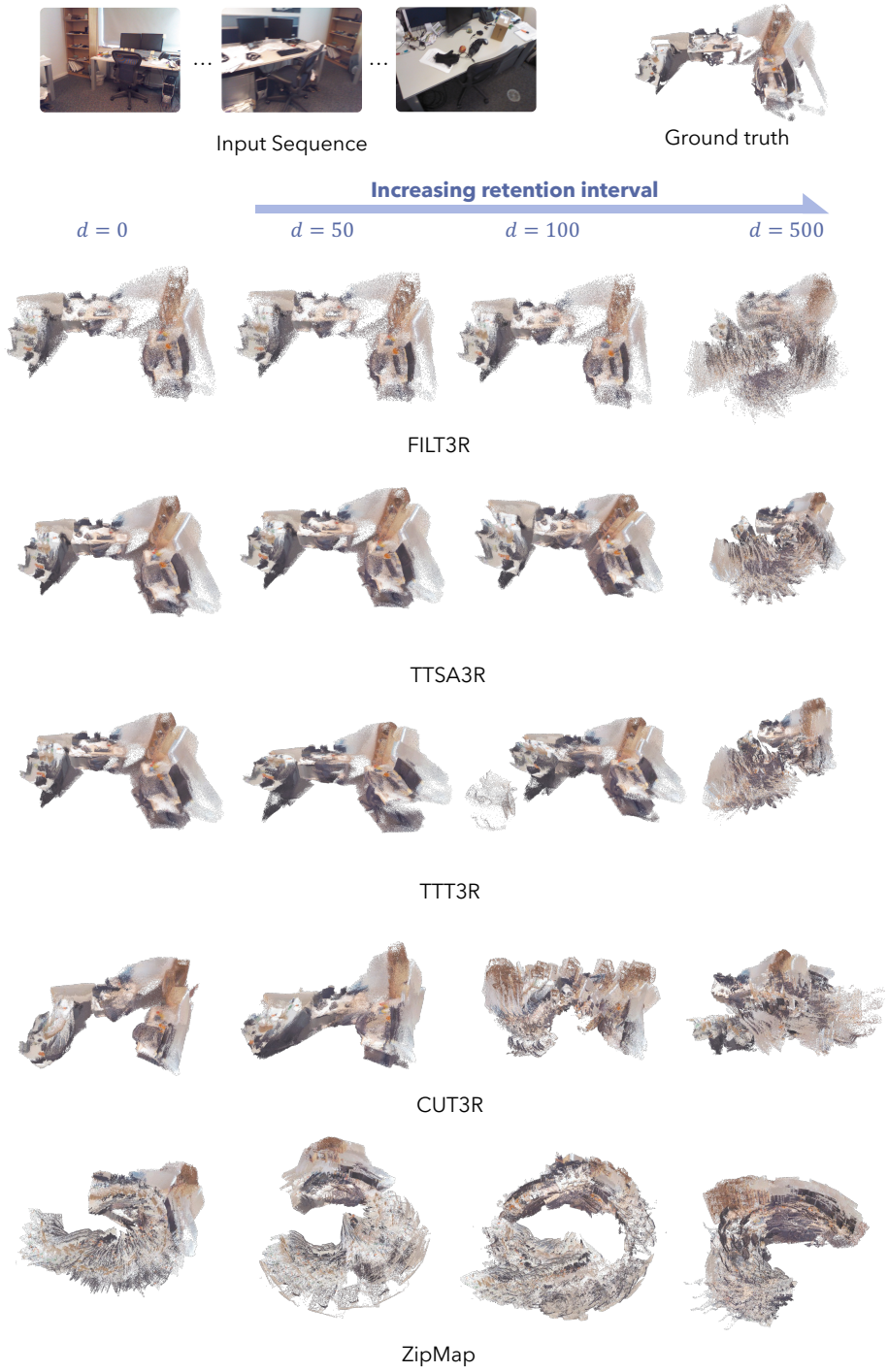


Figure A8: Qualitative examples of implicit-memory methods on 7-Scenes persistence test.

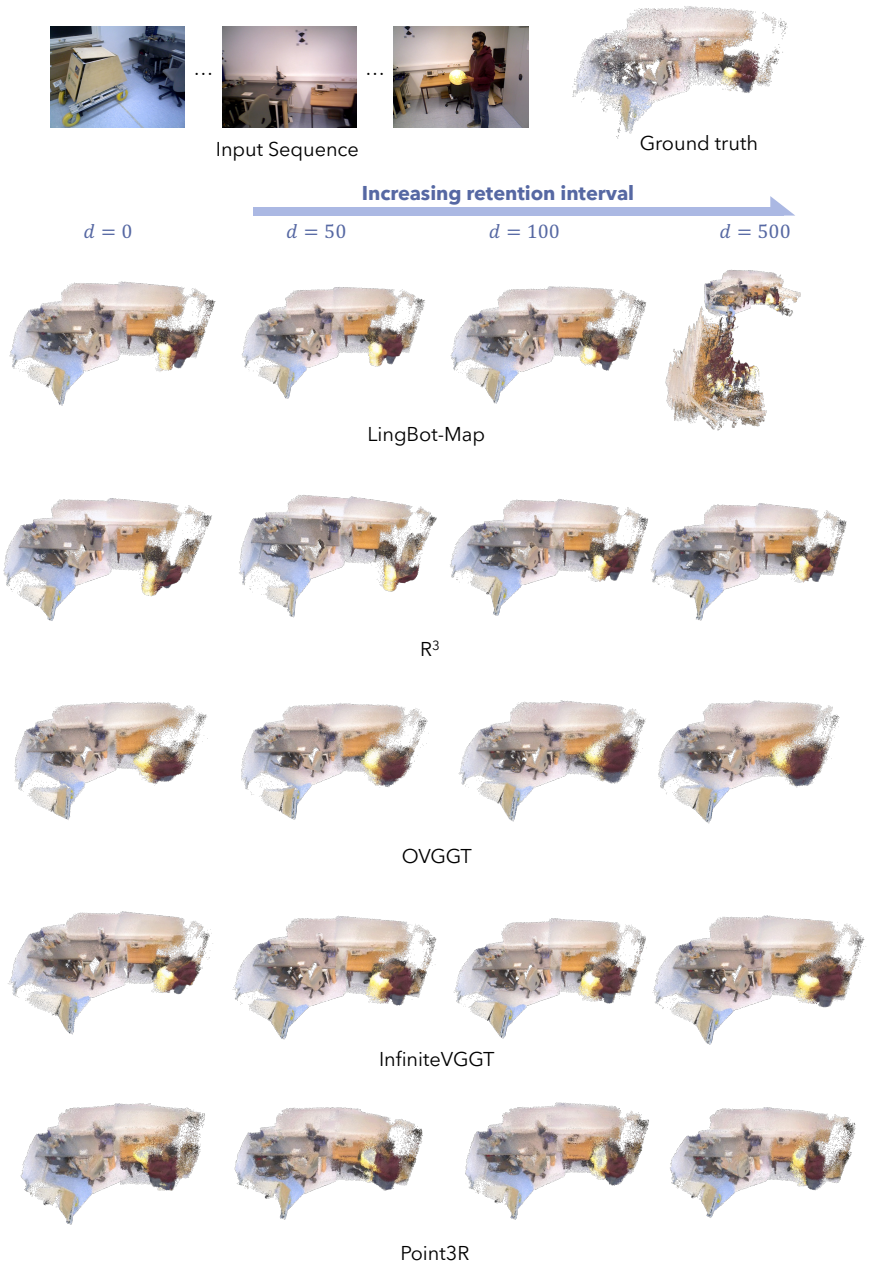


Figure A9: Qualitative examples of explicit-memory methods on Bonn persistence test.

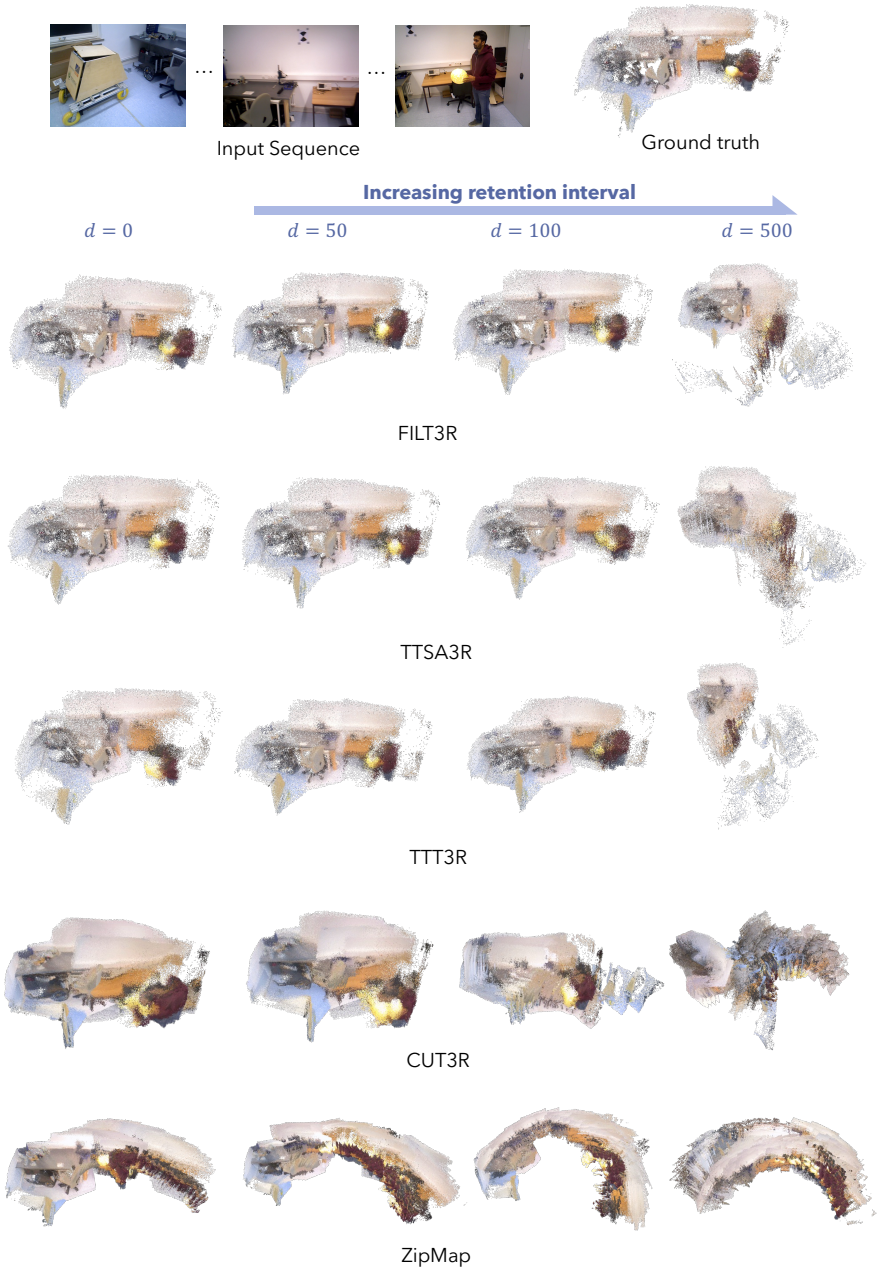


Figure A10: Qualitative examples of implicit-memory methods on Bonn persistence test.

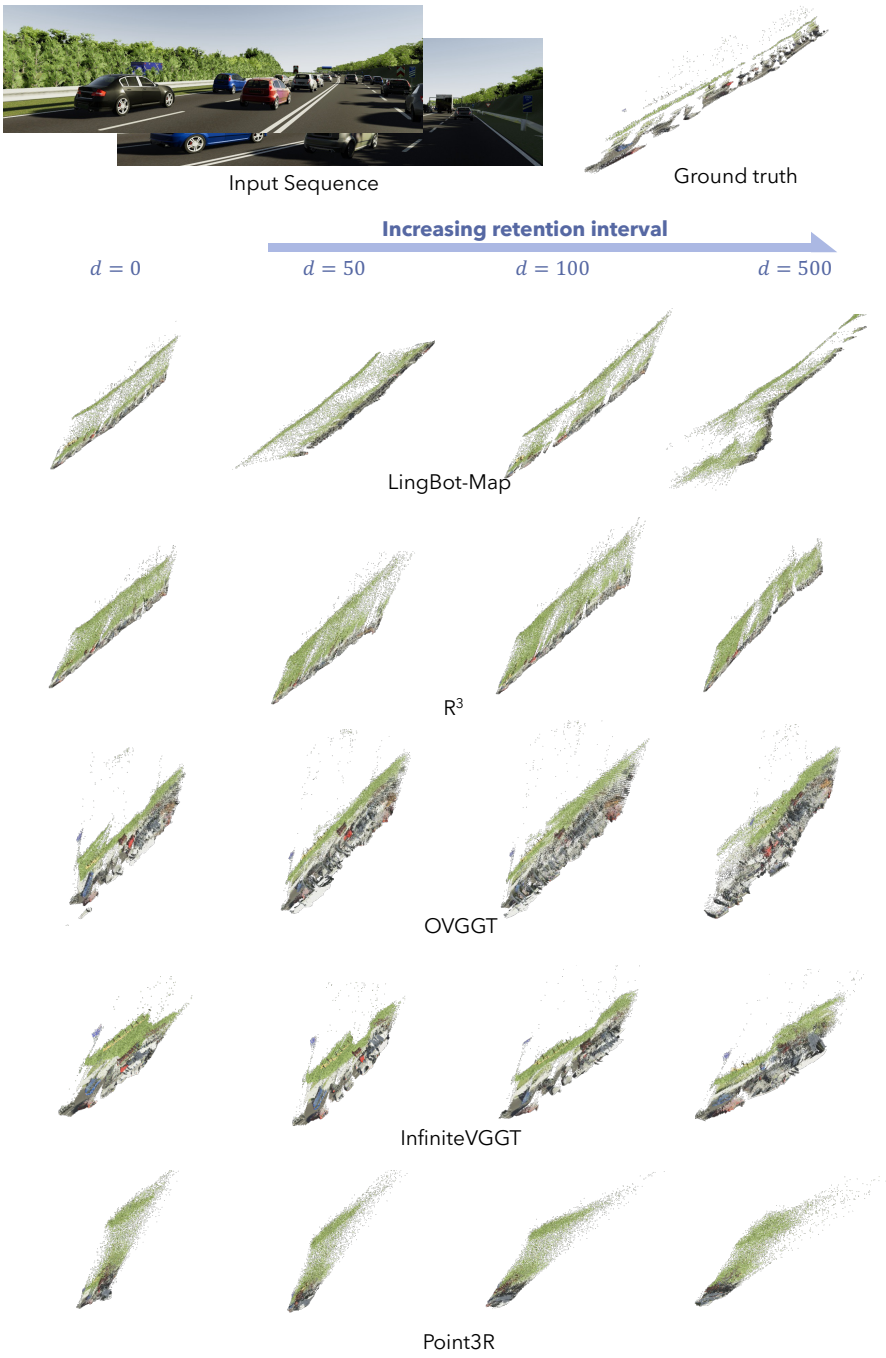


Figure A11: Qualitative examples of explicit-memory methods on VKITTI2 persistence test.

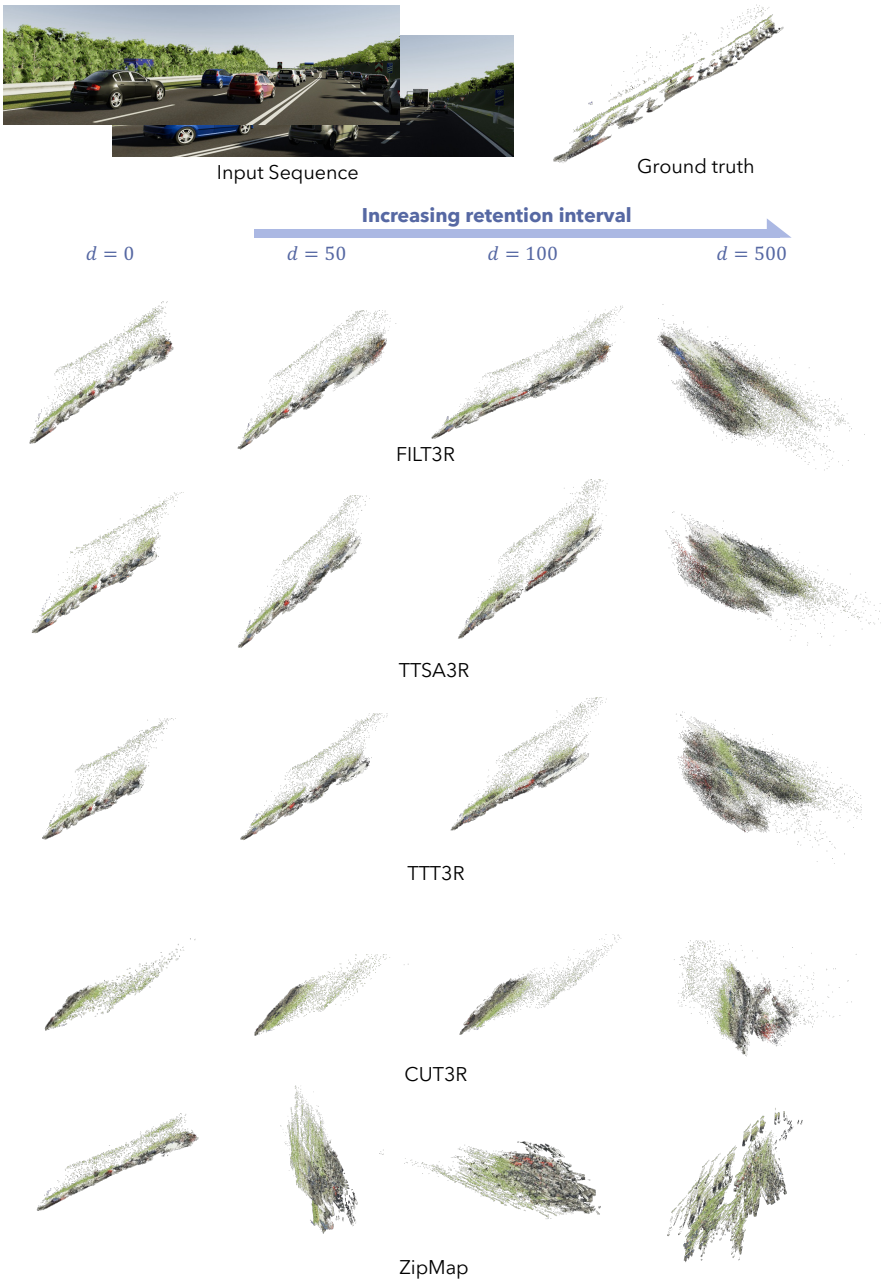


Figure A12: Qualitative examples of implicit-memory methods on VKITTI2 persistence test.