

Mem3R: Streaming 3D Reconstruction with Hybrid Memory

Changkun Liu^{1,2,*} Jiezhi Yang^{1,*} Zeman Li¹ Yuan Deng¹

Jiancong Guo¹ Binbin Xu¹ Luca Ballan¹

¹Google ²The Hong Kong University of Science and Technology

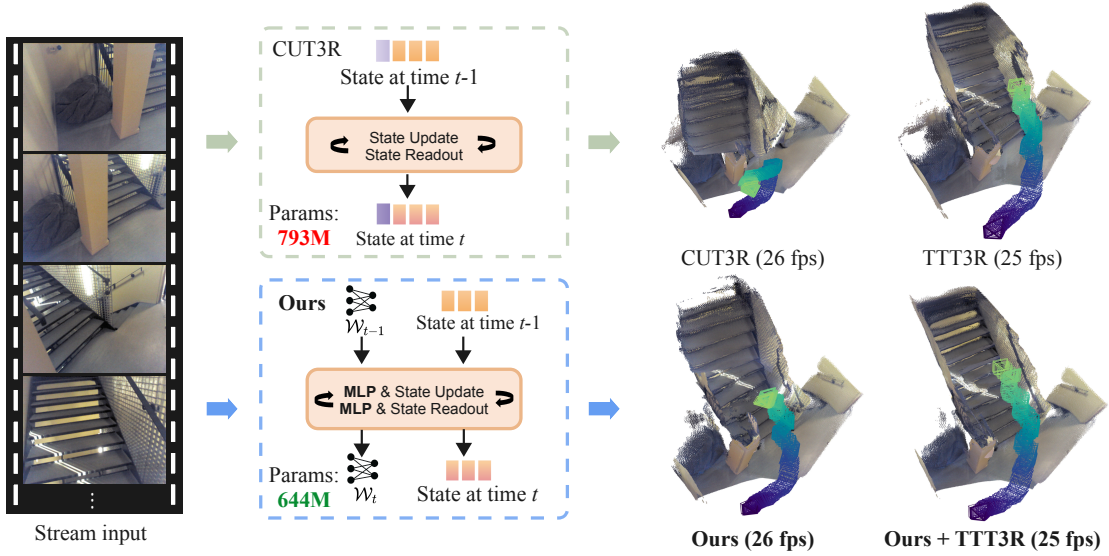


Figure 1. **Mem3R** is an RNN-based model built on the CUT3R paradigm that achieves stronger long-sequence streaming 3D perception. Built on a dual-memory design, Mem3R combines (i) an *implicit memory* $\mathcal{W}$ for camera pose estimation and (ii) an *explicit memory* of persistent tokens for global geometric context. Mem3R is further compatible with plug-and-play state-update strategies developed for CUT3R, such as TTT3R.

Abstract

Streaming 3D perception is well suited to robotics and augmented reality, where long visual streams must be processed efficiently and consistently. Recent recurrent models offer a promising solution by maintaining fixed-size states and enabling linear-time inference, but they often suffer from temporal forgetting over long sequences due to the limited capacity of compressed memories. We propose **Mem3R**, a streaming 3D reconstruction model with a hybrid memory design that decouples camera tracking from geometric mapping to improve temporal consistency over long sequences. For camera tracking, Mem3R employs an implicit fast-weight memory implemented as a lightweight Multi-Layer Perceptron updated via Test-Time Training. For geometric mapping, Mem3R maintains an explicit token-based

fixed-size state. Compared with CUT3R, this design not only significantly improves long-sequence performance but also reduces the model size from 793M to 644M parameters. Mem3R supports existing improved plug-and-play state update strategies developed for CUT3R. Specifically, Mem3R combined with TTT3R reduces absolute trajectory error by up to 39% relative to CUT3R combined with TTT3R on 500- to 1000-frame sequences from the TUM Dynamics dataset. The resulting improvements also extend to other downstream tasks, including video depth estimation and 3D reconstruction. Project page: <https://lck666666.github.io/Mem3R/>.

1. Introduction

A central challenge in real-time 3D perception is to recover accurate scene geometry and camera motion from long visual streams under strict computational and memory bud-

*Equal Contribution.

[†]Work done during a student researcher internship at Google.

gets. This capability is essential for a wide range of mobile applications, including robotic visual localization, navigation, and augmented reality. Recent progress in feed-forward multi-view geometry, driven by models such as DUST3R [53] and VGGT [51], has inspired a new wave of Transformer-based architectures [27, 42, 46, 54, 58, 64] that achieve impressive performance in 3D reconstruction and pose estimation. However, these methods typically rely on global pairwise alignment or dense cross-frame attention, leading to computational costs that scale quadratically with sequence length and limit their applicability in long-horizon streaming scenarios.

To enable scalable streaming, prior work has mainly evolved along two directions. The first line of work leverages causal-attention Key-Value (KV) caches to preserve the full observation history [26, 50, 61, 67]. While effective, this paradigm incurs memory consumption that grows linearly with sequence length, making it increasingly vulnerable to out-of-memory (OOM) failures during long-horizon inference. The second direction, exemplified by CUT3R, adopts a Recurrent Neural Network (RNN)-based architecture with a fixed-size latent state, enabling constant-memory inference. However, this compact state also creates an information bottleneck: once the sequence length exceeds the typical training horizon, CUT3R suffers from substantial temporal forgetting and trajectory drift. To alleviate these issues, TTT3R [10] reinterprets state updates through the perspective of Test-Time Training (TTT) and introduces per-token update weights that emphasize tokens most relevant to the current observations. Building on this idea, subsequent works [14, 66] have proposed alternative plug-and-play update strategies to further improve CUT3R.

In this work, we present **Mem3R**, an enhanced RNN-based architecture with a dual-memory design for long-horizon streaming 3D perception. Our key idea is to improve the representational efficiency of the recurrent state by decoupling camera tracking from geometric mapping. Specifically, Mem3R replaces CUT3R’s pose-related token memory and its Transformer-based memory read/write blocks (implemented as the `LocalMemory` module in the official CUT3R codebase) with a lightweight implicit Multi-Layer Perceptron (MLP)-based memory for pose estimation via TTT inspired by LaCT [65], while retaining an explicit token-based memory to preserve global geometric context. This dual-memory design yields a more compact and effective backbone, reducing the parameter count from 793M to 644M (a 19% reduction) while achieving stronger performance, as illustrated in Fig. 1. We also introduce a learnable channel-wise state update module, inspired by recent advances in finite-state RNN architectures [4, 47], to enable finer-grained control over state updates.

As a result, Mem3R remains fully compatible with training-free per-token state-update strategies such as

TTT3R [10] and TTSA3R [66], and it delivers further gains when combined with them. Across camera pose estimation, video depth estimation, and 3D reconstruction, Mem3R shows clear improvements in most evaluated settings while preserving the inference speed and constant-memory advantages of RNN-based streaming models. These results establish Mem3R as a stronger and more memory-efficient backbone for long-sequence streaming 3D perception.

2. Related Work

Offline Feed-forward 3D Reconstruction. DUST3R [53] proposes an end-to-end, feed-forward approach to 3D reconstruction, directly regressing point maps from image pairs. Subsequent extensions [9, 11, 16, 17, 23, 27, 32, 57, 62] broadened this formulation to more challenging settings, but most remain fundamentally pairwise and therefore require costly global alignment once more than two views are involved. To overcome this limitation, recent offline multiview models [18, 25, 42, 51, 54, 58, 64] employ large feed-forward Transformers with global attention to jointly process all input views and predict per-view geometry and camera poses in a single forward pass. Representative examples such as VGGT [51] achieve strong reconstruction quality, but their computational and memory costs scale quadratically with the number of frames. Moreover, they are inefficient for streaming data, as incorporating a newly arrived view typically requires reprocessing the entire sequence. Although methods such as FastVGGT [42] reduce inference overhead through token merging, these approaches remain better suited to offline reconstruction than to real-time long-sequence streaming settings.

Online 3D Reconstruction. Online 3D reconstruction methods [12, 26, 30, 50, 52, 55, 61, 67] process views incrementally and update scene representations on the fly, making them attractive for robotics and embodied perception. Existing approaches mainly differ in how they preserve historical context. Causal-cache methods, such as StreamVGGT [67], retain past keys and values for long-range temporal conditioning, but their memory and computation still grow with sequence length. In contrast, recurrent methods such as CUT3R [52] maintain a fixed-size latent state for constant-memory inference. However, compressing long-term observations into a compact state creates an information bottleneck, often causing temporal forgetting and drift on long sequences. Recent training-free update strategies, including TTT3R [10] and TTSA3R [66], improve CUT3R-style recurrent updates, while Point3R [55] enhances memory recall through explicit 3D point associations at the cost of memory growth. In contrast, our method improves the representational efficiency of the recurrent backbone through a dual-memory design, while remaining compatible with plug-and-play update strategies

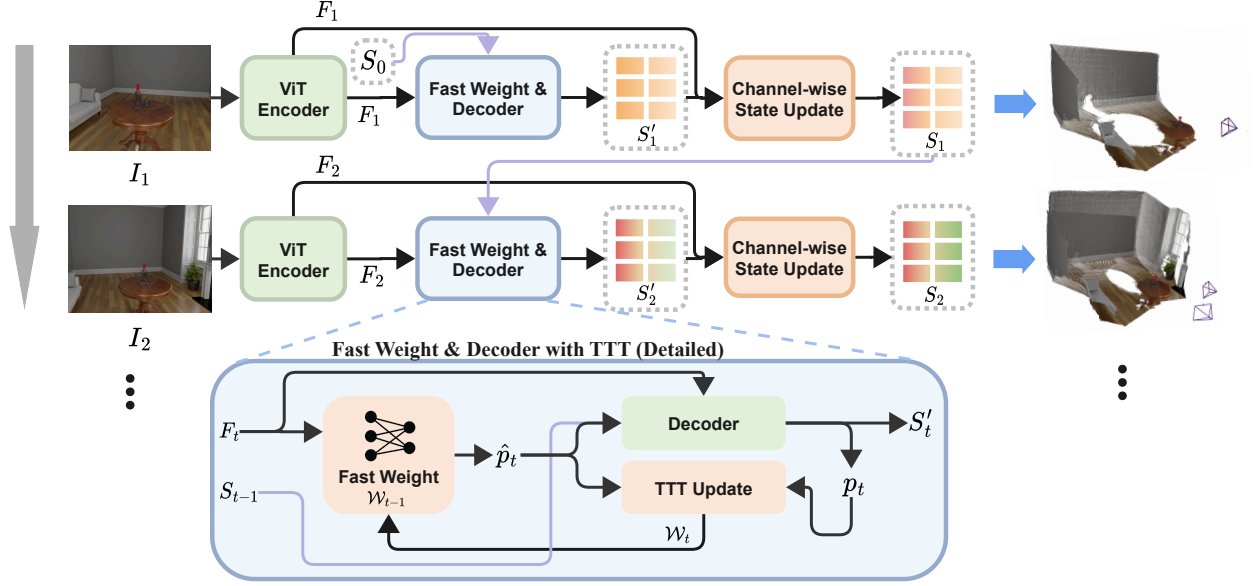


Figure 2. Overview of Mem3R. **Top:** For each frame I_t in the streaming image sequence, a ViT encoder extracts image features F_t . The MLP-based fast-weight module $\mathcal{W}$ performs camera tracking with decoder under Test-Time Training (TTT), while the fixed-size state S and decoder preserve and update geometric information, producing an intermediate state S'_t . S'_t is then fused with the previous state S_{t-1} through a channel-wise update gate module (see Sec. 3.3) to obtain the final updated state S_t . By replacing pose-related state token with $\mathcal{W}$, Mem3R maintains a constant-size recurrent state while reducing long-term temporal forgetting in streaming 3D reconstruction. **Bottom:** Illustration of the fast-weight and decoder module with TTT (see Sec. 3.2).

such as TTT3R and TTSA3R.

Fast Weight Programs and Memory Architectures.

The view of linear layers as associative memories dates back to Hopfield networks [21] and was later developed in fast weight programmers, where dynamic fast programs are integrated into recurrent neural networks as writable memory stores [38–40]. Building upon these foundational concepts, modern sequence modeling has diverged into two primary structural paradigms: *linear memory modules* and *deep memory modules*. *Linear memory modules* [20, 22, 35, 47, 59, 60] maintain matrix-valued hidden states with linear state transitions, while *deep memory modules* [4–7, 29, 45, 65] use online-adapted sub-networks with dynamically updated parameters to encode contextual information. Our method is particularly inspired by this latter line of work, using an online-adapted implicit memory to improve long-horizon performance. Concurrent works [24, 49, 63] also incorporate TTT memory layers into 3D perception but address different settings: tt-LRM [49] targets novel view synthesis with 3D Gaussian Splatting, [15, 24, 56] focus on faster offline reconstruction, and LoGeR [63] studies chunk-wise reconstruction with sliding-window attention rather than frame-by-frame streaming reconstruction.

3. Method

In streaming 3D perception, a model processes an image sequence $\{I_t\}_{t=1}^N$ incrementally, estimating camera motion and scene geometry online as new observations arrive.

3.1. Overview

We begin by briefly reviewing the recurrent state formulation in CUT3R to highlight the key differences in our design:

$$X'_t, \bar{S}_t = \text{Decoders}(X_t, S_{t-1}), \quad (1)$$

where S_{t-1} is the state at time $t-1$, $\bar{S}_t$ is the updated state produced directly by the decoder, and $X_t = [z_t, F_t]$ consists of a learnable pose token z_t and image tokens F_t extracted from the current frame I_t .

Our Mem3R is also an RNN-based architecture that maintains a fixed-size recurrent state for online 3D perception. The key idea is to improve the efficiency of the recurrent memory representation by replacing the pose-related components of the recurrent state and transformer-based decoder blocks with an implicit MLP. To this end, Mem3R uses a *hybrid memory* design: an *implicit memory* $\mathcal{W}$, implemented as an MLP-based fast-weight module for camera tracking, and an *explicit memory* S , implemented as persistent state tokens for geometric mapping, as shown in Fig. 1 and Fig. 2. Unlike CUT3R, which directly writes decoder outputs into the recurrent state, our model treats the state

output from the decoder as a candidate update and integrates it with a learnable channel-wise gate.

At time t , the current frame is encoded into visual tokens F_t . The implicit memory predicts a pose token prior $\hat{p}_t$, which is processed together with F_t and the previous explicit state S_{t-1} by a transformer decoder. The decoder produces a posterior pose token p_t , updated visual features F'_t , and a candidate state update S'_t . The implicit memory is then updated online by aligning prior $\hat{p}_t$ with posterior p_t , while the explicit memory is updated through channel-wise gated integration of S'_t .

3.2. Implicit Memory via Test-Time Training

To improve long-horizon camera tracking while preserving a constant inference speed, we introduce an adaptive fast-weight memory implemented as a lightweight SwiGLU MLP [41]:

$$f_{\mathcal{W}}(x) = W_2(\text{SiLU}(W_1x) \odot (W_3x)), \quad (2)$$

where $\mathcal{W} = \{W_1, W_2, W_3\}$ denotes the fast weights updated online during inference.

Given the current visual tokens F_t , we first compute a L_2 -normalized query feature κ_t and read a pose prior from the previous fast weights $\mathcal{W}_{t-1}$:

$$\hat{p}_t = f_{\mathcal{W}_{t-1}}(\kappa_t). \quad (3)$$

This prior replaces CUT3R's learned pose token z_t , yielding $X_t = [\hat{p}_t, F_t]$. The decoder then produces

$$X'_t, S'_t = \text{Decoders}(X_t, S_{t-1}), \quad (4)$$

where $X'_t = [p_t, F'_t]$, with p_t defined as the refined posterior pose token and F'_t as the state-enriched visual features.

We update the implicit memory by aligning the prior and posterior:

$$\mathcal{L}_{\text{ttt}}(\hat{p}_t, p_t) = \langle \hat{p}_t, p_t \rangle. \quad (5)$$

To make the update adaptive for different I_t , we predict a per-step decay factor $\alpha_t = 1 - \gamma \cdot \sigma(W_\alpha F_t)$ and per-layer learning rates η_t from the current features:

$$\eta_t = \text{Softplus}(W_\eta F_t + \text{Softplus}^{-1}(\eta_{\text{base}})), \quad (6)$$

where $\gamma = 0.01$ is the decay scale and $\eta_{\text{base}} = 0.001$ is the base learning rate. Here, $\text{Softplus}^{-1}(x) = \log(e^x - 1)$ ensures that $\eta_t = \eta_{\text{base}}$ when $W_\eta F_t = 0$. $\sigma(\cdot)$ denotes the sigmoid function. The fast weights are then updated by

$$\mathcal{W}_t = \alpha_t \mathcal{W}_{t-1} + \eta_t \odot \nabla_{\mathcal{W}} \mathcal{L}_{\text{ttt}}. \quad (7)$$

This update enables the model to estimate accurate pose cues while filtering out transient noise, effectively turning the fast-weight module into a compact implicit temporal memory for robust camera tracking. We provide more details about this module in the supplementary material.

3.3. Channel-wise Update of Explicit Memory

To represent persistent global geometry, we retain an explicit token-based memory $S_t \in \mathbb{R}^{N_s \times C}$, where N_s is the number of state tokens and C is the channel dimension. The decoder output S'_t in Eq. (4) is treated as a *candidate* state update rather than the final new state.

Directly overwriting the state can lead to temporal forgetting in long sequences. To address this issue, we introduce a channel-wise update gate $\zeta_t \in (0, 1)^{N_s \times C}$:

$$\zeta_t = \sigma(f_\zeta(F_t, S_{t-1})), \quad (8)$$

where f_ζ is an MLP. The explicit memory is then updated as

$$S_t = \zeta_t \odot S'_t + (1 - \zeta_t) \odot S_{t-1}. \quad (9)$$

This channel-wise gating enables finer control over memory evolution, allowing stable geometric features to be preserved while selectively integrating new observations. During inference, when applying the per-token state-update mechanism of TTT3R or TTSA3R, we combine its per-token gate G_t with our channel-wise gate ζ_t to obtain the final update gate. The state is updated as

$$S_t = G_t \odot \zeta_t \odot S'_t + (1 - G_t \odot \zeta_t) \odot S_{t-1}, \quad (10)$$

where $G_t \in \mathbb{R}^{N_s \times 1}$ is broadcast along the channel dimension and controls token-level updates, while $\zeta_t \in (0, 1)^{N_s \times C}$ controls channel-level updates. We provide more details on this module in the supplementary material.

3.4. Prediction Heads and Training

We follow CUT3R in using the same prediction heads and supervision. Given the refined tokens (p_t, F'_t) , Mem3R predicts local-frame pointmaps and confidence scores, $(\hat{Z}_t^{\text{self}}, C_t^{\text{self}}) = \text{Head}_{\text{self}}(F'_t)$, world-frame pointmaps and confidence scores, $(\hat{Z}_t^{\text{world}}, C_t^{\text{world}}) = \text{Head}_{\text{world}}(p_t, F'_t)$, and the 6-DoF camera pose $\hat{P}_t = \text{Head}_{\text{pose}}(p_t)$.

For training, we use the same objectives as CUT3R. The local- and world-frame pointmaps are supervised with the confidence-aware 3D regression loss in CUT3R and MAST3R [27]:

$$\mathcal{L}_{3\text{D}} = \sum_{f \in \{\text{self}, \text{world}\}} \mathcal{L}_{\text{conf}}(\hat{Z}^f, Z^f, C^f). \quad (11)$$

$$\mathcal{L}_{\text{conf}}(\hat{Z}, Z, C) = \sum_{(\hat{z}, z, c) \in (\hat{Z}, Z, C)} \left(c \cdot \left\| \frac{\hat{z}}{\hat{s}} - \frac{z}{s} \right\|_2 - \beta \log c \right), \quad (12)$$

where $\hat{s}$ and s are scale normalization factors for $\hat{Z}$ and Z , respectively.

Camera pose is supervised by $\mathcal{L}_{\text{pose}} = \sum_{t=1}^N (\|\hat{q}_t - q_t\|_2 + \|\frac{\hat{\tau}_t}{\hat{s}} - \frac{\tau_t}{s}\|_2)$, and, when raymap

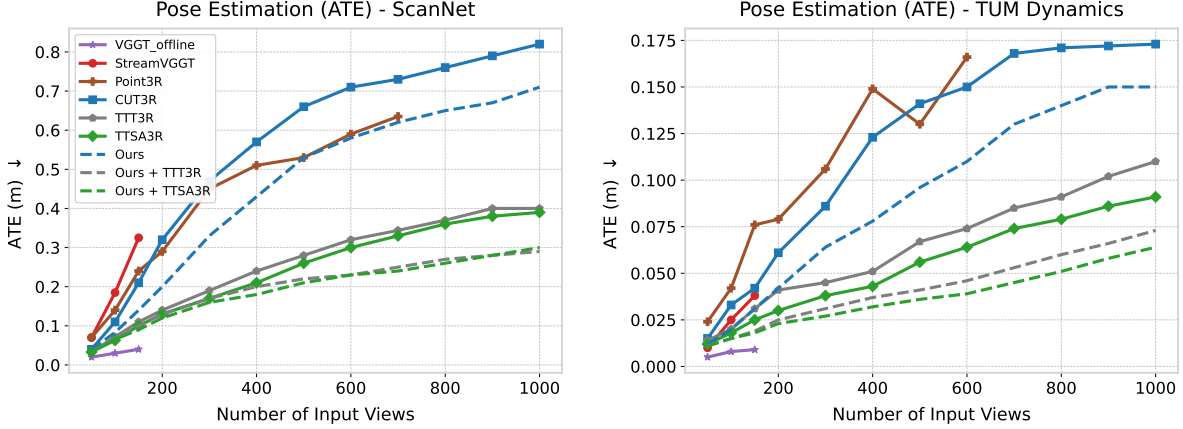


Figure 3. Quantitative evaluation of camera pose estimation from the ScanNet dataset (left) and the TUM Dynamics dataset (right). Mem3R achieves significant improvement compared with CUT3R. Moreover, when combined with the training-free, plug-and-play state update strategies TTT3R or TTSA3R, Mem3R further yields better results compared to applying these strategies to CUT3R.

inputs are available, we additionally use the RGB reconstruction loss $\mathcal{L}_{\text{rgb}} = \|\hat{I}_r - I_r\|_2^2$. The full objective is $\mathcal{L} = \mathcal{L}_{\text{3D}} + \mathcal{L}_{\text{pose}} + \mathbb{1}_{\text{raymap}} \mathcal{L}_{\text{rgb}}$, where $\mathbb{1}_{\text{raymap}}$ indicates whether the input is a raymap.

To ensure that the gains can be fairly attributed to our architectural design, we train Mem3R using the same 26-dataset mixture and training configuration as the final training stage of CUT3R, with input sequences ranging from 4 to 64 views. The training data cover diverse real and synthetic scenes, including CO3Dv2 [36], ARKitScenes [3], MegaDepth [28], MapFree [1], DL3DV [31], and Hyper-sim [37]. The full dataset list is provided in the supplementary material. We initialize the encoder, decoders, and prediction heads from pre-trained 512-DPT CUT3R weights and fine-tune all modules except the encoder. Optimization is performed with AdamW [33], using an initial learning rate of 1×10^{-6} , linear warmup, and cosine decay. Training takes approximately 10 hours on 32 NVIDIA H100 GPUs with a batch size of 8 per GPU.

4. Experiment

We evaluate our Mem3R on a variety of tasks, including camera pose estimation (Sec. 4.1), video depth estimation (Sec. 4.2), and 3D reconstruction (Sec. 4.3). We primarily focus on streaming 3D reconstruction models, specifically Point3R [55], StreamVGGT [67], and CUT3R [52]. Among these, we treat the RNN-based CUT3R as our closest baseline to demonstrate that Mem3R achieves superior performance on long sequences (≥ 200 frames). Furthermore, we show that Mem3R is fully compatible with existing training-free state update mechanisms developed for CUT3R, such as TTT3R [10] and TTSA3R [66]. Quantitative accuracy results are obtained using NVIDIA A100

40GB GPUs. Qualitative visualizations and efficiency measurements are conducted separately on a workstation equipped with a single NVIDIA RTX PRO 6000 Blackwell Max-Q GPU.

4.1. Camera Pose Estimation

Following [10, 66], we evaluate camera pose accuracy on the TUM Dynamics [44] and ScanNet [13] datasets using sequences ranging from 50 to 1,000 frames. We adopt the standard Absolute Trajectory Error (ATE) metric, computed after Sim(3) alignment [48] between the estimated and ground-truth trajectories. As illustrated in Fig. 3, Mem3R consistently outperforms CUT3R and Point3R. Furthermore, integrating TTT3R and TTSA3R into Mem3R yields significantly lower pose errors, particularly on sequences exceeding 400 streaming frames. In contrast, VGGT and StreamVGGT, both based on full attention, are prone to memory exhaustion for such long sequences. Qualitative comparisons in Fig. 4 demonstrate that Mem3R, when equipped with TTT3R and TTSA3R, achieves superior tracking accuracy compared to CUT3R under identical state-update strategies.

Notably, from Fig. 3, Mem3R with TTT3R provides a 39% relative reduction in ATE at the 500-frame mark on the TUM dynamics dataset compared with the base TTT3R model. More results including short sequences are provided in the supplementary material.

4.2. Video Depth Estimation

As in [52, 62], we evaluate long-sequence video depth estimation on the KITTI [19] and Bonn [34] datasets. These datasets span 300 to 500 frames and encompass a diverse range of dynamic, static, indoor, and outdoor scenes. We report both scale-invariant and metric-scale results using the

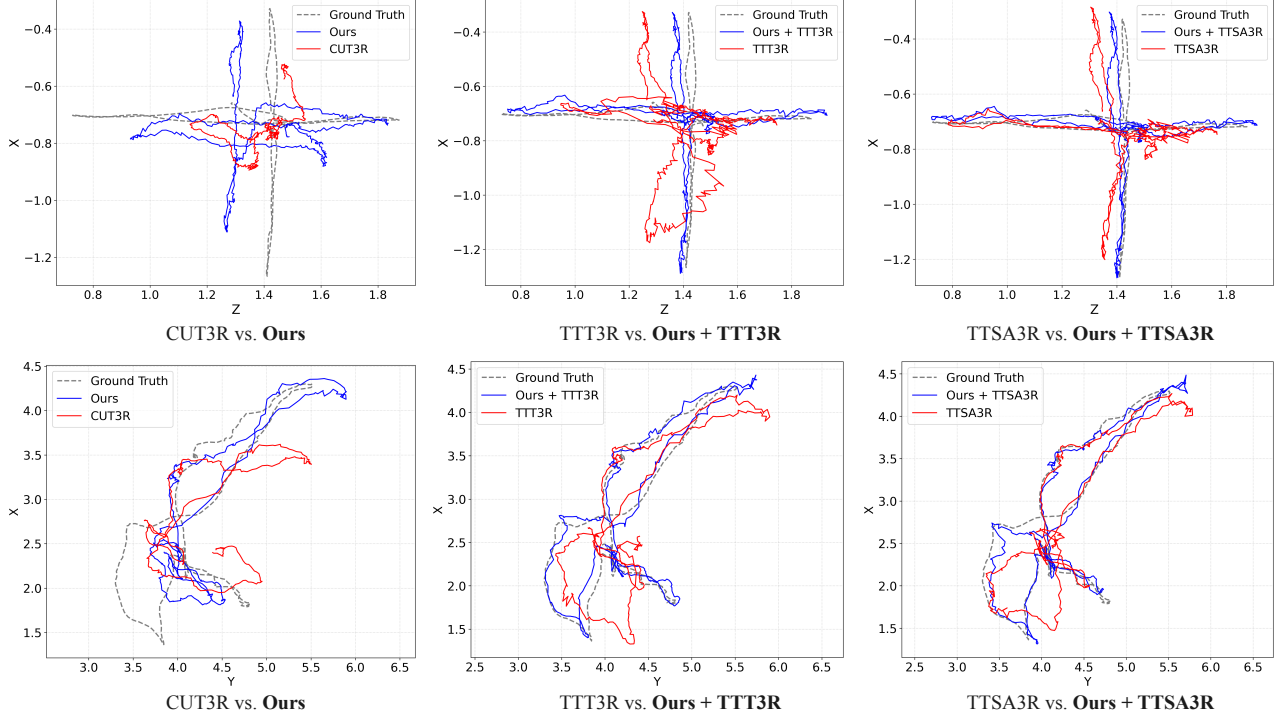


Figure 4. Qualitative visualization of predicted camera trajectories on the TUM Dynamics dataset (top row) and the ScanNet dataset (bottom row). Each row is evaluated on the same sequence. Our Mem3R achieves significant improvement compared with CUT3R. Moreover, when combined with TTT3R or TTSA3R, Mem3R further reduces drift compared to applying these strategies to CUT3R.

Table 1. Quantitative evaluation of video depth estimation on long sequences from the KITTI dataset. **Bold** indicates the best result per column within each depth category, and green indicates improved or maintained performance of our model over its respective base counterpart.

Method	300 frames		350 frames		400 frames		450 frames		500 frames	
	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑
Metric Depth										
CUT3R	0.134	83.8	0.140	82.7	0.144	82.1	0.149	81.1	0.152	80.4
Ours	0.119	88.4	0.123	87.4	0.126	87.1	0.129	86.8	0.132	86.0
TTT3R	0.117	88.5	0.122	87.7	0.125	87.3	0.130	86.6	0.132	86.5
Ours + TTT3R	0.112	89.5	0.115	88.9	0.119	88.6	0.122	88.2	0.125	87.5
TTSA3R	0.117	88.8	0.121	88.0	0.126	87.5	0.131	86.7	0.133	86.5
Ours + TTSA3R	0.116	89.0	0.118	88.5	0.122	88.2	0.125	87.7	0.129	87.0
Scale-Invariant Depth										
CUT3R	0.122	87.4	0.125	87.0	0.126	87.1	0.129	86.7	0.130	86.5
Ours	0.107	90.3	0.110	89.6	0.110	89.4	0.111	89.2	0.113	89.1
TTT3R	0.112	90.2	0.114	89.6	0.115	89.5	0.118	89.1	0.119	89.1
Ours + TTT3R	0.104	91.3	0.106	90.8	0.107	90.5	0.109	90.1	0.110	89.9
TTSA3R	0.109	90.9	0.111	90.2	0.113	89.9	0.116	89.5	0.117	89.4
Ours + TTSA3R	0.103	92.0	0.104	91.5	0.105	91.2	0.108	90.7	0.109	90.5

Absolute Relative error (Abs Rel) and the $\delta_{1.25}$ threshold (the percentage of pixels where $\max(\frac{d_{gt}}{d_{est}}, \frac{d_{est}}{d_{gt}}) < 1.25$).

Table 2. Quantitative evaluation on the Bonn dataset. **Bold** indicates the absolute best result per column within each depth category; green indicates improved or maintained performance over its direct baseline.

Method	300 frames		350 frames		400 frames		450 frames		500 frames	
	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑	Abs Rel ↓	$\delta_{1.25}$ ↑
Metric Depth										
CUT3R	0.106	88.8	0.107	88.8	0.106	89.5	0.101	90.2	0.10	90.6
Ours	0.105	89.7	0.107	89.6	0.105	90.2	0.101	90.8	0.10	91.2
TTT3R	0.108	90.2	0.105	90.8	0.103	91.3	0.102	91.8	0.10	92.1
Ours + TTT3R	0.105	91.4	0.101	91.9	0.100	92.3	0.099	92.6	0.098	92.9
TTSA3R	0.103	91.7	0.101	92.1	0.100	92.5	0.099	92.7	0.098	93.0
Ours + TTSA3R	0.102	92.5	0.099	92.8	0.098	93.2	0.096	93.4	0.094	93.7
Scale-Invariant Depth										
CUT3R	0.089	93.8	0.091	93.1	0.090	93.4	0.086	93.6	0.084	93.8
Ours	0.085	94.6	0.087	94.0	0.087	94.1	0.085	94.4	0.083	94.5
TTT3R	0.079	94.9	0.078	95.0	0.078	95.1	0.077	95.2	0.076	95.3
Ours + TTT3R	0.077	94.9	0.076	95.0	0.076	95.1	0.076	95.2	0.075	95.3
TTSA3R	0.078	95.1	0.077	95.1	0.077	95.2	0.077	95.3	0.076	95.4
Ours + TTSA3R	0.076	95.0	0.074	95.0	0.074	95.1	0.074	95.2	0.073	95.3

As demonstrated in Tabs. 1 and 2, Mem3R significantly outperforms CUT3R. Integrating Mem3R with TTT3R and TTSA3R further yields performance gains across most scenarios. Furthermore, we evaluate short-sequence performance (50 streaming frames) on the Sintel [8] dataset in the supplementary material.

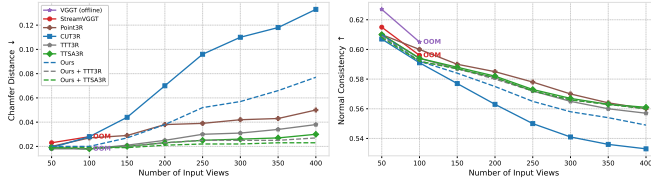


Figure 5. Quantitative evaluation of 3D Reconstruction on 7-Scenes. OOM denotes out-of-memory.

4.3. 3D Reconstruction

We evaluate our multiview reconstruction framework on the 7-Scenes [43] dataset over sequences of 50 to 400 frames, using Chamfer Distance (CD) and Normal Consistency (NC) as the primary metrics. Following TTT3R [10], CD is defined as the mean of accuracy and completeness, where accuracy measures the Euclidean distance from reconstructed points to the nearest ground-truth points and completeness is defined conversely. As shown in Fig. 5, Mem3R significantly outperforms CUT3R. Moreover, when combined with TTT3R or TTSA3R, Mem3R consistently achieves lower CD than Point3R and CUT3R under the same state-update strategy, further validating the

effectiveness of our architecture. Qualitative results in Figs. 1 and 6 show that Mem3R produces more accurate and better-aligned 3D reconstructions, with further improvements when combined with TTT3R or TTSA3R. For example, in Fig. 1, Ours + TTT3R reconstructs the staircase with more accurate step spacing and clearer handrails. Additional visualizations and results on NRGBD [2] are reported in the supplementary material.

4.4. Model Efficiency

Mem3R maintains constant GPU memory complexity with respect to sequence length. As shown in Tab. 3, Mem3R matches the throughput of CUT3R and its training-free state update variants while reducing GPU memory usage by 7% and parameter count by 19% relative to CUT3R. All evaluations are conducted on a workstation with a single NVIDIA RTX PRO 6000 Blackwell Max-Q GPU using 512×384 images from 7-Scenes.

Table 3. Efficiency comparison of runtime (fps), GPU memory usage (MiB), and parameter count. Green indicates that our model matches or outperforms its corresponding base model.

Method	Runtime (fps) ↑	Memory (MiB) ↓	Params ↓
CUT3R	26	7930	793M
Ours	26	7340	644M
TTT3R	25	8364	793M
Ours + TTT3R	25	7774	644M
TTSA3R	25	8786	793M
Ours + TTSA3R	25	8208	644M

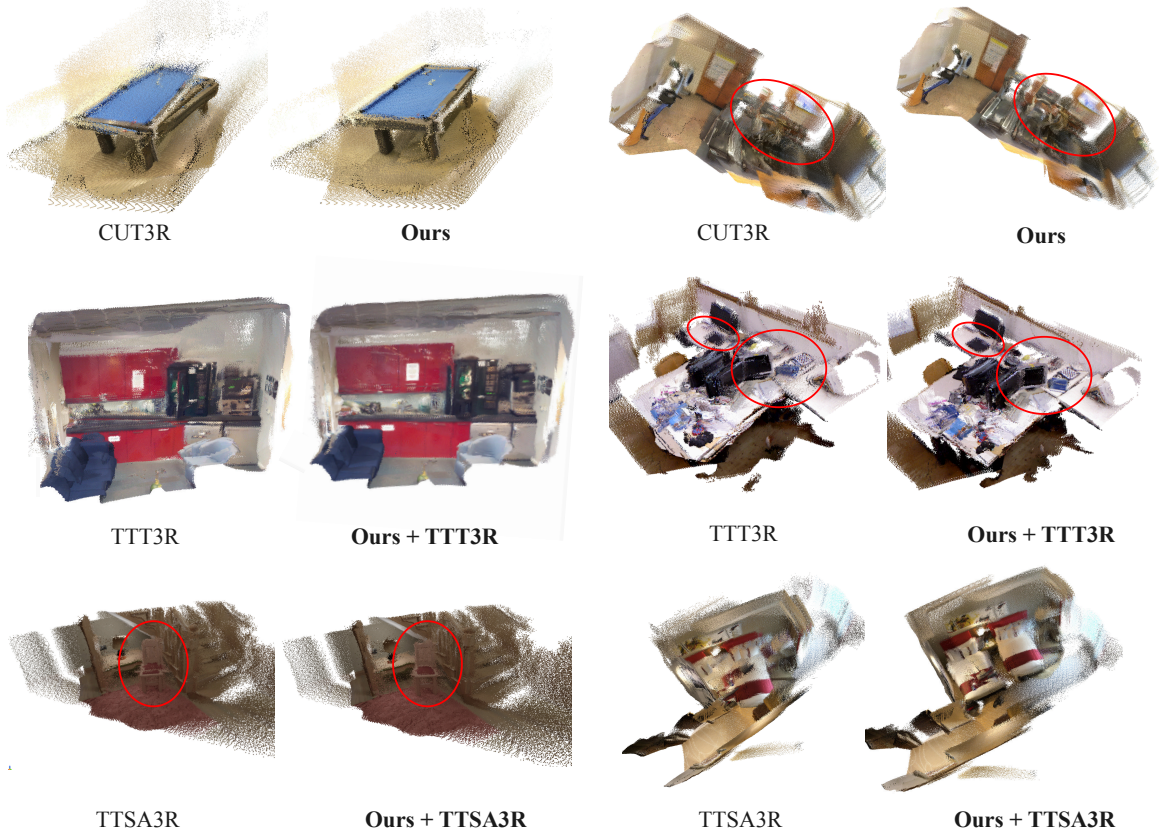


Figure 6. Qualitative 3D reconstruction results on long sequences. Red circles mark some representative regions where the difference is particularly clear.

Table 4. Ablation study of Mem3R. Both variants are independently retrained after removing the corresponding module. 1. w/o. channel-wise gate replaces CUT3R’s original 152M-parameter Transformer-based pose LocalMemory with our implicit fast-weight pose memory, while removing the proposed channel-wise state update mechanism. 2. w/o. fast weight uses CUT3R’s original 152M-parameter Transformer-based pose LocalMemory while retaining the proposed channel-wise update mechanism.

TUM Dynamics: ATE (m) ↓						
Method	#frames					
	200	400	600	800	1000	
TTT3R	0.041	0.051	0.074	0.091	0.110	
Ours + TTT3R (w/o. gate)	0.031	0.044	0.059	0.090	0.100	
Ours + TTT3R (w/o. fast weight)	0.027	0.039	0.054	0.067	0.078	
Ours + TTT3R	0.025	0.037	0.046	0.060	0.073	
KITTI: video depth						
Method	300 frames		400 frames		500 frames	
	Abs Rel ↓	$\delta_{1.25} \uparrow$	Abs Rel ↓	$\delta_{1.25} \uparrow$	Abs Rel ↓	$\delta_{1.25} \uparrow$
TTT3R	0.11	90.2	0.12	89.5	0.12	89.1
Ours + TTT3R (w/o. gate)	0.11	91.0	0.11	90.0	0.11	89.5
Ours + TTT3R (w/o. fast weight)	0.11	91.0	0.11	90.0	0.12	89.5
Ours + TTT3R	0.10	91.3	0.11	90.5	0.11	89.9

4.5. Ablation Study

To evaluate the contributions of the components introduced in Secs. 3.2 and 3.3, we perform ablation studies on long-

sequence camera pose estimation on the TUM Dynamics dataset and video depth estimation on the KITTI dataset, using TTT3R as the baseline. As shown in Tab. 4, both the fast-weight memory and the channel-wise state gate help mitigate temporal forgetting in streaming 3D perception, leading to better accuracy than the original TTT3R.

5. Conclusion

Mem3R is an RNN-based hybrid memory model for streaming 3D reconstruction that decouples camera tracking from geometric mapping, improving long-term consistency and long-sequence performance while preserving recurrent streaming efficiency. Compared with CUT3R, Mem3R is also more parameter-efficient, reducing the model size from 793M to 644M parameters. Despite being more compact, it significantly lowers the Absolute Trajectory Error (ATE) on the challenging 500-1000 frame benchmark of the TUM Dynamics and ScanNet datasets. The resulting improvements also extend to downstream tasks such as 3D reconstruction and video depth estimation. These results establish hybrid memory as an effective design for accurate and efficient long-horizon 3D perception.

Acknowledgments

We would like to thank Guangyao Zhai for the valuable discussion during the initial stages of this project, and Haian Jin for the assistance with the training setup. We thank the members of Android XR, Google for their help and support, especially Xiao Yuan, Luke Jia and Chao Guo.

References

- [1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *European Conference on Computer Vision*, pages 690–708. Springer, 2022. 5
- [2] 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*, pages 6290–6301, 2022. 7
- [3] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, and Elad Shulman. ARK-itscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 5
- [4] Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2024. 2, 3
- [5] Ali Behrouz, Zeman Li, Praneeth Kacham, Majid Daliri, Yuan Deng, Peilin Zhong, Meisam Razaviyayn, and Vahab Mirrokni. Atlas: Learning to optimally memorize the context at test time. *arXiv preprint arXiv:2505.23735*, 2025.
- [6] Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. Nested learning: The illusion of deep learning architectures. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [7] Ali Behrouz, Meisam Razaviyayn, Peilin Zhong, and Vahab Mirrokni. It’s all connected: A journey through test-time memorization, attentional bias, retention, and online optimization. In *The Fourteenth International Conference on Learning Representations*, 2026. 3
- [8] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012. 7
- [9] Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Easi3r: Estimating disentangled motion from dust3r without training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9158–9168, 2025. 2
- [10] Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. TTT3r: 3d reconstruction as test-time training. In *The Fourteenth International Conference on Learning Representations*, 2026. 2, 5, 7
- [11] Zequn Chen, Jiezhi Yang, and Heng Yang. Pref3r: Pose-free feed-forward 3d gaussian splatting from variable-length image sequence. *arXiv preprint arXiv:2411.16877*, 2024. 2
- [12] 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*, pages 5273–5284, 2025. 2
- [13] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, pages 2432–2443, 2017. 5
- [14] Jiacheng Dong, Huan Li, Sicheng Zhou, Wenhao Hu, Weili Xu, and Yan Wang. Memix: Writing less, remembering more for streaming 3d reconstruction. *arXiv preprint arXiv:2603.15330*, 2026. 2
- [15] Sven Elflein, Ruilong Li, Sérgio Agostinho, Zan Gojcic, Laura Leal-Taixé, Qunjie Zhou, and Aljosa Osep. Vgg-t³: Offline feed-forward 3d reconstruction at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 3
- [16] Zhiwen Fan, Jian Zhang, Wenyan Cong, Peihao Wang, Renjie Li, Kairun Wen, Shijie Zhou, Achuta Kadambi, Zhangyang Wang, Danfei Xu, et al. Large spatial model: End-to-end unposed images to semantic 3d. *Advances in neural information processing systems*, 37:40212–40229, 2024. 2
- [17] Xianze Fang, Jingnan Gao, Zhe Wang, Zhuo Chen, Xingyu Ren, Jiangjing Lyu, Qiao-Mu Ren, Zhonglei Yang, Xiaokang Yang, Yichao Yan, and chengfei lv. Dens3r: A foundation model for 3d geometry prediction. In *The Fourteenth International Conference on Learning Representations*, 2026. 2
- [18] Jingnan Gao, Zhe Wang, Xianze Fang, Xingyu Ren, Zhuo Chen, Shengqi Liu, Yuhao Cheng, Jiangjing Lyu, Xiaokang Yang, and Yichao Yan. More: 3d visual geometry reconstruction meets mixture-of-experts. *arXiv preprint arXiv:2510.27234*, 2025. 2
- [19] 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. 5
- [20] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First conference on language modeling*, 2024. 3
- [21] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982. 3
- [22] Jiaxi Hu, Yongqi Pan, Jusen Du, Disen Lan, Xiaqiang Tang, Qingsong Wen, Yuxuan Liang, and Weigao Sun. Comba: Improving bilinear rnns with closed-loop control. *arXiv preprint arXiv:2506.02475*, 2025. 3
- [23] Rohit Jayanti, Swayam Agrawal, Vansh Garg, Siddharth Tourani, Muhammad Haris Khan, Sourav Garg, and Madhava Krishna. Segmast3r: Geometry grounded segment matching. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2
- [24] Haian Jin, Rundui Wu, Tianyuan Zhang, Ruiqi Gao, Jonathan T. Barron, Noah Snaveley, and Aleksander Holyn-

- ski. ZipMap: Linear-time stateful 3d reconstruction via test-time training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. 3
- [25] Nikhil 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. *arXiv preprint arXiv:2509.13414*, 2025. 2
- [26] Yushi LAN, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Bo Dai, Shuai Yang, Chen Change Loy, and Xingang Pan. SStream3r: Scalable sequential 3d reconstruction with causal transformer. In *The Fourteenth International Conference on Learning Representations*, 2026. 2
- [27] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024. 2, 4
- [28] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018. 5
- [29] Zeman Li, Ali Behrouz, Yuan Deng, Peilin Zhong, Praneeth Kacham, Mahdi Karami, Meisam Razaviyayn, and Vahab Mirrokni. Tnt: Improving chunkwise training for test-time memorization. *arXiv preprint arXiv:2511.07343*, 2025. 3
- [30] Zizun Li, Jianjun Zhou, Yifan Wang, Haoyu Guo, Wenzheng Chang, Yang Zhou, Haoyi Zhu, Junyi Chen, Chunhua Shen, and Tong He. Wint3r: Window-based streaming reconstruction with camera token pool. In *The Fourteenth International Conference on Learning Representations*, 2026. 2
- [31] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. D13dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 5
- [32] Changkun Liu, Bin Tan, Zeran Ke, Shangzhan Zhang, Jiachen Liu, Ming Qian, Nan Xue, Yujun Shen, and Tristan Braud. Plana3r: Zero-shot metric planar 3d reconstruction via feed-forward planar splatting. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2
- [33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5
- [34] Emanuele Palazzolo, Jens Behley, Philipp Lottes, Philippe Giguere, and Cyrill Stachniss. Refusion: 3d reconstruction in dynamic environments for rgb-d cameras exploiting residuals. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7855–7862. IEEE, 2019. 5
- [35] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, et al. Rwkv: Reinventing rnns for the transformer era. In *Findings of the association for computational linguistics: EMNLP 2023*, pages 14048–14077, 2023. 3
- [36] 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*, pages 10901–10911, 2021. 5
- [37] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021. 5
- [38] Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In *International conference on machine learning*, pages 9355–9366. PMLR, 2021. 3
- [39] Jürgen Schmidhuber. Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, 4(1):131–139, 1992.
- [40] Jürgen Schmidhuber. Reducing the ratio between learning complexity and number of time varying variables in fully recurrent nets. In *ICANN’93: Proceedings of the International Conference on Artificial Neural Networks Amsterdam, The Netherlands 13–16 September 1993* 3, pages 460–463. Springer, 1993. 3
- [41] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 4
- [42] You Shen, Zhipeng Zhang, Yansong Qu, Xiawu Zheng, Jiayi Ji, Shengchuan Zhang, and Liujuan Cao. Fastvgg: Training-free acceleration of visual geometry transformer. *arXiv preprint arXiv:2509.02560*, 2025. 2
- [43] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2930–2937, 2013. 7
- [44] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 5
- [45] Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, et al. Learning to (learn at test time): Rnns with expressive hidden states. *arXiv preprint arXiv:2407.04620*, 2024. 3
- [46] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5283–5293, 2025. 2
- [47] Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiayi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, et al. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*, 2025. 2, 3

- [48] 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. [5](#)
- [49] Chen Wang, Hao Tan, Wang Yifan, Zhiqin Chen, Yuheng Liu, Kalyan Sunkavalli, Sai Bi, Lingjie Liu, and Yiwei Hu. ttllm: Test-time training for long context and autoregressive 3d reconstruction. *arXiv preprint arXiv:2602.20160*, 2026. [3](#)
- [50] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. In *2025 International Conference on 3D Vision (3DV)*, pages 78–89. IEEE, 2025. [2](#)
- [51] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vgg: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025. [2](#)
- [52] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10510–10522, 2025. [2](#), [5](#)
- [53] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jérôme Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, pages 20697–20709, 2024. [2](#)
- [54] 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 *The Fourteenth International Conference on Learning Representations*, 2026. [2](#)
- [55] Yuqi Wu, Wenzhao Zheng, Jie Zhou, and Jiwen Lu. Point3r: Streaming 3d reconstruction with explicit spatial pointer memory. *arXiv preprint arXiv:2507.02863*, 2025. [2](#), [5](#)
- [56] Tao Xie, Peishan Yang, Yudong Jin, Yingfeng Cai, Wei Yin, Weiqiang Ren, Qian Zhang, Wei Hua, Sida Peng, Xiaoyang Guo, and Xiaowei Zhou. Scal3r: Scalable test-time training for large-scale 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21760–21771, 2026. [3](#)
- [57] Qi Xu, Dongxu Wei, Lingzhe Zhao, Wenpu Li, Zhangchi Huang, Shunping Ji, and Peidong Liu. Siu3r: Simultaneous scene understanding and 3d reconstruction beyond feature alignment. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. [2](#)
- [58] Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21924–21935, 2025. [2](#)
- [59] Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. *arXiv preprint arXiv:2412.06464*, 2024. [3](#)
- [60] Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. *Advances in neural information processing systems*, 37:115491–115522, 2024. [3](#)
- [61] Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. Infinitvggt: Visual geometry grounded transformer for endless streams. *arXiv preprint arXiv:2601.02281*, 2026. [2](#)
- [62] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3r: A simple approach for estimating geometry in the presence of motion. In *The Thirteenth International Conference on Learning Representations*, 2025. [2](#), [5](#)
- [63] 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. [3](#)
- [64] 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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21936–21947, 2025. [2](#)
- [65] Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T Freeman, and Hao Tan. Test-time training done right. *arXiv preprint arXiv:2505.23884*, 2025. [2](#), [3](#)
- [66] 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. [2](#), [5](#)
- [67] Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Streaming 4d visual geometry transformer. *arXiv preprint arXiv:2507.11539*, 2025. [2](#), [5](#)