Geometric Context Transformer for Streaming 3D Reconstruction

Lin-Zhuo Chen[1,2,3⋆‡], Jian Gao[1,2⋆‡], Shangzhan Zhang[1,4⋆‡], Yihang Chen[1,3‡], Nan Xue[1], Jianyuan Wang[4], Christian Rupprecht[4], Xun Cao[2], Xing Zhu[1], Yujun Shen[1], Yao Yao[2†], and Yinghao Xu[3,1†]

[1] Robbyant, Ant Group, China
[2] Nanjing University, China
[3] Hong Kong University of Science and Technology, Hong Kong, China
[4] University of Oxford, United Kingdom

Abstract. Streaming 3D reconstruction aims to recover 3D information, such as camera poses and point clouds, from a video stream, which necessitates geometric accuracy, temporal consistency, and computational efficiency. Motivated by the principles of Simultaneous Localization and Mapping (SLAM), we introduce GCT, *geometric context transformer*, a feed-forward 3D foundation model for reconstructing scenes from streaming data. A defining aspect of GCT lies in its carefully designed attention mechanism, which integrates an anchor context, a pose-reference window, and a trajectory memory to address coordinate grounding, dense geometric cues, and long-range drift reduction, respectively. This design keeps the streaming state compact while retaining rich geometric context, enabling stable efficient inference at around 20 FPS on 518×378 resolution inputs over long sequences exceeding 10,000 frames. Extensive evaluations across a variety of benchmarks demonstrate that our approach achieves superior performance compared to both existing streaming and iterative optimization-based approaches.

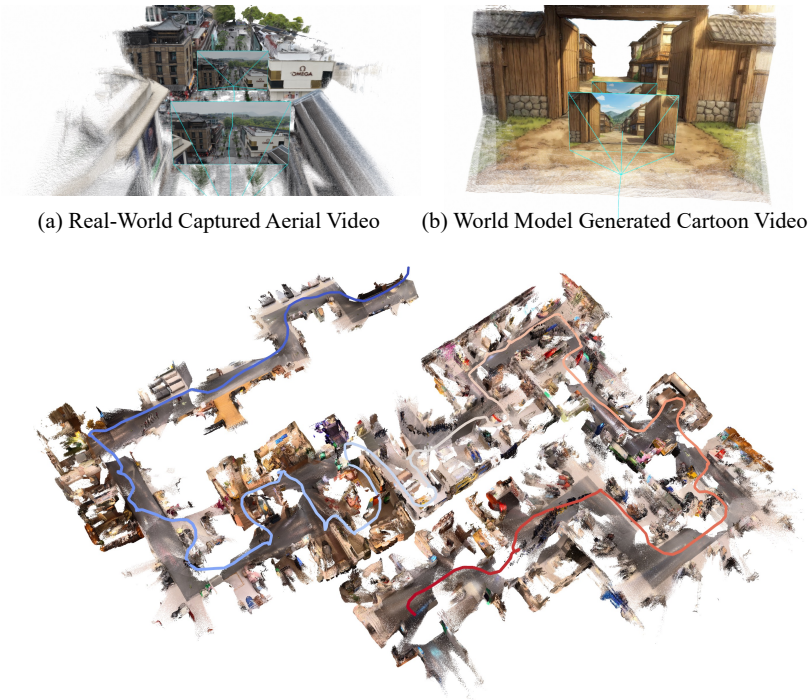
Keywords: 3D Foundation Models · Streaming Reconstruction

1 Introduction

We perceive the world through a continuous stream of visual input, yet our spatial memory is not a faithful recording of every moment: it is sparse, structured, and efficient. Rather than retaining every observation, human spatial cognition selectively preserves only the most essential cues, enabling coherent navigation through large-scale environments over extended periods. Can we build machines that operate similarly, performing streaming 3D reconstruction from continuous visual input, selectively and efficiently?

In recent years, 3D foundation models have advanced rapidly, with methods such as VGGT [72] and Depth Anything 3 [37] directly predicting camera poses, depth maps, and dense point maps from multiple images in a single feed-forward

⋆ Equal contribution. † Corresponding authors. ‡ Work done during internship.



(a) Real-World Captured Aerial Video (b) World Model Generated Cartoon Video

(c) Long-sequence Multi-Room Traversal Video

Fig. 1. Visualization results of GCT on varying scenes: Given a continuous video stream, GCT performs *efficient* **streaming** 3D reconstruction with accurate camera pose estimation and high-quality point cloud reconstruction. Examples include Real-World Captured Aerial Video (*top-left*), World Model Generated Cartoon Video (*top-right*) and Long-sequence Multi-Room Traversal Video (*bottom*). More details of these results can be found in supplementary materials.

pass. However, these successes are largely confined to offline settings, where the full image set is available and processed globally. Recent efforts [29, 34, 76, 98] have begun to adapt these capabilities to streaming reconstruction, but existing approaches still exhibit limited robustness to long sequences and complex scenes. The core difficulty is selective context management: balancing rich geometric context for long-term consistency with a compact state for efficient inference.

Existing methods adopt different strategies to manage streaming context, each involving distinct trade-offs. CUT3R [76] maintains a persistent recurrent-style state, but its aggressive compression can lead to state forgetting and weak retention of essential geometric priors. In contrast, StreamVGGT [98] and Stream3R [29] adopt causal attention and caching, yet retain near-complete history without explicit selection, mixing useful geometry with redundancy and causing memory and computation to grow rapidly. A third line of work, such as VGGT-SLAM [42] and MASt3R-SLAM [46], integrates learned 3D models with classical SLAM backends. While these methods incorporate structured

context management through keyframe selection and pose-graph maintenance, such selection relies on hand-crafted heuristics rather than learned priors, and iterative optimization further limits real-time applicability. These observations point to a key design principle: the streaming state should selectively retain *what* matters most, not merely how much, and this selection should be grounded in geometric priors yet learned end-to-end from data.

To this end, we introduce GCT, a streaming foundation model built around Geometric Context Attention (GCA) that realizes this principle within a unified attention framework. The design draws on a key insight from classical SLAM systems: robust real-time reconstruction requires maintaining distinct types of spatial context—a reference frame for coordinate grounding, a local window for dense local geometry estimation, and a global map for long-range drift reduction. Accordingly, GCA explicitly maintains three complementary types of context: an *anchor context* for coordinate and scale grounding, a local *pose-reference window* that retains dense visual features from recent frames for accurate local geometry estimation, and a *trajectory memory* that compresses the full observation history into compact per-frame tokens for global consistency. While the context structure is motivated by classical reconstruction principles, GCA replaces hand-crafted optimization with end-to-end learned attention that adaptively weights, encodes, and compresses information within each context type. This structured yet learned representation ensures stable and efficient inference even over arbitrarily long sequences, with nearly constant memory and computation per frame, as context beyond the local window is compressed into compact per-frame tokens.

To scale up training effectively, we employ a progressive training strategy combined with context parallelism [22], along with a relative loss formulation that facilitates stable optimization on long sequences. This allows GCT to learn efficiently from diverse, large-scale 3D datasets. As shown in Fig. 1, GCT produces high-quality streaming reconstructions with accurate camera poses across diverse scenes. We evaluate GCT on comprehensive benchmarks, including Oxford Spires, 7-Scenes, Tanks and Temples, ETH3D, and show consistent improvements over existing streaming methods in both camera pose estimation and dense 3D reconstruction quality.

Our contributions are summarized as follows:

- We introduce GCT, a streaming 3D foundation model built around Geometric Context Attention (GCA), which maintains three complementary context types—anchor, pose-reference window, and trajectory memory—for efficient and consistent long-sequence streaming inference.
- We propose an efficient training recipe based on progressive training and context parallelism with a relative loss formulation for stable long-sequence optimization.
- We demonstrate that GCT achieves state-of-the-art performance on multiple benchmarks (Oxford Spires, Tanks and Temples, ETH3D, and 7-Scenes), significantly outperforming existing streaming approaches in reconstruction quality and inference speed.

2 Related Work

Traditional 3D Reconstruction. Traditional 3D reconstruction methods mainly include Structure-from-Motion (SfM) [48,55,61], Simultaneous Localization and Mapping (SLAM) [5,44,45], and Multi-View Stereo (MVS) [16,56,87]. SfM and SLAM recover camera poses and scene geometry from multi-view observations, where SfM typically operates offline on unordered image collections, while SLAM processes video streams online. These systems are usually complex and highly modular, typically centered around optimization-based bundle adjustment for camera pose estimation. In contrast, MVS focuses on dense reconstruction given known camera poses. Over the past decade, many works have explored replacing individual components of these pipelines with deep learning modules, particularly for feature extraction [13] and matching [54, 63]. More recently, several approaches attempt to implement SfM, SLAM, or MVS in an end-to-end manner, such as VGGTSfM [73], DROID-SLAM [67], and MVSNet [87, 88].

3D Foundation Model. DUSt3R [77] represents a paradigm shift in feed-forward 3D reconstruction. Given a collection of unposed images, DUSt3R directly regresses a dense 3D reconstruction of the scene without explicit geometric modeling. However, DUSt3R [77] only supports two-view input and requires aligning all results via optimization for additional views. To support more than two views and improve reconstruction quality, VGGT [72] uses an advanced transformer architecture that includes cross-view attention layers, achieving state-of-the-art performance on standard benchmarks. Crucially, VGGT demonstrates that leveraging large-scale data together with powerful model architectures can significantly improve reconstruction quality. Building on this foundation, numerous subsequent works have advanced feed-forward reconstruction across various dimensions, including improving reconstruction accuracy [37, 72, 79, 95], enhancing computational efficiency [39, 86], handling dynamic scenes [7, 14, 25, 40, 62, 71, 82, 94], enabling novel view synthesis [9, 19, 24, 30, 36, 60, 65, 75, 84, 85, 90], and incorporating multi-modal inputs [23, 26, 41]. However, these methods are primarily designed for offline processing and do not address the unique challenges of streaming 3D reconstruction, such as maintaining long-term consistency and managing computational resources over extended sequences.

Streaming 3D Reconstruction. Driven by the need for online applications, streaming 3D reconstruction can be broadly categorized into hybrid SLAM-based approaches and end-to-end feed-forward methods. Hybrid methods typically integrate 3D foundation models with traditional SLAM pipelines [12, 42, 46, 83], aiming to leverage the strengths of both paradigms. However, these approaches often rely on hand-crafted components and careful parameter tuning to achieve optimal performance, lacking the benefits of a fully end-to-end learning framework. In contrast, recent feed-forward streaming methods [29,34,70,76,98] extend the offline paradigm to streaming settings by employing Recurrent Neural Network (RNN)-based architectures or by combining caching mechanisms with causal attention. Specifically, CUT3R [76] maintains a persistent state that is updated recurrently via RNN architectures. To mitigate state forgetting, TTT3R [8] adopts a test-time training strategy. Meanwhile, StreamVGGT [98],

Stream3R [29], and Wint3R [34] adapt the more advanced VGGT architecture using causal attention and caching strategies. Despite these advances, existing streaming methods often struggle to maintain performance over long input sequences and in complex environments. Common failure modes include significant trajectory drift, degraded reconstruction accuracy, and prohibitive growth in memory and computational requirements. We attribute these limitations to the absence of a principled mechanism for effectively retaining essential geometric context during the streaming process.

3 Method

In Sec. 3.1, we present an overview of our method and the problem formulation. We then introduce Geometric Context Attention (GCA) (Sec. 3.2), a mechanism specifically designed for streaming 3D reconstruction. Sec. 3.3 describes the network architecture and the training procedure. Finally, Sec. 3.4 presents the inference pipeline for efficient inference.

3.1 Overview

Given a continuous stream of images $\mathcal{I} = \{I_1, I_2, \ldots\}$, GCT processes each new frame I_t upon arrival and estimates its camera pose $\hat{P}_t$ and depth map $\hat{D}_t$ using only the current and previously observed frames $\{I_1, \ldots, I_t\}$, without access to future observations. Building on the architecture of recent feed-forward 3D models [72], we design a streaming variant where each frame is encoded by a ViT backbone and processed through alternating layers of frame-wise attention and Geometric Context Attention (GCA), before task-specific heads predict the camera pose and depth map (see Fig. 3). The key to enabling efficient streaming inference is GCA, which maintains three complementary geometric contexts: anchor, pose-reference window, and trajectory memory, thereby balancing long-term consistency with compact state representation. We detail GCA in Sec. 3.2, the overall architecture and training strategy in Sec. 3.3, and the inference pipeline in Sec. 3.4.

3.2 Geometric Context Attention

The central challenge of streaming 3D reconstruction is managing geometric context: the model must retain sufficient long-range context to ensure global consistency, yet keep its streaming state compact enough for efficient inference. Classical SLAM and SfM systems provide structural insights into this trade-off by decomposing the streaming state into three distinct types of spatial context, each serving a complementary role: a reference frame for coordinate and scale grounding, a local window of recent observations for dense geometry estimation, and a global map for reducing accumulated drift. Drawing on this principle, GCA decomposes the streaming context into three complementary learned attention mechanisms, replacing hand-crafted optimization with end-to-end differentiable

attention: *anchor context*, a local *pose-reference window*, and *trajectory memory*. We describe each below.

Anchor Context. Monocular reconstruction is inherently scale-ambiguous, so a consistent coordinate system and absolute scale must be established before streaming begins. Offline methods such as DUSt3R [77] and VGGT [72] resolve this by normalizing with respect to the *global* point cloud, but this requires access to all frames and is therefore incompatible with causal streaming inference. Instead, we designate the first n images ($n \ll N$) as anchor frames and use them to fix the scale. We apply full attention among these frames. Every frame carries a learnable anchor token, but anchor frames and streaming frames use *different* learned values for this token, so the network can identify and distinguish anchor frames from subsequent streaming frames. After initialization, the anchor tokens and image tokens of the anchor frames are retained in the attention context, and all subsequent frames attend to them as fixed references. We refer to all information carried by these initialization frames collectively as the *anchor context*. During training, we normalize all ground-truth annotations to a canonical scale derived from the anchor frames: we compute $s = \frac{1}{|\bar{\mathcal{X}}^{\text{anchor}}|} \sum_{\mathbf{x} \in \bar{\mathcal{X}}^{\text{anchor}}} \|\mathbf{x}\|_2$ as the mean distance of the ground-truth point cloud $\bar{\mathcal{X}}^{\text{anchor}}$ from the coordinate origin, and divide all ground-truth depths and camera translations by s.

Local Pose-Reference Window. Accurately registering each new frame requires dense visual overlap with nearby observations, context that the distant anchor frames alone cannot provide. To address this, we maintain a sliding window of the k most recent frames during inference, retaining their *full* image tokens. This dense local context provides essential relative pose cues from immediate visual connections, enabling the network to accurately register new frames into the global trajectory. To further encourage geometric consistency within the local window, we apply a relative pose loss between frames in this window, as detailed in Sec. 3.3.

Trajectory Memory. The anchor context and local window together provide a fixed global reference and dense recent observations, but without any record of intermediate frames, pose errors accumulate unchecked over long sequences, causing the estimated trajectory to drift. To mitigate this, we retain a compact trajectory context that summarizes the full observation history. Specifically, for frames that fall outside both the anchor set and the active sliding window, we retain only the camera, anchor, and register tokens (i.e., 6 context tokens per frame) while discarding the memory-intensive image tokens (M tokens per frame). Additionally, we incorporate video temporal positional encodings [69] into the retained tokens to impose temporal ordering on the global trajectory. By maintaining this lightweight yet temporally ordered record of all past observations, the trajectory memory provides long-range cues that help correct accumulated drift and ensure global consistency.

Attention Mask Design. Fig. 2 compares different attention patterns for streaming inference. Global attention (a) attends to all frames but cannot operate in a streaming fashion. Causal attention (b) enables streaming but causes memory and computation to grow linearly with sequence length. Sliding window

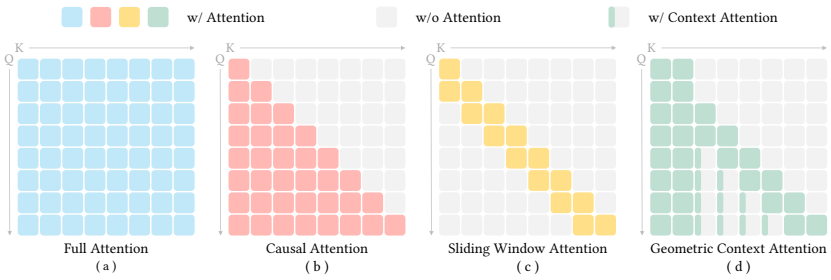


Fig. 2. Comparison of attention masks. Each square block represents one frame's tokens, consisting of a small leading segment of context tokens (camera, anchor, and register tokens) and a larger segment of image tokens. (a) Full attention attends to all frames. (b) Causal attention enables streaming but grows linearly with sequence length. (c) Sliding-window attention bounds cost but loses long-range context. (d) GCA partitions the streaming context into *anchors* (n=2), a *local window* (k=2), and *trajectory memory*, retaining rich long-range context while keeping cost nearly constant as the sequence length increases.

attention (c) bounds compute but sacrifices long-term context. Our GCA (d) combines the anchor context, trajectory memory, and local window into a structured attention mask that retains long-range consistency with bounded per-frame cost.

Complexity Analysis. For a T-frame sequence, the per-frame attention context in GCA comprises n anchor frames with full tokens ($n \cdot (M+6)$), k window frames with full tokens ($k \cdot (M+6)$), and ($T-n-k$) trajectory frames with compact tokens (6 each). Since n and k are fixed constants, the total context simplifies to $(n{+}k){\cdot}M{+}6T$, where the first term is constant and the second grows at a rate of just 6 tokens per frame. In contrast, causal attention retains $T \cdot (M+6) = MT + 6T$ tokens, sharing the same $6T$ term but incurring an additional MT that grows with the full token count. Since each new frame adds (M+6) tokens under causal attention but only 6 under GCA, the per-frame growth rate is reduced by roughly 80× for typical values ($M{\approx}500$). Concretely, with n=3, k=16, and T=10,000, causal attention accumulates $\sim 5{\times}10^6$ tokens, while GCA retains only $\sim 7{\times}10^4$, yielding nearly constant memory and computation per frame.

3.3 Geometric Context Transformer Framework

Architecture. The overall architecture is illustrated in Fig. 3. Each input image is first encoded by a Vision Transformer (ViT) backbone initialized from DINOv2 [47] to produce M image tokens per frame. These image tokens are augmented with a camera token $\mathbf{c} \in \mathbb{R}^C$, four register tokens $\mathbf{r}_j \in \mathbb{R}^C$ ($j = 1, \ldots, 4$), and a learnable anchor token $\mathbf{a} \in \mathbb{R}^C$. Following VGGT [72], the register tokens act as shared scratch tokens that participate in attention but are not decoded by the prediction heads. The augmented tokens are then processed through multiple

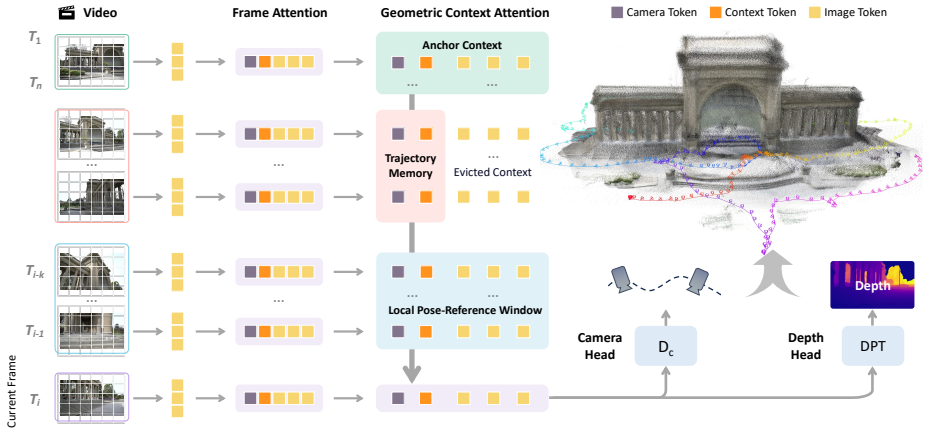


Fig. 3. Pipeline of the proposed GCT. The framework processes the current view T_i relative to an initialization set $[T_1, T_n)$. A DINO backbone extracts image features, which are then refined through alternating layers of Frame Attention and GCA. Within the GCA module, the input view aggregates information from the *Anchor Context*, *Local Pose-Reference Window* $[T_{i-k}, T_i)$, and *Trajectory Memory Context*. Finally, task-specific heads predict camera pose and depth maps, enabling robust, memory-efficient streaming 3D reconstruction for long-range sequences.

alternating layers of Frame Attention and GCA. Frame Attention operates independently within each frame, enabling per-frame feature refinement, while GCA operates across frames according to the structured attention mask described in Sec. 3.2, enabling cross-frame geometric reasoning. Finally, a camera head takes the camera token to predict the absolute camera pose $\hat{P}_t$, and a depth head takes the image tokens to predict the corresponding depth map $\hat{D}_t$.

Loss Function. We train GCT using a composite loss function consisting of depth, absolute pose, and relative pose terms:

$$\mathcal{L} = \lambda_{\text{depth}}\,\mathcal{L}_{\text{depth}} + \lambda_{\text{abs-pose}}\,\mathcal{L}_{\text{abs-pose}} + \lambda_{\text{rel-pose}}\,\mathcal{L}_{\text{rel-pose}}. \tag{1}$$

The depth loss ($\mathcal{L}_{\text{depth}}$) and absolute pose loss ($\mathcal{L}_{\text{abs-pose}}$) follow the definitions in VGGT [72]: $\mathcal{L}_{\text{depth}} = \sum_{i=1}^{N} \left\| \Sigma_i^D \odot (\hat{D}_i - D_i) \right\| + \left\| \Sigma_i^D \odot (\nabla \hat{D}_i - \nabla D_i) \right\| - \alpha \log \Sigma_i^D$, $\mathcal{L}_{\text{abs-pose}} = \sum_{i=1}^{N} \left\| \hat{\mathbf{P}}_i - \mathbf{P}_i \right\|_\epsilon$. Here, Σ_i^D represents the predicted uncertainty map, and $\odot$ denotes element-wise multiplication. Unlike VGGT [72], we supervise the network using camera-to-world transformations rather than world-to-camera ones. In the world-to-camera parameterization, rotation and translation are inherently coupled, making translation estimation highly sensitive to rotation errors, particularly in long sequences. Inspired by π^3 [79], we incorporate a relative pose loss over all frame pairs within the sliding window:

$$\mathcal{L}_{\text{rel-pose}} = \frac{1}{k(k-1)} \sum_{\substack{i \neq j \\ i,j \in \{1,\ldots,k\}}} \big(\mathcal{L}_{\text{rot}}(i,j) + \lambda_{\text{trans}} \mathcal{L}_{\text{trans}}(i,j) \big), \tag{2}$$

where $\mathcal{L}_{\text{rot}}(i,j)$ and $\mathcal{L}_{\text{trans}}(i,j)$ denote the geodesic rotation error and ℓ_1 translation error of the relative pose between frames i and j, respectively. Because the window comprises only already-observed frames, this loss is inherently causal and encourages local trajectory consistency.

Progressive View Training. Training directly on long sequences is challenging: early-stage pose errors propagate along the trajectory and destabilize the loss landscape, leading to slow or divergent optimization. To address this, we adopt a progressive training strategy that starts with short subsequences and gradually increases the number of views over training. This curriculum enables the network to first acquire reliable local geometry estimation from short clips before learning to maintain global consistency across progressively longer trajectories.

Context Parallel. As the number of training views grows, GPU memory becomes the primary bottleneck due to the quadratic cost of cross-frame attention. To address this, we employ the Ulysses [22] context-parallelism strategy, which distributes different views across multiple GPUs to enable parallel attention computation via efficient all-to-all collective communication.

3.4 Inference Pipeline

Inference design. Like autoregressive LLMs, our causal model caches the key-value (KV) states of previously processed frames to avoid redundant recomputation. However, with naive causal attention, the KV cache scales linearly with the number of frames, increasing memory consumption and per-frame latency. GCA addresses this by keeping the per-frame context compact (see Sec. 3.2), but the sliding-window and trajectory-eviction logic still requires frequent cache updates (appending new entries and discarding old ones), which incur overhead due to repeated memory reallocation under a standard contiguous layout. We eliminate this overhead with a paged KV-cache layout [28], in which updates affect only newly appended tokens rather than the entire cached sequence.

We implement the runtime on FlashInfer [91], which provides native support for paged KV-cache management, as well as optimized attention kernels for paged and sparse KV layouts. In the 518×378 setting with video sequences up to 1000 frames and a sliding window of 64 frames, our FlashInfer-based implementation achieves $\sim$20 FPS, compared to $\sim$10.5 FPS for an otherwise identical PyTorch baseline with contiguous KV-cache updates. To support robust long-sequence inference, we select a key frame every m frames to be retained in the KV cache, when the input views are more than our max training views in training.

4 Experiments

In this section, we first compare GCT with the current SOTA methods for camera pose estimation and 3D reconstruction. Then, we conduct ablation studies on different training strategies and design choices.

4.1 Implementation Details

Training Details. We initialize GCT, including the ViT backbone and attention layers, with weights pretrained from DINOv2 [47]. The standard patch size is set to 14 pixels. Training is conducted in two stages. In the first stage, we pretrain GCT on a diverse collection of datasets that include both video and non-video data. At this stage, we replace GCA with standard global attention to train an offline, short-sequence base model, maximizing the utility of available datasets. In the second stage, we load the pretrained weight, replace global attention with our proposed GCA, and fine-tune on long-sequence reconstruction datasets using context parallelism with a dimension of 16. Both stages use 160K iterations with the AdamW optimizer. Training takes approximately two weeks per stage on 64 GPUs. The learning rate is set to 2×10^{-3} for the first stage and 5×10^{-4} for the second. Data augmentation techniques include color jittering, Gaussian blur, and greyscale conversion. Images are resized to a maximum dimension of 518 pixels. During training, the local window size k is randomly sampled from 16 to 64, while at inference we set $k = 64$ and the keyframe interval $m = 1$.
Training Data. We first pretrain GCT on a diverse set of datasets, including BlendedMVS [89], HyperSim [53], MegaDepth [32], Unreal4K [68], WildRGBD [81], TartanAir [78], TartanAirV2 [78], TartanGround [50], PointOdyssey [97], VirtualKITTI [4], Kubric [18], DL3DV [38], MVS-Synth [21], GTA-SFM [74], CO3D [52], SceneRGBD [43], Mapfree [1], Aria Synthetic Environments [49], ADT [49], Objaverse [11], Texverse [96], ScanNet++ [92], ScanNet [10], MatrixCity [31], MidAir [15], and our internal game dataset, following the offline feed-forward training paradigm of VGGT [72]. The number of training views is randomly sampled between 2 and 24. Subsequently, we fine-tune GCT on long-sequence reconstruction datasets, adapting our streaming design and training strategies. Relative to the first stage, we no longer use MegaDepth [32], MVS-Synth [21], GTA-SFM [74], CO3D [52], ADT [49], Objaverse [11], and Texverse [96], while additionally introducing Waymo [64], KITTI-360 [35], Gibson [80], Matterport3D [6], and HM3D [51]. During fine-tuning, the number of training views is progressively increased linearly from 24 to 320.

Our method consistently outperforms state-of-the-art online baselines across all evaluated datasets. On the challenging ETH3D benchmark, GCT achieves an F1 score of 86.80, surpassing the substantial runner-up Wint3R by 9.52 points. This advantage extends to the NRGBD dataset, where our approach achieves an F1 score of 65.10, demonstrating robust generalization compared to existing solutions such as StreamVGGT and TTT3R.

4.2 Camera pose estimation

Large-Scale Trajectory Estimation on Oxford Spires. First, we evaluate the camera pose estimation performance of GCT on the challenging Oxford Spires [66] dataset. The dataset spans large-scale indoor and outdoor environments, resulting in complex scene changes within each sequence. We conduct experiments under two settings: *sparse* and *dense*. Following the evaluation

Table 1. Pose and trajectory accuracy comparison on the Oxford Spires dataset. Our method achieves the best performance on most metrics among prior offline, optimization-based, and online methods.

Methods	Type	AUC@15 ↑	AUC@30 ↑	ATE ↓	RPE-trans ↓	RPE-rot ↓
Fast3R [86]	offline	1.20	2.99	34.80	8.21	59.51
VGGT [72]	offline	23.84	35.09	24.78	8.87	22.79
DA3 [37]	offline	49.84	56.68	12.87	3.22	16.17
FastVGGT [58]	offline	21.68	34.64	22.43	7.25	16.12
π^3 [79]	offline	38.64	48.65	14.03	2.58	10.33
DroidSLAM [67]	optim	8.58	21.41	21.84	1.02	6.90
MegaSAM [33]	optim	15.91	28.03	13.80	0.76	7.24
VIPE [20]	optim	45.35	51.88	10.52	**0.43**	5.98
StreamVGGT [98]	online	10.91	17.04	28.41	6.35	16.28
SLAM3R [39]	online	1.67	5.10	29.69	7.57	27.50
InfiniteVGGT [93]	online	10.25	16.33	30.49	5.72	15.01
Spann3R [70]	online	2.06	5.09	32.12	3.54	14.37
Stream3R-w [29]	online	6.56	11.03	33.03	4.73	16.79
Stream3R [29]	online	9.67	15.21	29.58	6.67	16.90
CUT3R [76]	online	5.98	14.95	18.16	1.17	7.18
TTT3R [8]	online	13.92	25.90	19.35	2.28	13.30
Wint3R [34]	online	11.61	23.42	21.10	1.62	6.27
GCT (Ours)	online	**63.22**	**76.46**	**5.37**	0.93	**3.69**

Table 2. Inference time comparison on the Oxford Spires dataset. We compare with state-of-the-art feed-forward streaming 3D reconstruction methods in the dense setting (3,840 frames).

	CUT3R	TTT3R	Wint3R	Infinite-VGGT	Stream3R-w	GCT (Ours)
ATE ↓	32.47	25.05	32.90	31.75	33.73	**5.60**
FPS	**29.21**	28.97	3.88	7.78	13.66	20.29

protocol, on the Oxford Spires [66] dataset, we select 10 scenes with available ground-truth trajectories: "bodleian library 02", "christ church 02", "christ church 03", "christ church 05", "keble college 02", "keble college 03", "keble college 04", "keble college 05", "observatory quarter 01" and "observatory quarter 02". We select these scenes because ground truth trajectories in these scenes are available, and we use camera 0 as the viewpoint.

In the *sparse* setting, we sample one frame every 12 frames from the Oxford Spires sequences, resulting in 320 frames per scene. Under this configuration, offline, optimization-based, and online methods can all run on our hardware. The results are shown in Tab. 1 and Fig. 4. GCT demonstrates remarkable performance on the Oxford Spires dataset, achieving the best results across almost all metrics.

Notably, despite operating in a streaming online manner, our method surpasses the strongest offline baselines such as VGGT [72] and DA3 [37], reducing the Absolute Trajectory Error (ATE) from 24.78 and 12.87 to 5.37. VGGT and

Table 3. Quantitative comparison with state-of-the-art feed-forward streaming 3D reconstruction methods. Results on ETH3D, 7-Scenes, and Tanks and Temples show that our method achieves the best performance across all datasets.

Methods	Type	ETH3D			7-Scenes			Tanks & Temples		
		AUC3↑	AUC30↑	ATE↓	AUC3↑	AUC30↑	ATE↓	AUC3↑	AUC30↑	ATE↓
SLAM3R [39]	online	1.10	19.03	1.98	5.13	69.05	0.11	1.68	37.48	1.42
Spann3R [70]	online	0.72	23.34	2.11	2.99	59.59	0.20	1.80	25.85	2.12
InfiniteVGGT [93]	online	13.88	63.81	1.47	8.45	74.58	0.12	19.27	71.39	1.01
CUT3R [76]	online	8.04	47.52	1.43	1.63	43.86	0.30	1.23	19.93	1.80
TTT3R [8]	online	8.47	60.05	1.22	5.79	72.19	0.11	5.42	64.16	0.67
Wint3R [34]	online	8.07	64.02	0.86	2.94	65.02	0.12	2.88	49.48	0.88
Stream3R [29]	online	12.41	60.57	1.67	9.84	74.81	0.11	31.88	71.32	0.76
Stream3R-w [29]	online	6.76	50.90	1.72	7.91	63.21	0.26	18.86	62.45	1.23
GCT (Ours)	online	**40.34**	**87.96**	**0.43**	**13.20**	**79.06**	**0.08**	**44.99**	**92.34**	**0.21**

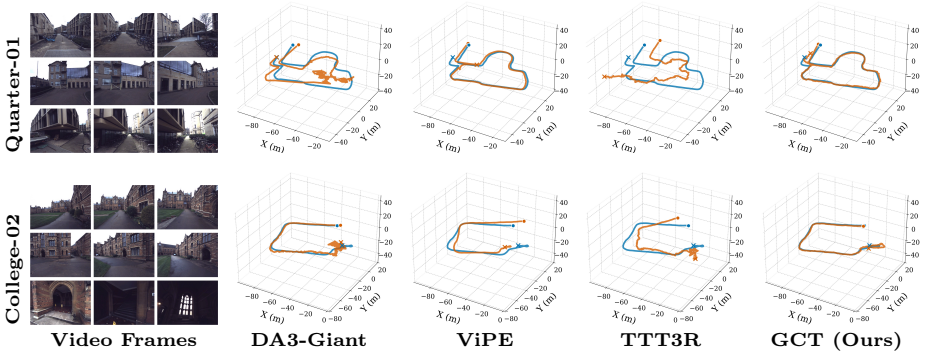


Fig. 4. Trajectories on the Oxford-Spires dataset. GCT directly predicts camera trajectories in a streaming manner and outperforms existing streaming (TTT3R), bidirectional (DA3-Giant), and optimization-based (ViPE) methods, accurately preserving trajectories during highly varying outdoor scenes, complex outdoor-to-indoor transitions, and climbs of dark staircases. In the figure, ground truth is shown in blue, predicted trajectories in orange, start points with •, and end points with ×.

DA3 are trained on datasets with very limited viewpoints, where the captured frames largely stay around the scene and consecutive frames are close to each other. As a result, these methods struggle to handle the complex scene changes present in Oxford Spires. Compared with optimization-based methods, GCT also achieves superior performance, outperforming the best optimization-based method, VIPE [20], by a significant margin (ATE: 5.37 vs. 10.52).

When compared strictly with other online methods, GCT maintains a clear performance advantage. All online methods tend to fail in most scenes in Oxford Spires due to severe memory forgetting, with the best ATE of 18.16 achieved by CUT3R [76], while our GCT achieves an ATE of 5.37.

The superior performance of GCT can be attributed to our efficient long-sequence training strategy and to the stability and efficiency of the GCA mechanism in long sequences and large scenes. We further evaluate the performance of GCT on the long sequence of Oxford Spires, as shown in Tab. 2. The results

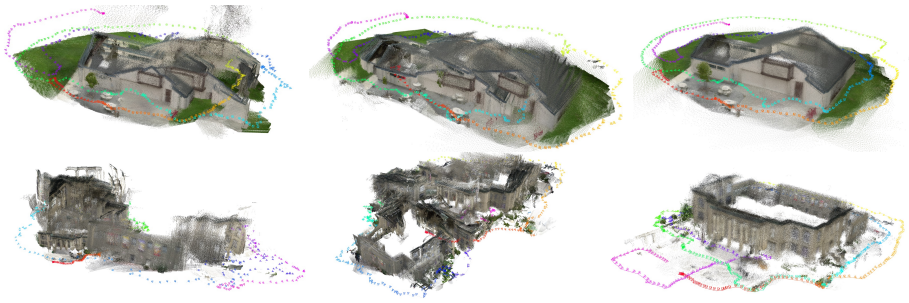


Fig. 5. Comparisons of reconstruction quality. Our reconstruction results significantly outperform other streaming methods, including the representative TTT3R and Wint3R. These approaches, due to excessive information compression, suffer from trajectory drift in long sequences.

show that GCT maintains its superior performance even in the long sequence setting, demonstrating its robustness and scalability in streaming scenarios.

In the *dense* setting, we evaluate the full 3840-frame sequences (Tab. 2). As trajectory length increases, most feed-forward methods degrade significantly due to accumulated drift (e.g., CUT3R: 18.16 → 32.47, Wint3R: 21.10 → 32.90). Our method maintains consistently low error (5.37 → 5.60), suggesting that Geometric Context Attention effectively preserves long-range geometric consistency without explicit optimization.

Generalization on Diverse Benchmarks. We compare GCT with state-of-the-art streaming methods on ETH3D [57] under the DA3 [37] protocol, 7-Scenes [59] with stride-5 sampling, and Tanks and Temples [27]; results are reported in Tab. 3. For Tanks and Temples, we use the Barn, Caterpillar, Church, Ignatius, Meetingroom, and Truck scenes. The results demonstrate that our GCT consistently outperforms all competing streaming approaches by a substantial margin across these diverse scenarios. Beyond numerical metrics, a qualitative comparison of reconstruction quality is provided in Fig. 5. When revisiting the scene after long temporal gaps, our method exhibits minimal drift, producing clear and consistent reconstructions of the building structures. In contrast, other methods suffer from severe trajectory drift and fragmented point clouds due to memory forgetting. This demonstrates the effectiveness of our Geometric Context Attention mechanism in maintaining long-sequence consistency.

We further show comparison on the dynamic Sintel [3] and the city-scale KITTI [17] odometry benchmark. As reported in Tab. 4, GCT attains the lowest ATE on both Sintel (0.10) and KITTI (24.12), surpassing DROID-SLAM [67], VGGT-Long [12], and TTT3R [8], despite running in a single online pass without state reset, pose-graph optimization, or loop closure.

Table 4. Comparison on Sintel and KITTI datasets. ATE (m)↓.

Method	Sintel	KITTI
DROID-SLAM [67]	0.20	129.34
VGGT-Long [12]	0.17	31.89
TTT3R [8]	0.22	171.63
GCT (Ours)	**0.10**	**24.12**

Table 5. Quantitative comparison with state-of-the-art streaming 3D reconstruction methods on ETH3D, 7-Scenes, and NRGBD. Our method achieves the best results.

Methods	Type	ETH3D			7-Scenes			NRGBD		
		Acc ↓	Comp ↓	F1 ↑	Acc ↓	Comp ↓	F1 ↑	Acc ↓	Comp ↓	F1 ↑
StreamVGGT [98]	online	0.64	0.34	58.11	0.04	0.11	69.44	0.13	0.05	45.08
InfiniteVGGT [93]	online	0.65	0.35	57.69	0.04	0.11	68.53	0.13	0.05	42.27
CUT3R [76]	online	0.57	0.50	67.63	0.07	0.10	58.98	0.25	0.15	32.22
TTT3R [8]	online	0.41	0.22	68.48	0.03	0.08	77.25	0.16	0.06	53.55
Wint3R [34]	online	0.28	0.21	77.28	0.03	0.07	78.81	0.09	0.04	56.96
Stream3R [29]	online	0.44	0.28	72.87	**0.02**	0.09	78.79	0.21	0.07	54.07
Stream3R-w [29]	online	0.58	0.37	67.09	0.04	0.15	71.94	0.20	0.06	53.74
GCT (Ours)	online	**0.17**	**0.08**	**86.80**	0.03	**0.04**	**82.38**	**0.07**	**0.03**	**65.10**

4.3 3D reconstruction

We evaluate the reconstruction performance of GCT on the 7-Scenes, ETH3D, and NRGBD [2] datasets. For the 7-Scenes and ETH3D datasets, we follow the evaluation settings established in the pose estimation experiments. Regarding the NRGBD [2] dataset, we sample images using a stride of 5. The quantitative results are summarized in Tab. 5.

The reconstructed and ground-truth point clouds are first aligned using the Umeyama method, followed by ICP refinement (threshold 0.1) to compute F1, accuracy, and completeness. For ETH3D, which features large-scale scenes with relatively sparse ground-truth points, we adopt an F1 threshold of 0.25. By contrast, 7-Scenes and NRGBD contain many frames and dense depth projections that produce tens of millions of points. For these datasets, we first perform Umeyama alignment, then downsample both point clouds to a uniform voxel grid (voxel size 4.0/512) before ICP, and set the F1 threshold to 0.05.

4.4 Analysis

Ablation study on GCA. As shown in Tab. 6, we conduct ablation studies to analyze the contributions of components in GCA. We initialized our model with the first-stage checkpoint and fine-tuned it on the TartanAir and TartanGround datasets, followed by evaluation on the TartanGround validation set.

Anchoring via Initialization Reference: Introducing Anchor Initialization (A. Init.) underscores the need for an initialization reference for coordinate and scale anchoring. Adding A. Init. to the baseline boosts AUC@3 from 9.80 to 13.63 and reduces ATE from 8.59 to 7.88, indicating that establishing an explicit initial geometric frame prevents drift accumulation in streaming scenarios.

Local Pose-Reference Context and Relative Supervision: Relative Loss proves indispensable for constructing a reliable local pose-reference graph for accurate relative motion estimation. Removing Relative Loss (third row) degrades rotational error (RPE-rot) to 5.35 and increases ATE to 8.25 compared to full supervision (fourth row, ATE 7.46, RPE-rot 2.26). This demonstrates that while

Table 6. Ablation study on long-sequence pose estimation and trajectory accuracy. All components contribute significantly to the final performance.

Rel. Loss	A. Init.	Co. Tok.	V. RoPE	AUC@3 ↑	AUC@30 ↑	ATE ↓	RPE-trans ↓	RPE-rot ↓
✓				9.80	65.84	8.59	1.62	2.57
✓	✓			13.63	68.71	7.88	1.60	2.90
	✓	✓		13.91	68.25	8.25	1.67	5.35
✓	✓	✓		15.75	69.92	7.46	1.48	2.26
✓	✓	✓	✓	**16.39**	**71.87**	**5.98**	**1.33**	**1.93**

Table 7. Effect of inference-time window size k and anchor count n on Oxford Spires. Each cell reports ATE (m)↓ / FPS on H800 ↑ / FPS on RTX 4090 ↑ at 518×378 resolution.

Anchor n \ Window k	16	32	48	64	128
2	12.0 / 26 / 19	7.9 / 25 / 16	6.0/ 22 / 14	5.5 / 20 / 12	5.6 / 15 / 9
4	11.9 / 26 / 19	7.8 / 25 / 15	5.8/ 22 / 14	5.4 / 20 / 12	5.4 / 15 / 9
8	11.2 / 25 / 18	8.0 / 24 / 15	5.5/ 22 / 14	5.3 / 20 / 12	5.3 / 15 / 9

global context aids long-term consistency, direct relative supervision constrains frame-to-frame motion within the local graph structure.

Global Trajectory Cues via Context Token and RoPE: Combining Context Tokens (Co. Tok.) and Video RoPE (V. RoPE) retains global trajectory cues for long-term consistency. Context Tokens aggregate global context, improving stability with an AUC@30 increase to 69.92 (fourth row). Integrating Video RoPE (last row) yields the best performance (AUC@3 16.39, ATE 5.98), suggesting accurate encoding of temporal order and relative distances.

Window Size and Anchor Count. As shown in Tab. 7 on Oxford Spires (518×378), accuracy saturates beyond k=48 and is largely insensitive to n. We thus adopt k=64, n=8, attaining the ATE of 5.3 at 20/12 FPS on an H800/RTX 4090.

5 Conclusion

We have presented GCT, a streaming foundation model for long-range 3D reconstruction from continuous visual input. At its core is Geometric Context Attention (GCA), which decomposes the streaming state into three complementary context types, anchor, local pose-reference window, and trajectory memory, inspired by the structure of classical SLAM systems but learned end-to-end. This design reduces per-frame context growth by roughly 80× compared to causal attention, enabling stable inference over long sequences at around 20 FPS.

Acknowledgements

We thank Ka Leong Cheng, Yipengjing Sun and Liangxiao Hu for their helpful discussions and support.

References

1. Arnold, E., Wynn, J., Vicente, S., Garcia-Hernando, G., Monszpart, A., Prisacariu, V., Turmukhambetov, D., Brachmann, E.: Map-free visual relocalization: Metric pose relative to a single image. In: Eur. Conf. Comput. Vis. pp. 690–708. Springer (2022)
2. Azinović, D., Martin-Brualla, R., Goldman, D.B., Nießner, M., Thies, J.: Neural rgb-d surface reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6290–6301 (2022)
3. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: A. Fitzgibbon et al. (Eds.) (ed.) Eur. Conf. Comput. Vis. pp. 611–625. Part IV, LNCS 7577, Springer-Verlag (Oct 2012)
4. Cabon, Y., Murray, N., Humenberger, M.: Virtual kitti 2. arXiv preprint arXiv:2001.10773 (2020)
5. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M., Tardós, J.D.: Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. IEEE transactions on robotics **37**(6), 1874–1890 (2021)
6. Chang, A., Dai, A., Funkhouser, T., Halber, M., Nießner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from RGB-D data in indoor environments. In: International Conference on 3D Vision (3DV) (2017)
7. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Easi3r: Estimating disentangled motion from dust3r without training. In: Int. Conf. Comput. Vis. pp. 9158–9168 (2025)
8. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Ttt3r: 3d reconstruction as test-time training. Int. Conf. Learn. Represent. (2026)
9. Chen, Z., Qin, M., Yuan, T., Liu, Z., Zhao, H.: Long3r: Long sequence streaming 3d reconstruction. In: Int. Conf. Comput. Vis. pp. 5273–5284 (2025)
10. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 5828–5839 (2017)
11. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 13142–13153 (2023)
12. Deng, K., Ti, Z., Xu, J., Yang, J., Xie, J.: Vggt-long: Chunk it, loop it, align it–pushing vggt's limits on kilometer-scale long rgb sequences. arXiv preprint arXiv:2507.16443 (2025)
13. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 224–236 (2018)
14. Feng, H., Zhang, J., Wang, Q., Ye, Y., Yu, P., Black, M.J., Darrell, T., Kanazawa, A.: St4rtrack: Simultaneous 4d reconstruction and tracking in the world. In: Int. Conf. Comput. Vis. pp. 8503–8513 (2025)
15. Fonder, M., Droogenbroeck, M.V.: Mid-air: A multi-modal dataset for extremely low altitude drone flights. In: Conference on Computer Vision and Pattern Recognition Workshop (CVPRW) (June 2019)
16. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE transactions on pattern analysis and machine intelligence **32**(8), 1362–1376 (2009)
17. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: IEEE Conf. Comput. Vis. Pattern Recog. (2012)

18. Greff, K., Belletti, F., Beyer, L., Doersch, C., Du, Y., Duckworth, D., Fleet, D.J., Gnanapragasam, D., Golemo, F., Herrmann, C., et al.: Kubric: A scalable dataset generator. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3749–3761 (2022)
19. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: LRM: Large reconstruction model for single image to 3D. In: Int. Conf. Learn. Represent. (2024)
20. Huang, J., Zhou, Q., Rabeti, H., Korovko, A., Ling, H., Ren, X., Shen, T., Gao, J., Slepichev, D., Lin, C.H., et al.: Vipe: Video pose engine for 3d geometric perception. arXiv preprint arXiv:2508.10934 (2025)
21. Huang, P.H., Matzen, K., Kopf, J., Ahuja, N., Huang, J.B.: Deepmvs: Learning multi-view stereopsis. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2821–2830 (2018)
22. Jacobs, S.A., Tanaka, M., Zhang, C., Zhang, M., Song, S.L., Rajbhandari, S., He, Y.: Deepspeed ulysses: System optimizations for enabling training of extreme long sequence transformer models. arXiv preprint arXiv:2309.14509 (2023)
23. Jang, W., Weinzaepfel, P., Leroy, V., Agapito, L., Revaud, J.: Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1071–1081 (2025)
24. Jiang, L., Mao, Y., Xu, L., Lu, T., Ren, K., Jin, Y., Xu, X., Yu, M., Pang, J., Zhao, F., et al.: Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. ACM Trans. Graph. **44**(6), 1–16 (2025)
25. Jin, L., Tucker, R., Li, Z., Fouhey, D., Snavely, N., Holynski, A.: Stereo4D: Learning How Things Move in 3D from Internet Stereo Videos. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
26. Keetha, N., Müller, N., Schönberger, J., Porzi, L., Zhang, Y., Fischer, T., Knapitsch, A., Zauss, D., Weber, E., Antunes, N., et al.: Mapanything: Universal feed-forward metric 3d reconstruction. arXiv preprint arXiv:2509.13414 (2025)
27. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Trans. Graph. **36**(4) (2017)
28. Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J.E., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles (2023)
29. Lan, Y., Luo, Y., Hong, F., Zhou, S., Chen, H., Lyu, Z., Yang, S., Dai, B., Loy, C.C., Pan, X.: STream3R: Scalable sequential 3D reconstruction with causal transformer. In: Int. Conf. Learn. Represent. (2026)
30. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3D with sparse-view generation and large reconstruction model. In: Int. Conf. Learn. Represent. (2024)
31. Li, Y., Jiang, L., Xu, L., Xiangli, Y., Wang, Z., Lin, D., Dai, B.: Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In: Int. Conf. Comput. Vis. pp. 3205–3215 (2023)
32. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2041–2050 (2018)
33. Li, Z., Tucker, R., Cole, F., Wang, Q., Jin, L., Ye, V., Kanazawa, A., Holynski, A., Snavely, N.: Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 10486–10496 (2025)
34. Li, Z., Zhou, J., Wang, Y., Guo, H., Chang, W., Zhou, Y., Zhu, H., Chen, J., Shen, C., He, T.: Wint3r: Window-based streaming reconstruction with camera token pool. In: Int. Conf. Learn. Represent. (2026)

35. Liao, Y., Xie, J., Geiger, A.: Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE Trans. Pattern Anal. Mach. Intell. **45**(3), 3292–3310 (2022)
36. Lin, C.Y., Sun, C., Yang, F.E., Chen, M.H., Lin, Y.Y., Liu, Y.L.: Longsplat: Robust unposed 3d gaussian splatting for casual long videos. In: Int. Conf. Comput. Vis. pp. 27412–27422 (2025)
37. Lin, H., Chen, S., Liew, J.H., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
38. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., et al.: Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 22160–22169 (2024)
39. Liu, Y., Dong, S., Wang, S., Yin, Y., Yang, Y., Fan, Q., Chen, B.: Slam3r: Real-time dense scene reconstruction from monocular rgb videos. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 16651–16662 (2025)
40. Lu, J., Huang, T., Li, P., Dou, Z., Lin, C., Cui, Z., Dong, Z., Yeung, S.K., Wang, W., Liu, Y.: Align3r: Aligned monocular depth estimation for dynamic videos. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 22820–22830 (2025)
41. Lu, Y., Zhang, J., Fang, T., Nahmias, J.D., Tsin, Y., Quan, L., Cao, X., Yao, Y., Li, S.: Matrix3d: Large photogrammetry model all-in-one. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 11250–11263 (2025)
42. Maggio, D., Lim, H., Carlone, L.: Vggt-slam: Dense rgb slam optimized on the sl (4) manifold. Adv. Neural Inform. Process. Syst. **39** (2025)
43. McCormac, J., Handa, A., Leutenegger, S., J.Davison, A.: Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? (2017)
44. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: Orb-slam: A versatile and accurate monocular slam system. IEEE transactions on robotics **31**(5), 1147–1163 (2015)
45. Mur-Artal, R., Tardós, J.D.: Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. IEEE transactions on robotics **33**(5), 1255–1262 (2017)
46. Murai, R., Dexheimer, E., Davison, A.J.: Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 16695–16705 (2025)
47. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Howes, R., Huang, P.Y., Xu, H., Sharma, V., Li, S.W., Galuba, W., Rabbat, M., Assran, M., Ballas, N., Synnaeve, G., Misra, I., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: Dinov2: Learning robust visual features without supervision (2023)
48. Pan, L., Baráth, D., Pollefeys, M., Schönberger, J.L.: Global structure-from-motion revisited. In: Eur. Conf. Comput. Vis. pp. 58–77. Springer (2024)
49. Pan, X., Charron, N., Yang, Y., Peters, S., Whelan, T., Kong, C., Parkhi, O., Newcombe, R., Ren, Y.C.: Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In: Int. Conf. Comput. Vis. pp. 20133–20143 (2023)
50. Patel, M., Yang, F., Qiu, Y., Cadena, C., Scherer, S., Hutter, M., Wang, W.: Tartanground: A large-scale dataset for ground robot perception and navigation. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 20524–20531. IEEE (2025)

51. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., Savva, M., Zhao, Y., Batra, D.: Habitat-matterport 3D dataset (HM3D): 1000 large-scale 3D environments for embodied AI. In: Adv. Neural Inform. Process. Syst. (2021)
52. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: Int. Conf. Comput. Vis. pp. 10901–10911 (2021)
53. Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: Int. Conf. Comput. Vis. pp. 10912–10922 (2021)
54. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4938–4947 (2020)
55. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 4104–4113 (2016)
56. Schönberger, J.L., Zheng, E., Frahm, J.M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: European conference on computer vision. pp. 501–518. Springer (2016)
57. Schops, T., Schonberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 3260–3269 (2017)
58. Shen, Y., Zhang, Z., Qu, Y., Zheng, X., Ji, J., Zhang, S., Cao, L.: Fastvggt: Training-free acceleration of visual geometry transformer. arXiv preprint arXiv:2509.02560 (2025)
59. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in rgb-d images. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2930–2937 (2013)
60. Smart, B., Zheng, C., Laina, I., Prisacariu, V.A.: Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. arXiv preprint arXiv:2408.13912 (2024)
61. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: ACM siggraph 2006 papers, pp. 835–846 (2006)
62. Sucar, E., Lai, Z., Insafutdinov, E., Vedaldi, A.: Dynamic point maps: A versatile representation for dynamic 3d reconstruction. In: Int. Conf. Comput. Vis. pp. 7295–7305 (2025)
63. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 8922–8931 (2021)
64. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 2446–2454 (2020)
65. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: LGM: Large multi-view gaussian model for high-resolution 3D content creation. In: Eur. Conf. Comput. Vis. (2024)
66. Tao, Y., Muñoz-Bañón, M.Á., Zhang, L., Wang, J., Fu, L.F.T., Fallon, M.: The oxford spires dataset: Benchmarking large-scale lidar-visual localisation, reconstruction and radiance field methods. International Journal of Robotics Research (2025)
67. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. Adv. Neural Inform. Process. Syst. **34**, 16558–16569 (2021)

68. Tosi, F., Liao, Y., Schmitt, C., Geiger, A.: Smd-nets: Stereo mixture density networks. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 8942–8952 (2021)
69. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., Zeng, J., Wang, J., Zhang, J., Zhou, J., Wang, J., Chen, J., Zhu, K., Zhao, K., Yan, K., Huang, L., Feng, M., Zhang, N., Li, P., Wu, P., Chu, R., Feng, R., Zhang, S., Sun, S., Fang, T., Wang, T., Gui, T., Weng, T., Shen, T., Lin, W., Wang, W., Wang, W., Zhou, W., Wang, W., Shen, W., Yu, W., Shi, X., Huang, X., Xu, X., Kou, Y., Lv, Y., Li, Y., Liu, Y., Wang, Y., Zhang, Y., Huang, Y., Li, Y., Wu, Y., Liu, Y., Pan, Y., Zheng, Y., Hong, Y., Shi, Y., Feng, Y., Jiang, Z., Han, Z., Wu, Z.F., Liu, Z.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
70. Wang, H., Agapito, L.: 3d reconstruction with spatial memory. In: 3DV. pp. 78–89. IEEE (2025)
71. Wang, J., Yuan, Y., Zheng, R., Lin, Y., Gao, J., Chen, L.Z., Bao, Y., Zhang, Y., Zeng, C., Zhou, Y., et al.: Spatialvid: A large-scale video dataset with spatial annotations. In: IEEE Conf. Comput. Vis. Pattern Recog. (2026)
72. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
73. Wang, J., Karaev, N., Rupprecht, C., Novotny, D.: Vggsfm: Visual geometry grounded deep structure from motion. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 21686–21697 (2024)
74. Wang, K., Shen, S.: Flow-motion and depth network for monocular stereo and beyond. IEEE Robotics and Automation Letters **5**(2), 3307–3314 (2020)
75. Wang, P., Tan, H., Bi, S., Xu, Y., Luan, F., Sunkavalli, K., Wang, W., Xu, Z., Zhang, K.: PF-LRM: Pose-free large reconstruction model for joint pose and shape prediction. In: Int. Conf. Learn. Represent. (2024)
76. Wang*, Q., Zhang*, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. In: IEEE Conf. Comput. Vis. Pattern Recog. (2025)
77. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 20697–20709 (2024)
78. Wang, W., Zhu, D., Wang, X., Hu, Y., Qiu, Y., Wang, C., Hu, Y., Kapoor, A., Scherer, S.: Tartanair: A dataset to push the limits of visual slam. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4909–4916. IEEE (2020)
79. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3: Permutation-equivariant visual geometry learning. arXiv preprint arXiv:2507.13347 (2025)
80. Xia, F., Zamir, A.R., He, Z., Sax, A., Malik, J., Savarese, S.: Gibson env: Real-world perception for embodied agents. In: IEEE Conf. Comput. Vis. Pattern Recog. (2018)
81. Xia, H., Fu, Y., Liu, S., Wang, X.: Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 22378–22389 (2024)
82. Xiao, Y., Wang, J., Xue, N., Karaev, N., Makarov, I., Kang, B., Zhu, X., Bao, H., Shen, Y., Zhou, X.: Spatialtrackerv2: 3d point tracking made easy. In: Int. Conf. Comput. Vis. (2025)

83. Xie, T., Yang, P., Jin, Y., Cai, Y., Yin, W., Ren, W., Zhang, Q., Hua, W., Peng, S., Guo, X., Zhou, X.: Scal3r: Scalable test-time training for large-scale 3d reconstruction. In: IEEE Conf. Comput. Vis. Pattern Recog. (2026)
84. Xu, Y., Shi, Z., Yifan, W., Chen, H., Yang, C., Peng, S., Shen, Y., Wetzstein, G.: GRM: Large gaussian reconstruction model for efficient 3d reconstruction and generation. In: Eur. Conf. Comput. Vis. (2024)
85. Xu, Y., Tan, H., Luan, F., Bi, S., Wang, P., Li, J., Shi, Z., Sunkavalli, K., Wetzstein, G., Xu, Z., Zhang, K.: DMV3D: Denoising multi-view diffusion using 3D large reconstruction model. In: Int. Conf. Learn. Represent. (2024)
86. Yang, J., Sax, A., Liang, K.J., Henaff, M., Tang, H., Cao, A., Chai, J., Meier, F., Feiszli, M.: Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 21924–21935 (2025)
87. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. In: Eur. Conf. Comput. Vis. pp. 767–783 (2018)
88. Yao, Y., Luo, Z., Li, S., Shen, T., Fang, T., Quan, L.: Recurrent mvsnet for high-resolution multi-view stereo depth inference. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 5525–5534 (2019)
89. Yao, Y., Luo, Z., Li, S., Zhang, J., Ren, Y., Zhou, L., Fang, T., Quan, L.: Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 1790–1799 (2020)
90. Ye, B., Liu, S., Xu, H., Li, X., Pollefeys, M., Yang, M.H., Peng, S.: No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. In: Int. Conf. Learn. Represent. (2025)
91. Ye, Z., Chen, L., Lai, R., Lin, W., Zhang, Y., Wang, S., Chen, T., Kasikci, B., Grover, V., Krishnamurthy, A., Ceze, L.: Flashinfer: Efficient and customizable attention engine for llm inference serving. arXiv preprint arXiv:2501.01005 (2025)
92. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Int. Conf. Comput. Vis. pp. 12–22 (2023)
93. Yuan, S., Yang, Y., Yang, X., Zhang, X., Zhao, Z., Zhang, L., Zhang, Z.: Infinitevggt: Visual geometry grounded transformer for endless streams. arXiv preprint arXiv:2601.02281 (2026)
94. Zhang, J., Herrmann, C., Hur, J., Jampani, V., Darrell, T., Cole, F., Sun, D., Yang, M.H.: Monst3r: A simple approach for estimating geometry in the presence of motion. In: Int. Conf. Learn. Represent. (2025)
95. Zhang, S., Wang, J., Xu, Y., Xue, N., Rupprecht, C., Zhou, X., Shen, Y., Wetzstein, G.: Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In: IEEE Conf. Comput. Vis. Pattern Recog. pp. 21936–21947 (2025)
96. Zhang, Y., Zhang, L., Ma, R., Cao, N.: Texverse: A universe of 3d objects with high-resolution textures. arXiv preprint arXiv:2508.10868 (2025)
97. Zheng, Y., Harley, A.W., Shen, B., Wetzstein, G., Guibas, L.J.: Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In: Int. Conf. Comput. Vis. pp. 19855–19865 (2023)
98. Zhuo, D., Zheng, W., Guo, J., Wu, Y., Zhou, J., Lu, J.: Streaming 4d visual geometry transformer. In: Int. Conf. Learn. Represent. (2026)