GHOST: Geometry-Hierarchical Online Streaming Token Eviction for Efficient 3D Reconstruction

Leyang Chen[*], **Junyi Wu**[*], **Zhiteng Li**[*], **Yulun Zhang**[†]
Shanghai Jiao Tong University

Abstract

Streaming 3D reconstruction from long monocular video sequences requires maintaining a key-value (KV) cache that grows linearly with sequence length, creating a severe memory bottleneck. Existing approaches either truncate the cache to a fixed set of anchor frames, leading to reconstruction quality degradation, or rely on attention-score heuristics that are agnostic to 3D scene structure, failing to preserve geometrically valuable tokens. To address these problems, we present **GHOST** (**G**eometry-**H**ierarchical **O**nline **S**treaming **T**oken Eviction), a KV cache management framework that requires no additional model training and uses the model's own 3D geometry outputs to evict redundant tokens online. GHOST introduces three mutually reinforcing innovations: a hierarchical dual-level importance scoring scheme, a privilege mechanism that prioritizes special tokens for retention, and a cosine-similarity-guided layer-wise budget allocation. Experiments on various benchmarks show that GHOST preserves excellent reconstruction quality while cutting the KV cache by nearly half and delivering $1.75\times$ faster inference compared to state-of-the-art methods. Our code is available at https://github.com/lokiniuniu/GHOST.

1 Introduction

Transformer models [17] have achieved remarkable results in 3D reconstruction from monocular images [21, 10, 18], learning to predict dense depth, point maps, and camera poses in a single forward pass. VGGT [18] extends this to multi-view sequences by attending jointly over all input frames, delivering SOTA reconstruction quality. However, joint attention scales quadratically with sequence length [34], making it infeasible for longer video-length inputs.

Enabling long-sequence 3D reconstruction [18, 8, 24] therefore requires a streaming paradigm where the model processes frames incrementally. StreamVGGT [37] pioneers this direction by processing frames one at a time and maintaining a growing KV cache of past observations. InfiniteVGGT [33] pushes this further by introducing an explicit token eviction strategy: once the KV cache exceeds a fixed budget, tokens whose keys are least similar to the current query are discarded for new frames. Yet this strategy has a fundamental limitation, as attention scores

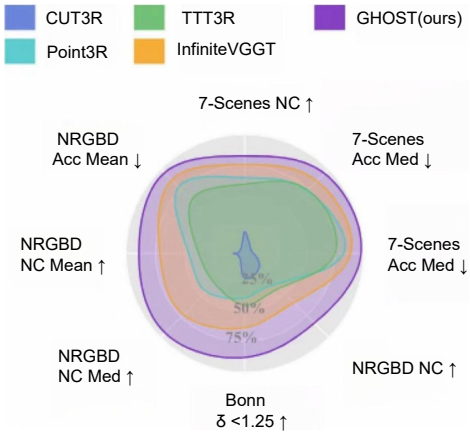


Figure 1: Radar comparison across 7-Scenes, NRGBD and Bonn, averaged over all input lengths. Metrics with downward arrows are inverted for this plot, so farther from the center is better.

[*]Equal contribution
[†]Corresponding author: Yulun Zhang, yulun100@gmail.com

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

from the current query frame cannot faithfully reflect the long-term geometric value of historical tokens [11].

As shown in Figure 2, the Spearman correlation between key-similarity scores and camera pose change is only -0.07, and with depth gradient variance $+0.31$. It confirms that key-similarity cannot make full use of the signals that determine a token's long-term geometric value. However, both signals are already available from the model's own outputs [18]. Moreover, transformer layers vary significantly in transformation strength: blocks with near-identical input and output representations can tolerate smaller token budgets without accuracy loss [31], yet uniform allocation ignores this heterogeneity.

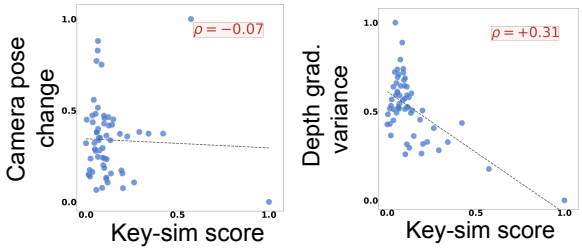


Figure 2: Correlation between Key-sim score and two frame attributes: Left: Negligible linear correlation between Key-sim score and camera pose change ($\rho = -0.07$); right: Moderate positive linear correlation between Key-sim score and depth gradient variance ($\rho = +0.31$). Dashed lines denote linear fitting trends.

Building on these observations, we propose **GHOST** (**G**eometry-**H**ierarchical **O**nline **S**treaming **T**oken Eviction), a method requiring no additional model training, eviction module for StreamVGGT [37]. GHOST scores each cached token by a *hierarchical dual-level importance* signal derived entirely from the model's own outputs: a *frame-level* component integrating camera pose change, depth gradient variance, and temporal recency, which captures whether a viewpoint is geometrically distinctive; a *token-level* component integrating visual saliency, depth confidence, and 3D point confidence, which identifies which patches carry reliable reconstruction value. The two levels are combined per-patch and updated incrementally online without extra forward passes. A *special-token privilege mechanism* applies a deterministic priority boost to camera and register tokens, increasing retention of globally important pose and structure encodings. A *cosine-similarity-guided layer-wise budget allocation* concentrates the global token budget where transformer layers perform the strongest transformations, while reducing budgets for near-identity later layers.

At half the token budget, GHOST matches InfiniteVGGT's reconstruction accuracy (0.041 versus 0.040) while reducing KV-cache memory from 6.94 to 3.51 GB and increasing throughput by $1.75\times$ (Table 8). The method still requires held-out hyperparameter selection and a one-time offline layer profile. Additional cross-backbone, dynamic-scene, matched-budget, and overhead results are reported in Appendix G.

Our contributions are fourfold:

- **Hierarchical dual-level importance scoring**: we decompose token importance into geometry-aware frame-level and token-level components derived from the model's own outputs, updated incrementally online without extra forward passes.
- **Special-token privilege mechanism**: we introduce a deterministic importance boost that prioritizes camera and register tokens for retention, reducing the risk of losing global pose and structure encodings.
- **Cosine-similarity-guided layer-wise budget allocation**: we profile transformer layers offline and concentrate the token budget where transformations are strongest.
- **Comprehensive empirical evaluation**: as shown in Fig 1, experiments on 7-Scenes [16], NRGBD [1], Bonn [14], and Long3D [33] across four input-length regimes show that GHOST outperforms most baselines at matched budgets. GHOST also maintains competitive accuracy while cutting the KV cache by nearly half and delivering $1.75\times$ faster inference compared to state-of-the-art methods.

2 Related Work

Streaming and long-sequence 3D reconstruction. Feed-forward multi-view reconstruction models such as DUSt3R [21] and MASt3R [10] regress 3D point maps and camera poses from image pairs, enabling downstream dense reconstruction by global alignment. VGGT [18] scales to multi-view sequences by processing all frames jointly with a full cross-frame attention transformer. StreamVGGT [37] adopts a streaming inference paradigm where each new frame attends to a fixed-size KV cache populated by prior frames, enabling causal online processing. InfiniteVGGT [33]

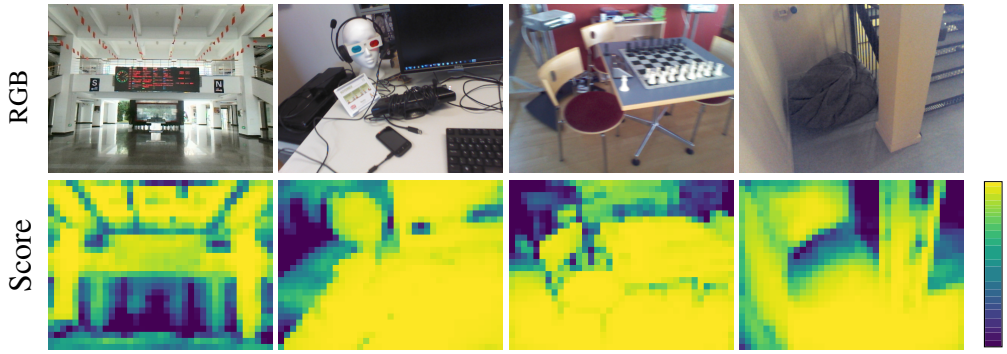


Figure 3: Top row: Raw RGB inputs from Long3D Lecture Hall, 7-Scenes Heads, 7-Scenes Chess, and 7-Scenes Stairs. Bottom row: Corresponding GHOST token importance heatmaps, illustrating that GHOST prioritizes visually/structurally salient regions across diverse scenarios.

extends this to unbounded sequences by evicting tokens ranked by key-space cosine similarity to the current query. Our work proposes an alternative eviction criterion based on multi-modal 3D geometry signals, which we show to be more informative than attention-based rankings.

KV cache compression for transformers. The memory bottleneck of long-context transformers [28, 22] has spurred extensive research on KV cache compression [26, 36, 12, 27, 23]. StreamingLLM [26] retains only "attention sink" tokens and recent frames, discarding most of historical context. H2O [36] identifies "heavy hitter" tokens via accumulated attention scores and preserves them across steps. SnapKV [12] clusters KV entries by key similarity to select a representative set. PyramidKV [3] allocates larger budgets to lower transformer layers, motivated by the observation that lower layers attend more broadly. All of these methods derive importance from attention patterns or token similarities *within the language modality*. Our work departs fundamentally by exploiting *3D geometry outputs*—depth confidence, point confidence, and camera pose, as the primary importance signal, a direction that has not been explored in prior KV cache compression work.

Token reduction in vision transformers. EViT [6] and DynamicViT [15] progressively discard tokens based on attention-derived relevance scores. A-ViT [32] learns per-token adaptive halting thresholds, while Evo-ViT [29] routes tokens into slow and fast processing streams. On the merging side, ToMe [2] fuses redundant tokens via bipartite matching of key-space similarities, and DiffRate [4] relaxes the compression rate into a differentiable objective for joint optimisation. For video transformers, STTS [19] jointly selects tokens along spatial and temporal axes to exploit inter-frame redundancy. Evict3R [13] performs training-free token eviction for memory-bounded streaming 3D reconstruction, selecting tokens via attention-based importance and KV-cache budgeting. Unlike attention-based selection, our scoring exploits depth, point confidence, and camera-pose signals produced by the model itself.

3 Motivation

3.1 Observation and Motivation

Key-similarity eviction lacks geometric grounding. InfiniteVGGT [33] retains tokens whose keys are least similar to the others. As shown in Figure 2, however, Key-sim scores correlate negligibly with camera pose change ($\rho{=}-0.07$) and moderately with depth gradient variance ($\rho{=}+0.31$). These weak-to-moderate correlations suggest that key similarity alone does not capture the geometric distinctiveness of cached frames. Frames with large camera shifts or rich structural detail may therefore be underrepresented.

Importance is inherently two-level. Figure 3 shows that within a single frame, structurally salient patches carry far more reconstruction value than flat, textureless regions. Token importance is thus two-level: a *frame-level* component captures whether a viewpoint is geometrically distinctive, while a *token-level* component identifies which patches within that frame are most informative.

Transformer layers exhibit varying transformation strengths. Transformer layers differ substantially in how strongly they transform their inputs [31]. Near-identity layers tolerate smaller token budgets without accuracy loss, whereas early high-transformation layers benefit from larger budgets. Uniform allocation ignores this heterogeneity, motivating a layer-wise budget strategy guided by each layer's transformation strength. Specifically, we allocate larger budgets to layers with stronger transformations and smaller budgets to near-identity layers.

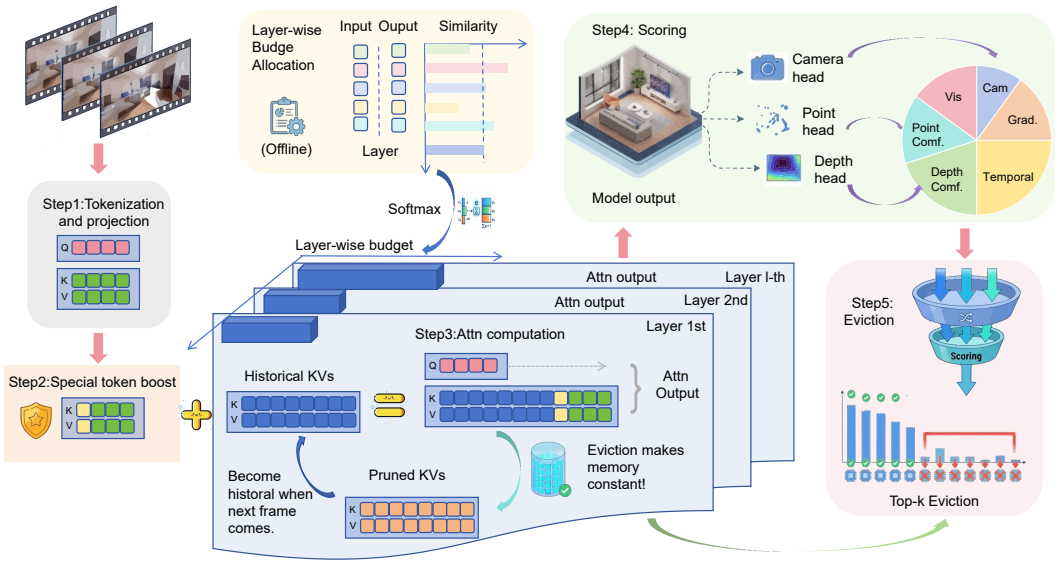


Figure 4: GHOST inference pipeline. **Offline:** Cosine-similarity profiling allocates per-layer budgets. **Online:** An eviction mode that prunes KV cache to layer-wise budget computed offline with Geometry-Hierarchical Importance scoring and Special token boost .

4 GHOST: Geometry-Hierarchical Online Streaming Token Eviction

4.1 Preliminaries

StreamVGGT [37] processes frames $\{I_t\}_{t=1}^{T}$ with $N_p = H_p \times W_p$ patch tokens, prepending $p_0 = 1+R$ special tokens ($\mathbf{c}_t$ and $\{\mathbf{r}_t^i\}_{i=1}^{R}$) for a total of N tokens per frame. L global self-attention layers maintain per-layer KV caches with total budget B_{total}. Outputs comprise depth $\{d_t, c_t^d\}$, point maps $\{P_t, c_t^p\}$, and pose $\boldsymbol{\pi}_t = (\mathbf{T}_t, \mathbf{q}_t, f_t)$. Our final goal is to allocate the limited KV cache budget across layers to maximize overall prediction accuracy.

GHOST assigns per-patch importance $\phi(t,p) = w_f s_{\text{frame}}(t) + w_k s_{\text{token}}(t,p)$, where s_{frame} combines camera motion s_{cam}, depth variance s_{geo}, and recency s_{temp}, while s_{token} aggregates visual saliency s_{sal} from feature map $\mathbf{F}_t$, depth confidence $c_t^d(p)$, and point confidence $c_t^p(p)$. Special tokens receive a deterministic boost with hyperparameters $\Delta_{\text{boost}}, \epsilon_{\text{tb}}, r_{p_{\text{sp}}}$. Per-layer budgets B_ℓ are allocated by importance $\pi_\ell \propto \exp(a_\ell/\tau)$ where $a_\ell = 1 - \bar{\rho}_\ell$ is derived from offline cosine similarity statistics.Figure 4 provides an overview of the GHOST inference pipeline.

4.2 Hierarchical Dual-Level Importance Scoring

GHOST decomposes token importance into a frame-level component $s_{\text{frame}}(t)$ and a token-level component $s_{\text{token}}(t,p)$, combined into a per-patch importance score $\phi(t,p)$. This two-level decomposition mirrors the hierarchical structure of 3D scene understanding: relevant information must first be identified at the viewpoint level (which frames are geometrically distinctive?) and then refined at the patch level (which spatial regions within those frames are most informative?).

4.2.1 Frame-Level Importance

The frame-level score $s_{\text{frame}}(t)$ captures the global geometric value of an entire frame as a KV-cache contributor, reflecting distinct aspects of a frame's reconstruction value.

Camera pose change. Frames from significantly different viewpoints provide complementary geometric coverage that cannot be inferred from nearby frames. Redundant frames captured from nearly identical poses duplicate information already present in the cache, whereas frames with large pose differences introduce new scene geometry and viewing directions critical for triangulation and novel-view synthesis. Let $\boldsymbol{\pi}_t = (\mathbf{T}_t, \mathbf{q}_t, f_t)$ encode translation, unit quaternion, and focal length. The camera change score relative to the preceding frame is:

$$s_{\text{cam}}(t) \;=\; \|\mathbf{T}_t - \mathbf{T}_{t-1}\|_2 + 1 - |\mathbf{q}/\|\mathbf{q}\|_t \cdot \mathbf{q}/\|\mathbf{q}\|_{t-1}|. \tag{1}$$

The two terms respectively penalise translational proximity and rotational similarity, ensuring that both lateral movement and pure rotations are treated as geometrically meaningful changes.

Depth gradient variance. Frames with high spatial depth variation contain richer geometric detail (edges, object boundaries, depth discontinuities) and are harder to approximate by neighbouring frames. Conversely, frames depicting large uniform surfaces—such as blank walls or open floors—contribute little unique structural information and are less critical to retain. Given depth map d_t:

$$s_{\text{geo}}(t) \;=\; \mathrm{Var}\big(\|\nabla d_t\|\big). \tag{2}$$

High variance in the depth gradient magnitude indicates a spatially diverse scene structure, making the frame a geometrically rich reference for reconstruction. Such frames are therefore more informative for preserving fine-grained 3D structure across time. Accordingly, we prioritize them during memory retention or budget allocation to improve reconstruction fidelity.

Temporal recency. Under causal streaming, the model processes frames sequentially and must infer scene state incrementally. Recent frames share greater contextual overlap with the current query frame—in terms of scene content, lighting, and object configuration—and therefore provide more directly applicable geometric context. We assign $s_{\text{temp}}(t) = t/T_{\text{cur}}$, where T_{cur} is the current frame index. This linear schedule prioritises recency without entirely discarding older frames, which may still encode unique viewpoints not revisited since.

All three raw scores operate on different scales and distributions, so each is mapped through sigmoid $\sigma(\cdot)$ to $[0, 1]$ before aggregation. The frame-level score is then their normalised weighted sum:

$$s_{\text{frame}}(t) \;=\; \frac{w_{\text{cam}}\,\sigma(s_{\text{cam}}(t)) + w_{\text{geo}}\,\sigma(s_{\text{geo}}(t)) + w_{\text{temp}}\,\sigma(s_{\text{temp}}(t))}{\max_{t'\in\mathcal{T}}\big(w_{\text{cam}}\,\sigma(s_{\text{cam}}(t')) + w_{\text{geo}}\,\sigma(s_{\text{geo}}(t')) + w_{\text{temp}}\,\sigma(s_{\text{temp}}(t'))\big) + \epsilon}, \tag{3}$$

where $t' \in \mathcal{T}$ ranges over all frames currently stored in the KV cache. Division by the maximum value across all cached frames ensures $s_{\text{frame}} \in [0, 1]$, thus keeping it commensurate with the token-level score in the final combination. This normalisation makes the scoring robust to variations in scene dynamics and sequence length. It also enables consistent weighting across different video segments without requiring per-sequence tuning. Consequently, $s_{\text{frame}}(t)$ can be directly integrated with token-level importance for unified budget allocation.

4.2.2 Token-Level Importance

While the frame-level score determines how valuable a viewpoint is as a whole, the token-level score $s_{\text{token}}(t, p)$ provides fine-grained discrimination *within* a frame by measuring the geometric informativeness of each individual patch. Even within a highly distinctive frame, only a subset of patches—those at structural boundaries or exhibiting high reconstruction confidence—contribute disproportionately to scene understanding.

Visual saliency. High-gradient patches correspond to edges, corners, and texture-rich regions that anchor geometric correspondence and are thus most informative for depth and point estimation. Uniform or featureless patches, by contrast, provide weak cues and are more readily interpolated from neighbouring tokens. Let $\mathbf{F}_t \in \mathbb{R}^{H_p \times W_p \times d}$ be the patch feature map extracted from the visual encoder. Saliency is measured as the spatial gradient magnitude of the feature representation:

$$s_{\text{sal}}(t, p) \;=\; \sqrt{\big\|\delta_x \mathbf{F}_t(p)\big\|^2 + \big\|\delta_y \mathbf{F}_t(p)\big\|^2}, \tag{4}$$

where δ_x, δ_y denote horizontal and vertical finite differences over the spatial patch grid. Computing saliency in feature space rather than raw pixel space makes it robust to illumination variation while remaining sensitive to semantically meaningful structure.

Depth and point confidence. Model-predicted confidence scores provide a direct signal of which patches yield reliable 3D estimates. The depth head outputs a per-patch confidence $c_t^d(p) \in [0, 1]$ and the point head outputs $c_t^p(p) \in [0, 1]$; both reflect the model's epistemic certainty about the geometric reconstruction at patch p. Patches with low confidence—typically arising from occlusion boundaries, reflective surfaces, or textureless regions—contribute noisier estimates and are therefore assigned lower importance. For past frames, confidence maps are pooled to patch resolution via adaptive average pooling and stored as lightweight per-patch metadata alongside the KV cache. The token-level score aggregates all three signals:

$$s_{\text{token}}(t, p) \;=\; \frac{w_{\text{sal}}\,\sigma(s_{\text{sal}}(t, p)) + w_{\text{dc}}\,\sigma(c_t^d(p)) + w_{\text{pc}}\,\sigma(c_t^p(p))}{\max_{t'\in\mathcal{T},\, p'\in\mathcal{P}}\big(w_{\text{sal}}\,\sigma(s_{\text{sal}}(t', p')) + w_{\text{dc}}\,\sigma(c_{t'}^d(p')) + w_{\text{pc}}\,\sigma(c_{t'}^p(p'))\big) + \epsilon}, \tag{5}$$

where $t' \in \mathcal{T}$ iterates over all cached frames and $p' \in \mathcal{P}$ iterates over all N_p patch positions within each frame. The joint maximum over (t', p') normalises scores globally across the entire cache so that $s_{\text{token}} \in [0, 1]$ and patch scores from different frames remain directly comparable during eviction.

Table 1: Quantitative comparison on Bonn under different input lengths. **Bold**: best result in each column per input-length block.

Method	Input 200 / 300		Input 400 / 500	
	Abs Rel↓	$\delta < 1.25 \uparrow$	Abs Rel↓	$\delta < 1.25 \uparrow$
VGGT (*Off.*) [18]	*OOM / OOM*	*OOM / OOM*	*OOM / OOM*	*OOM / OOM*
StreamVGGT [37]	*OOM / OOM*	*OOM / OOM*	*OOM / OOM*	*OOM / OOM*
CUT3R [20]	0.072 / 0.089	0.947 / 0.938	0.090 / 0.084	0.934 / 0.939
Point3R [25]	0.069 / 0.081	0.954 / 0.946	0.081 / 0.081	0.945 / 0.946
TTT3R [5]	0.068 / 0.079	0.953 / 0.949	0.078 / 0.076	0.951 / 0.953
InfiniteVGGT [33]	0.063 / 0.072	0.964 / 0.958	0.070 / 0.069	0.958 / 0.960
GHOST (ours)	**0.054 / 0.062**	**0.971 / 0.970**	**0.064 / 0.061**	**0.962 / 0.964**

4.2.3 Combined Score

As established in Section 3, token importance is inherently *two-level*: a frame-level component captures whether a viewpoint is geometrically distinctive (large camera displacement, rich depth structure, high recency), while a token-level component identifies which patches *within* that frame carry the most reconstruction value (salient edges, reliable depth and point estimates). Neither level alone is sufficient. A frame-level score applied uniformly across all patches of a highly distinctive viewpoint would waste budget on flat, textureless regions of that frame. Conversely, a token-level score that ignores inter-frame geometry may retain visually active patches from redundant viewpoints while discarding tokens from geometrically critical but low-contrast frames.

The combined per-patch importance score integrates both levels via a weighted sum:

$$\phi(t,p) \;=\; w_f\, s_{\text{frame}}(t) + w_k\, s_{\text{token}}(t,p), \tag{6}$$

where $w_f + w_k = 1$, $w_f, w_k \geq 0$. The additive structure is deliberate: $s_{\text{frame}}(t)$ acts as a *prior* that uniformly elevates or suppresses all patches from frame t according to its global geometric value, while $s_{\text{token}}(t,p)$ provides *within-frame discrimination* by further differentiating patches inside the same frame. A patch therefore receives a high combined score only when it belongs to a geometrically important viewpoint *and* constitutes an informative region within that viewpoint, naturally capturing the two-level structure of token importance identified in our observations. The final score is re-normalised to $[0,1]$ across all candidate tokens before eviction decisions are made.

4.3 Special-Token Privilege Mechanism

Camera tokens $\mathbf{c}_t$ and register tokens $\{\mathbf{r}_t^i\}$ encode global scene geometry state and structural priors. In the StreamVGGT configuration evaluated here, each frame has one camera token followed by four register tokens; these first five positions are known from the architecture rather than predicted by a separate selector. Evicting these tokens can corrupt pose estimation and scene globalisation, yet standard importance scoring may rank them below informative patch tokens due to their low visual saliency. We introduce a *deterministic importance boost* for special tokens:

$$\phi(t,p_{\text{sp}}) \;=\; s_{\text{frame}}(t) + \Delta_{\text{boost}} + \epsilon_{\text{tb}} \cdot r_{p_{\text{sp}}}, \tag{7}$$

where Δ_{boost} raises the priority of special tokens, and $r_{p_{\text{sp}}} \in \{0,1,\ldots,p_0{-}1\}$ is the intra-frame rank of the special token (camera: $r{=}0$; registers: $r{=}1,\ldots,R$). The infinitesimal tiebreak ϵ_{tb} enforces a fully deterministic, reproducible eviction order among special tokens (values given in §5.1). This prioritizes global scene anchors during streaming inference.

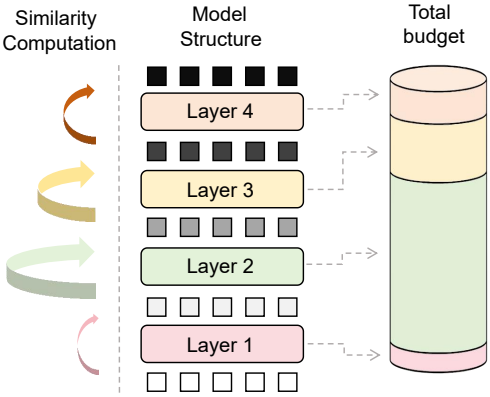


Figure 5: Layer-wise budget allocation guided by cosine similarity. Larger input–output colour discrepancy and larger arrows indicate lower $\bar{\rho}_\ell$; the cylinder shows how B_{total} is distributed, with such layers receiving larger B_ℓ.

4.4 Cosine-Similarity-Guided Layer-Wise Budget Allocation

A global token budget B_{total} must be distributed across L transformer layers. Uniform allocation $B_l = B_{\text{total}}/L$ ignores the heterogeneous importance of different layers.

Offline profiling. We run inference on representative sequences and register forward hooks on each global block ℓ to capture its input $\mathbf{x}_\ell$ and output $\mathbf{y}_\ell$. The mean cosine similarity for layer ℓ across

Table 2: Quantitative comparison on 7-Scenes and NRGBD under different input lengths. **Bold**: best result in each column per input-length block.

Method	Input	7-Scenes						NRGBD					
		Acc.↓		Comp.↓		NC↑		Acc.↓		Comp.↓		NC↑	
		Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.	Mean	Med.
VGGT (*Offline*) [18]	300	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
StreamVGGT [37]		OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
CUT3R [20]		0.135	0.091	0.071	0.032	0.543	0.562	0.224	0.126	0.074	0.012	0.579	0.624
Point3R [25]		0.047	0.027	0.029	0.011	0.563	0.596	0.076	0.043	**0.014**	0.005	0.618	0.695
TTT3R [5]		0.041	0.025	0.024	0.005	0.565	0.599	0.103	0.045	0.025	0.005	0.608	0.673
InfiniteVGGT [33]		0.040	0.015	0.025	0.005	0.570	0.607	0.051	0.032	0.022	0.005	0.649	0.756
GHOST (ours)		**0.023**	**0.006**	**0.022**	**0.004**	**0.573**	**0.612**	**0.046**	**0.020**	0.025	**0.003**	**0.690**	**0.835**
VGGT (*Offline*) [18]	400	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
StreamVGGT [37]		OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
CUT3R [20]		0.162	0.114	0.093	0.050	0.532	0.546	0.315	0.215	0.101	0.032	0.551	0.572
Point3R [25]		0.049	0.023	0.026	0.009	0.559	0.589	0.093	0.045	0.024	0.005	0.613	0.685
TTT3R [5]		0.052	0.031	0.027	0.005	0.558	0.587	0.140	0.070	0.058	0.014	0.599	0.657
InfiniteVGGT [33]		0.043	0.016	0.026	0.005	0.565	0.599	0.069	0.040	0.034	0.005	0.653	0.763
GHOST (ours)		**0.026**	**0.007**	**0.025**	**0.004**	**0.571**	**0.608**	**0.043**	**0.020**	**0.023**	**0.003**	**0.682**	**0.825**
VGGT (*Offline*) [18]	500	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
StreamVGGT [37]		OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
CUT3R [20]		0.183	0.130	0.091	0.033	0.530	0.543	0.326	0.243	0.132	0.042	0.556	0.582
Point3R [25]		0.063	0.026	0.031	0.015	0.555	0.583	0.113	0.048	0.037	0.005	0.613	0.684
TTT3R [5]		0.062	0.036	0.029	0.005	0.552	0.577	0.165	0.084	0.095	0.015	0.594	0.648
InfiniteVGGT [33]		0.043	0.018	0.025	0.005	0.561	0.593	0.080	0.054	0.037	0.008	0.643	0.746
GHOST (ours)		**0.027**	**0.006**	**0.024**	**0.003**	**0.565**	**0.595**	**0.045**	**0.021**	**0.023**	**0.003**	**0.672**	**0.781**

frames and sequences is:

$$\bar{\rho}_\ell \;=\; \frac{1}{|S|}\sum_{s\in S}\frac{\langle \mathbf{x}_\ell^{(s)},\, \mathbf{y}_\ell^{(s)}\rangle}{\|\mathbf{x}_\ell^{(s)}\|\,\|\mathbf{y}_\ell^{(s)}\|}. \tag{8}$$

Layers with high $\bar{\rho}_\ell$ are near-identity and hence redundant.

Budget allocation. Layer importance is $a_\ell = 1 - \bar{\rho}_\ell$. The per-layer budget is:

$$\pi_\ell = \frac{\exp(a_\ell/\tau)}{\sum_{\ell'}\exp(a_{\ell'}/\tau)}, \qquad B_\ell = \big\lfloor \pi_\ell \cdot B_{\text{total}} \big\rfloor. \tag{9}$$

Rounding residuals are assigned to the highest-π_ℓ layer. The budget vector $\{B_\ell\}$ is computed once offline and fixed at inference time. The full eviction procedure is illustrated in Fig 5.

5 Experiments

5.1 Setup

We evaluate on four benchmarks: Bonn [14] (200–500 frames, depth only), 7-Scenes [16] (300–500 frames), NRGBD [1] (300–500 frames), and Long3D [33] (2,128–9,545 frames). Metrics include Accuracy, Completeness, and Normal Consistency (reconstruction), and Abs Rel/δ<1.25 (depth). Baselines are VGGT [18], StreamVGGT [37], CUT3R [20], Point3R [25], TTT3R [5], and InfiniteVGGT [33] (primary comparison). Our GHOST uses B_{total}=1,200,000, τ=0.5, Δ_{boost}=0.3, $\epsilon_{\text{tb}}=10^{-6}$, $(w_{\text{cam}}, w_{\text{geo}}, w_{\text{temp}})=(0.55, 0.55, 0.25)$, w_f=w_k=0.5, and $(w_{\text{sal}}, w_{\text{dc}}, w_{\text{pc}})=(0.28, 0.45, 0.35)$ via grid search on held-out sequences; cosine-similarity profiling uses 13 sequences on one RTX 4090. All experiments run in anchor-1 mode [33].

5.2 Main Results

Tables 1 and 2 report quantitative results on Bonn, 7-Scenes, and NRGBD under different input lengths. Table 3 reports results on the Long3D benchmark. GHOST consistently achieves the best performance across nearly all metrics and input lengths on all four benchmarks.

Bonn. In Table 1, GHOST reduces Abs Rel by up to 14.3% and improves δ<1.25 over InfiniteVGGT across all input lengths, confirming the strategy transfers well to depth estimation.

Table 3: Quantitative comparison on Long3D across five scenes. The scene number indicates the sequence length in frames. **Bold**: best result in each column per scene block.

Method	Scene	Acc.↓		NC↑		CD↓
		Mean	Med.	Mean	Med.	
CUT3R [20]	*Classroom* 2128	0.496	0.374	0.520	0.525	0.291
TTT3R [5]		0.396	0.319	0.530	0.540	0.239
InfiniteVGGT [33]		0.357	0.298	0.576	0.612	0.207
GHOST (ours)		**0.332**	**0.277**	**0.592**	**0.653**	**0.197**
CUT3R [20]	*Dormitory* 4208	1.800	1.372	0.501	0.495	1.102
TTT3R [5]		1.965	1.749	0.515	0.509	1.147
InfiniteVGGT [33]		1.438	1.159	0.526	0.538	1.007
GHOST (ours)		**1.135**	**0.976**	**0.541**	**0.551**	**0.996**
CUT3R [20]	*Library* 4726	1.907	1.437	0.504	0.507	1.050
TTT3R [5]		2.175	1.484	0.494	0.481	1.303
InfiniteVGGT [33]		1.121	0.821	0.508	0.514	0.846
GHOST (ours)		**0.745**	**0.532**	**0.514**	**0.533**	**0.834**
CUT3R [20]	*Badminton Court* 6067	2.489	2.432	0.495	0.483	4.146
TTT3R [5]		2.791	2.392	0.509	0.502	2.975
InfiniteVGGT [33]		1.843	1.555	0.510	0.509	1.848
GHOST (ours)		**1.312**	**1.007**	**0.531**	**0.542**	**1.612**
CUT3R [20]	*Academic Building* 9545	8.062	5.650	0.496	0.491	4.638
TTT3R [5]		7.710	5.793	**0.513**	0.519	6.951
InfiniteVGGT [33]		5.733	4.603	0.495	0.490	3.470
GHOST (ours)		**4.325**	**3.961**	0.511	**0.535**	**3.251**

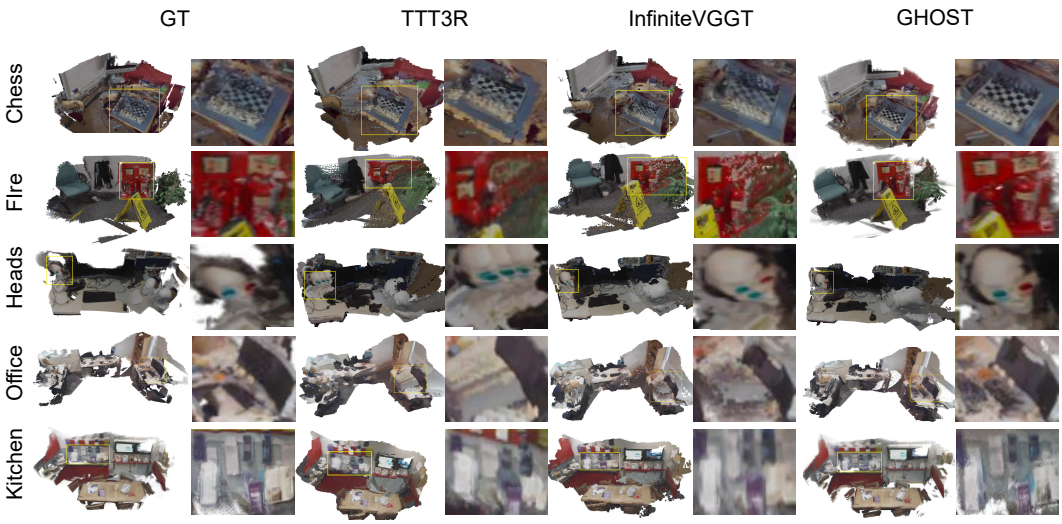


Figure 6: Qualitative reconstruction comparison on 7-Scenes (Chess, Fire, Heads, Office, Kitchen sequences). Columns show GT, TTT3R, InfiniteVGGT, and GHOST results. Yellow boxes highlight fine geometric details where GHOST most closely matches GT.

7-Scenes. In Table 2, at 300 frames, GHOST reduces mean accuracy by 42.5% (0.040→0.023) and improves NC over InfiniteVGGT. The gains hold at 400 and 500 frames, showing geometry-aware eviction retains structurally informative tokens as sequences grow.

NRGBD. In Table 2, while existing methods degrade sharply with sequence length—CUT3R by $+45.5\%$, TTT3R by $+60.2\%$, InfiniteVGGT by $+56.9\%$ from 300 to 500 frames, GHOST remains stable, with accuracy mean staying around 0.045 across all lengths.

Long3D. On this long-sequence benchmark in Table 3, GHOST achieves the best results on nearly every metric. Accuracy gains scale with length: 33.5% on Library and 24.6% on Academic Building, confirming geometry-aware eviction scales gracefully where key-similarity fails.

Table 4: Frame/token importance decomposition on 7-Scenes (300 frames). Lower Acc and Comp, and higher NC, are better.

Score	Acc↓	Comp↓	NC↑
Frame only	0.029	0.024	0.568
Token only	0.033	0.028	0.561
GHOST (both)	**0.023**	**0.022**	**0.573**

Table 7: Importance weight sensitivity on 7-Scenes (300 frames, Acc Mean↓). Perturbations of ± 0.1 applied to each weight group independently.

Group	Frame weights			Token weights			GHOST
Perturbation	$+0.1$ on w_{cam}	-0.1 on w_{geo}	$+0.1$ on w_{temp}	$+0.1$ on w_{sal}	-0.1 on w_{dc}	$+0.1$ on w_{pc}	Default
Acc Mean↓	0.023	0.024	0.024	0.024	0.025	0.023	**0.023**

5.3 Ablation Study

We ablate the two importance levels, layer-wise budget allocation, and the special-token privilege mechanism on 7-Scenes. Sensitivity tests follow below; the signal-level decomposition is in Appendix B.

The two levels are complementary: removing either raises reconstruction error (Table 4). The leave-one-token-signal-out results appear in Appendix G.

Layer-wise budget allocation strategy. Table 5 compares different layer profiling strategies with GHOST's scheme. We evaluate a uniform allocation baseline (evenly allocating the total budget) alongside various network profiling criteria applied to layer-wise budgeting: Fisher information [9], gradient magnitude [7], Hessian-based saliency [30], and magnitude pruning [35]. Uniform allocation and all other profiling alternatives underperform the cosine-similarity criterion on Accuracy, confirming that measuring the transformation strength of each layer provides the most reliable budget distribution signal. The cosine-based score is also the most computationally efficient, requiring no backward passes.

Table 5: Ablation of layer budget allocation strategies on 7-Scenes (300 frames). **Bold**: best result.

Scoring function	Acc↓	Comp↓	NC↑	NC Med↑
Uniform allocation	0.027	0.023	0.565	0.608
Fisher information [9]	0.033	0.024	0.570	0.610
Gradient magnitude [7]	0.028	0.023	0.568	0.611
Hessian saliency [30]	0.026	0.022	0.566	0.608
Magnitude pruning [35]	0.025	0.023	0.571	0.606
Cosine similarity (GHOST)	**0.023**	**0.022**	**0.573**	**0.612**

Special-token privilege. Table 6 shows the effect of disabling the special-token boost, which allows camera and register tokens to compete with patch tokens for the shared budget and be evicted. Without the boost, Accuracy degrades from 0.023 to 0.026, Comp degrades from 0.022 to 0.025 and NC drops from 0.573 to 0.566, consistent with camera and register tokens carrying structural context that matters for reconstruction quality.

Table 6: Effect of special-token privilege on 7-Scenes.

Configuration	Acc↓	Comp↓	NC↑	NC Med↑
w/o special-token boost (Δ_{boost}=0)	0.026	0.025	0.566	0.599
GHOST (w/ boost, Δ_{boost}=0.3)	**0.023**	**0.022**	**0.573**	**0.612**

Importance weight sensitivity. GHOST uses six importance weights: $(w_{\text{cam}}, w_{\text{geo}}, w_{\text{temp}}) = (0.55, 0.55, 0.25)$ for frame-level scoring and $(w_{\text{sal}}, w_{\text{dc}}, w_{\text{pc}}) = (0.28, 0.45, 0.35)$ for token-level scoring. These were found by a lightweight grid search on five held-out sequences and kept fixed for all reported experiments. To verify that GHOST is not sensitive to the precise weight values, we perturb each group by ± 0.1 around the selected values (one group at a time, keeping the other fixed) and measure the change in Accuracy Mean on 7-Scenes (300 frames).

Table 7 reports results under representative perturbations. Across all six independent perturbations, Accuracy Mean remains extremely stable, shifting from the optimal 0.023 to at most 0.025 (e.g., when perturbing w_{dc} by -0.1). This minimal degradation confirms that GHOST is robust to moderate deviations from the tuned weights and that the grid search result does not overfit to the held-out set.

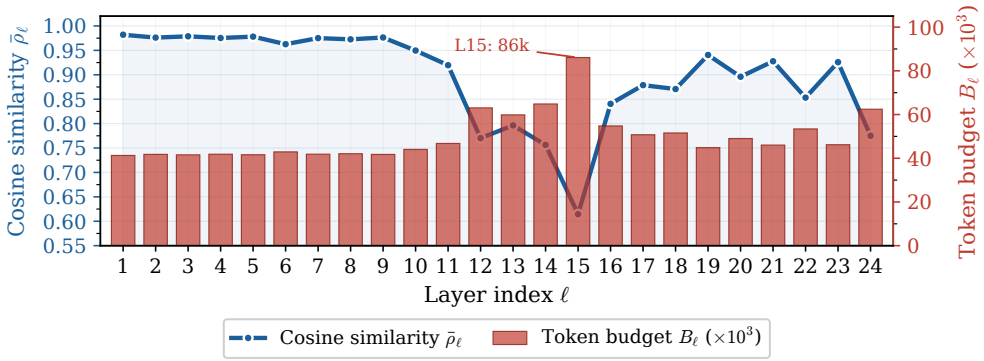


Figure 7: Per-layer cosine similarity $\bar{\rho}_\ell$ (blue) and GHOST budget B_ℓ (orange, τ=0.5). Lower similarity layers receive larger budgets.

5.4 Layer Budget Analysis

Figure 7 shows the cosine-similarity profile of 24 transformer layers from offline profiling on 7-Scenes. Early layers (1–9) show high similarity ($\bar{\rho} \approx 0.95$–1.00) and receive smaller budgets, while the 15th layer with the lowest similarity ($\bar{\rho} \approx 0.61$) gets the largest budget (86,000). This trend indicates that layers with stronger input–output transformations require larger token budgets to preserve informative representations, whereas near-identity layers can operate effectively with substantially fewer tokens.

Budget sensitivity. Table 8 reports GHOST on 7-Scenes (300 frames) under varying budgets. GHOST at B=800,000 surpasses InfiniteVGGT at B=1,200,000, and at B=600,000 (half budget) remains competitive (Acc 0.041 vs. 0.040). At B=600,000, GHOST's KV cache is **3.51 GB** vs. **6.94 GB** (49% reduction) with a 1.75× speedup. These results demonstrate that GHOST achieves a substantially better accuracy–efficiency trade-off by dynamically allocating computation to the most transformation-sensitive layers while aggressively reducing redundant tokens in near-identity layers.

Table 8: Budget sensitivity on 7-Scenes (300 frames). GHOST achieves comparable accuracy at half the budget with significantly lower memory and higher throughput.

Method	Budget	Acc↓	Comp↓	NC↑	NC Med↑	KV Cache (GB)↓	Speedup↑
InfiniteVGGT [33]	1,200,000	0.040	0.025	0.570	0.607	6.94	1.00×
GHOST	1,000,000	**0.023**	**0.022**	**0.573**	**0.613**	5.80	1.38×
GHOST	800,000	0.024	0.022	0.573	0.612	4.65	1.53×
GHOST	600,000	0.041	0.029	0.568	0.604	**3.51**	**1.75**×

6 Conclusion

We presented GHOST, a geometry-hierarchical online streaming token eviction framework for efficient long-sequence 3D reconstruction. By exploiting the model's own depth, point, and camera pose outputs as importance signals, GHOST addresses the fundamental limitation of attention-based eviction, which cannot distinguish historically redundant tokens from geometrically valuable ones. Three innovations work in concert: hierarchical dual-level importance scoring, special-token privilege for structural camera and register tokens, and cosine-similarity-guided layer-wise budget allocation. Experimental results validate that geometry-aware importance signals provide a more principled eviction criterion than attention-based and key similarity-based heuristics, yielding a favorable result across diverse benchmarks and input lengths. Additional evaluations in Appendix G delimit the cross-backbone and dynamic-scene scope of these findings.

Acknowledgments and Disclosure of Funding

Funding. This work received no funding. **Competing interests.** The authors declare no competing interests.

References

[1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *ICCV*, 2022.

[2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

[3] Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Baobao Chang, Junjie Hu, and Wen Xiao. PyramidKV: Dynamic KV cache compression based on pyramidal information funneling. *arXiv:2406.02069*, 2024.

[4] Mengzhao Chen, Wenqi Shao, Peng Xu, Mingbao Lin, Kaipeng Zhang, Fei Chao, Rongrong Ji, Yu Qiao, and Ping Luo. Diffrate: Differentiable compression rate for efficient vision transformers. In *ICCV*, 2023.

[5] Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Ttt3r: 3d reconstruction as test-time training. In *ICLR*, 2026.

[6] Qihua Feng, Peiya Li, Zhixun Lu, Chaozhuo Li, Zefan Wang, Zhiquan Liu, Chunhui Duan, Feiran Huang, Jian Weng, and Philip S Yu. Evit: Privacy-preserving image retrieval via encrypted vision transformer in cloud computing. *TCSVT*, 2024.

[7] Wei Huang, Anda Cheng, and Yinggui Wang. Gradpruner: Gradient-guided layer pruning enabling efficient fine-tuning and inference for llms. *arXiv preprint arXiv:2601.19503*, 2026.

[8] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 2023.

[9] Woosuk Kwon, Sehoon Kim, Michael W Mahoney, Joseph Hassoun, Kurt Keutzer, and Amir Gholami. A fast post-training pruning framework for transformers. In *NeurIPS*, 2022.

[10] Vincent Leroy, Yohann Cabon, and Jerome Revaud. MASt3R: Grounding image matching in 3D with MASt3R. In *ECCV*, 2024.

[11] Huan Li, Longjun Luo, Yuling Shi, and Xiaodong Gu. Analyzing the mechanism of attention collapse in vggt from a dynamics perspective. *arXiv preprint arXiv:2512.21691*, 2025.

[12] Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. SnapKV: LLM knows what you are looking for before generation. In *NeurIPS*, 2024.

[13] Soroush Mahdi, Fardin Ayar, Ehsan Javanmardi, Manabu Tsukada, and Mahdi Javanmardi. Evict3r: Training-free token eviction for memory-bounded streaming visual geometry transformers. *arXiv preprint arXiv:2509.17650*, 2025.

[14] E. Palazzolo, J. Behley, P. Lottes, P. Giguère, and C. Stachniss. ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In *IROS*, 2019.

[15] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In *NeurIPS*, 2021.

[16] 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 *CVPR*, 2013.

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[18] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded deep structured feature transformer. In *CVPR*, 2025.

[19] Junke Wang, Xitong Yang, Hengduo Li, Li Liu, Zuxuan Wu, and Yu-Gang Jiang. Efficient video transformers with spatial-temporal token selection. In *ECCV*, 2022.

[20] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025.

[21] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *CVPR*, 2024.

[22] Junyi Wu, Zhiteng Li, Zheng Hui, Yulun Zhang, Linghe Kong, and Xiaokang Yang. Quantcache: Adaptive importance-guided quantization with hierarchical latent and layer caching for video generation. In *ICCV*, 2025.

[23] Junyi Wu, Zhiteng Li, Haotong Qin, Xiaohong Liu, Linghe Kong, Yulun Zhang, and Xiaokang Yang. Flashedit: Decoupling speed, structure, and semantics for precise image editing. *arXiv preprint arXiv:2509.22244*, 2025.

[24] Junyi Wu, Jiaming Xu, Jinhao Li, Yongkang Zhou, Jiayi Pan, Xingyang Li, and Guohao Dai. Balancegs: Algorithm-system co-design for efficient 3d gaussian splatting training on gpu. In *Asia and South Pacific Design Automation Conference*, 2026.

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

[26] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *ICLR*, 2024.

[27] Jiaming Xu, Jiayi Pan, Hanzhen Wang, Yongkang Zhou, Jiancai Ye, Yu Wang, and Guohao Dai. Specontext: Enabling efficient long-context reasoning with speculative context sparsity in llms. In *ACM International Conference on Architectural Support for Programming Languages and Operating Systems*, 2026.

[28] Ruyi Xu, Guangxuan Xiao, Yukang Chen, Liuning He, Kelly Peng, Yao Lu, and Song Han. Streamingvlm: Real-time understanding for infinite video streams. *ICLR*, 2026.

[29] Yifan Xu, Zhijie Zhang, Mengdan Zhang, Kekai Sheng, Ke Li, Weiming Dong, Liqing Zhang, Changsheng Xu, and Xing Sun. Evo-vit: Slow-fast token evolution for dynamic vision transformer. In *AAAI*, 2022.

[30] Huanrui Yang, Hongxu Yin, Maying Shen, Pavlo Molchanov, Hai Li, and Jan Kautz. Global vision transformer pruning with hessian-aware saliency. In *CVPR*, 2023.

[31] Yifei Yang, Zouying Cao, Qiguang Chen, Libo Qin, Dongjie Yang, Hai Zhao, and Zhi Chen. Kvsharer: Efficient inference via layer-wise dissimilar kv cache sharing. *arXiv preprint arXiv:2410.18517*, 2024.

[32] Hongxu Yin, Arash Vahdat, Jose M Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-vit: Adaptive tokens for efficient vision transformer. In *ICCV*, 2022.

[33] Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. Infinitevggt: Visual geometry grounded transformer for endless streams. *arXiv preprint arXiv:2601.02281*, 2026.

[34] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. In *NeurIPS*, 2020.

[35] Mingxuan Zhang, Yan Sun, and Faming Liang. Magnitude pruning of large pretrained transformer models with a mixture gaussian prior. *Journal of Data Science: JDS*, 2024.

[36] Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, Zhangyang Wang, and Beidi Chen. H2O: Heavy-hitter oracle for efficient generative inference of large language models. In *NeurIPS*, 2023.

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

A GHOST Eviction Algorithm

A.1 Online Incremental Computation

Computing full importance from scratch at every eviction step would require $\mathcal{O}(T^2)$ operations. GHOST maintains an *importance cache* that stores per-frame raw scores:

- On the first eviction, full importance is computed for all candidate frames.
- At subsequent evictions, *only the new frame's* importance is computed and concatenated to retained-frame scores, reducing per-step cost to $\mathcal{O}(N_p)$.
- Camera and geometry raw scores are updated lazily from stored metadata (depth, confidence, pose), since frame outputs become available only after the next forward pass.

This incremental scheme reduces scoring cost by reusing cached metadata; measured runtime overhead should be reported with the final evaluation.

A.2 Full Eviction Procedure

Algorithm 1 GHOST Online Token Eviction at frame t

Require: Metadata $\mathcal{M}=\{(\boldsymbol{\pi}_s, d_s, c_s^d, c_s^p)\}_{s<t}$, current patch tokens $\mathbf{F}_t$, importance cache $\mathcal{C}$, per-layer budgets $\{B_\ell\}$, KV cache $\mathcal{K}$.
Ensure: Updated KV cache $\mathcal{K}'$ with $|\mathcal{K}'_\ell| \le B_\ell$.

for each layer $\ell = 1, \dots, L$ **do**
  if $|\mathcal{K}_\ell| \le B_\ell$ **then**
    continue
  end if
  if cached scores $\mathcal{C}_\ell$ available from prior step **then**
    Compute $\phi(t, \cdot)$ for current frame only via Eqs. (1)–(6)
    $\Phi \leftarrow [\mathcal{C}_\ell;\ \phi(t, \cdot)]$
  else
    Compute full $\Phi = \{\phi(s, p)\}$ for all candidate frames
  end if
  Apply special-token boost via Eq. (7)
  $\mathrm{idx} \leftarrow \mathrm{TopK}(\Phi,\ B_\ell)$
  $\mathcal{K}'_\ell \leftarrow \mathcal{K}_\ell[\mathrm{idx}]; \quad \mathcal{C}_\ell \leftarrow \Phi[\mathrm{idx}]$
end for
Append current frame KV pairs to each $\mathcal{K}'_\ell$.
return $\mathcal{K}'$, updated $\mathcal{C}$

B Extended Ablation: Dual-Level Decomposition

The main paper ablates the importance estimation strategy and the special-token privilege mechanism. Here we provide a finer decomposition ablation to justify the hierarchical dual-level design of the importance score $\phi(t, p)$ (Eq. (6)).

Frame-level vs. token-level vs. combined. Table 9 isolates the contribution of each level by replacing the full score with its frame-only or token-only counterpart. Using frame-level importance alone ($\phi = s_{\text{frame}}$) assigns a uniform importance to all patches within a frame, ignoring within-frame spatial variation (Accuracy degrades from 0.023 to 0.029, and Normal Consistency drops from 0.573 to 0.568). Using token-level importance alone ($\phi = s_{\text{token}}$) captures patch saliency but disregards whether the parent frame is geometrically distinctive (Accuracy degrades further to 0.033, and Normal Consistency to 0.561). GHOST combines both, consistently outperforming either ablated variant across all four metrics, confirming that the two levels are complementary and jointly necessary.

Frame-level sub-component ablation. Table 10 ablates the three sub-signals of the frame-level score individually: camera pose change s_{cam}, depth gradient variance s_{geo}, and temporal recency s_{temp}. Each is removed in turn (weight set to zero, remaining weights renormalised) to isolate its marginal

Table 9: Ablation of hierarchical dual-level decomposition on 7-Scenes (300 frames). **Bold**: best result.

Configuration	Acc↓	Comp↓	NC↑	NC Med↑
Frame-level only ($\phi = s_{\text{frame}}$)	0.029	0.024	0.568	0.607
Token-level only ($\phi = s_{\text{token}}$)	0.033	0.028	0.561	0.598
GHOST (frame + token)	**0.023**	**0.022**	**0.573**	**0.612**

contribution. Removing camera pose change (s_{cam}) or temporal recency (s_{temp}) causes Accuracy to worsen from 0.023 to 0.026 and 0.028 respectively. Removing depth gradient variance (s_{geo}) yields the most severe drop, degrading Accuracy to 0.030 and Normal Consistency from 0.573 to 0.558, indicating that structural variation is the strongest single indicator of a frame's long-term geometric value.

Table 10: Ablation of frame-level importance sub-components on 7-Scenes (300 frames). **Bold**: best result.

Configuration	Acc↓	Comp↓	NC↑	NC Med↑
w/o s_{cam} (no pose change)	0.026	0.025	0.569	0.605
w/o s_{geo} (no depth variance)	0.030	0.026	0.558	0.595
w/o s_{temp} (no recency)	0.028	0.025	0.566	0.596
GHOST (all three)	**0.023**	**0.022**	**0.573**	**0.612**

C Extended Ablation: Temperature Sensitivity

The layer-wise budget allocation in GHOST is governed by the temperature parameter τ (Eq. (9)), which controls how aggressively the total budget is redistributed across layers according to their cosine-similarity profiles. A higher τ produces a more uniform distribution; a lower τ concentrates budget more heavily on early, high-transformation layers.

Table 11 reports GHOST performance on 7-Scenes (300 frames) under four values of τ. Performance is remarkably robust across a wide range ($\tau \in [0.3, 0.7]$), with Accuracy fluctuating only marginally between 0.023 and 0.025, and Normal Consistency between 0.571 and 0.573. This confirms that the offline profiling result generalises and the method is not sensitive to the precise choice of τ. We use $\tau = 0.5$ in all main-paper experiments.

Table 11: Sensitivity of GHOST to temperature τ on 7-Scenes (300 frames). **Bold**: best result.

τ	Acc↓	Comp↓	NC↑	NC Med↑
0.3	0.024	0.022	0.572	0.610
0.5 (ours)	**0.023**	**0.022**	**0.573**	**0.612**
0.7	0.025	0.023	0.571	0.608
1.0	0.025	0.024	0.572	0.610

D Comparison with LLM KV Cache Eviction Methods

A natural question is whether general-purpose KV cache eviction methods developed for large language models (LLMs) can be applied to streaming 3D reconstruction. Table 12 summarises the key differences between GHOST and representative LLM KV cache methods.

LLM KV cache methods rely exclusively on attention-based signals (accumulated attention scores, recent token windows, or observation-window attention patterns) to decide which tokens to evict. These signals are query-dependent and context-agnostic: they measure how much the *current* query attends to past tokens, which correlates poorly with the long-term geometric value of historical 3D

Table 12: Conceptual comparison between GHOST and LLM KV cache eviction methods.

Method	Domain	Importance Signal	Geometry-Aware	Layer-Wise Budget
H2O [36]	LLM	Attention score accumulation	✗	✗
StreamingLLM [26]	LLM	Fixed sink + recent tokens	✗	✗
SnapKV [12]	LLM	Observation window attention	✗	✗
PyramidKV [3]	LLM	Layer-wise attention budget	✗	✓
InfiniteVGGT [33]	3D	Key-query cosine similarity	✗	✗
GHOST (ours)	3D	Geometry-hierarchical score	✓	✓

observations (Spearman $\rho = -0.07$ with pose change, $\rho = +0.31$ with depth gradient variance, as shown in the main paper).

GHOST differs in two fundamental respects. First, the importance signal is *geometry-grounded*: it is derived from the model's own 3D outputs (depth maps, point confidence, camera poses) rather than from attention patterns. This makes the score query-independent and reflective of a token's long-term structural value. Second, GHOST introduces *geometry-aware layer-wise budget allocation* guided by offline cosine-similarity profiling, which concentrates the token budget where transformer layers perform the strongest transformations. PyramidKV [3] also allocates budgets layer-wise, but based on attention entropy rather than transformation strength, and does not exploit any domain-specific structure. These design differences make direct application of LLM KV cache methods to streaming 3D reconstruction ineffective, and motivate the geometry-specialised design of GHOST.

E Broader Applicability of GHOST

The demonstrated implementation uses an accessible causal historical KV cache with frame-associated tokens and geometry outputs. Appendix G reports a separate transfer test on STream3R under a matched KV footprint. This evidence supports transfer to that backbone under the stated protocol; it does not establish compatibility with arbitrary reconstruction architectures.

Prerequisites. GHOST's frame-level importance score $s_{\text{frame}}(t)$ requires three signals: (i) a camera pose estimate $\boldsymbol{\pi}_t$ (translation, rotation, focal length), (ii) a per-frame depth map d_t, and (iii) per-patch confidence maps c_t^d, c_t^p for depth and point reconstruction quality. These outputs are produced by the StreamVGGT backbone evaluated here. The online scoring and top-K selection still have a measurable cost, quantified in Appendix G.

The token-level importance score $s_{\text{token}}(t, p)$ requires patch-level depth and point confidence maps from the chosen backbone. The visual saliency signal $s_{\text{sal}}(t, p)$ is computed from the patch feature map. Offline or pairwise architectures such as VGGT, MASt3R, and DUSt3R cannot use this policy directly without a suitable causal cache interface.

Proxy signals for models without geometry heads. For a backbone without geometry heads, the author-response experiment in Appendix G tests an image-gradient-saliency and temporal-recency proxy. Its lower accuracy bounds the claim of geometry-free transfer. Optical flow and attention entropy remain untested possible extensions.

Limitations. The frame-level score assumes predominantly static scenes. On the tested dynamic MPI-Sintel sequences, GHOST improves over Evict3R only modestly at the matched budget and remains less accurate than the full cache (Appendix G). Explicit motion separation is a possible future extension, not a component of the evaluated method.

F Long-Sequence Scalability Analysis

A key motivation for GHOST is that key-similarity eviction degrades disproportionately as sequences grow longer, because the current query is increasingly remote from early frames, causing them to receive systematically low similarity scores and be evicted even when they contain unique scene content. We verify this hypothesis using the Long3D benchmark, which spans sequences from 2,128 to 9,545 frames—an order of magnitude longer than the 7-Scenes and NRGBD sequences in the main paper.

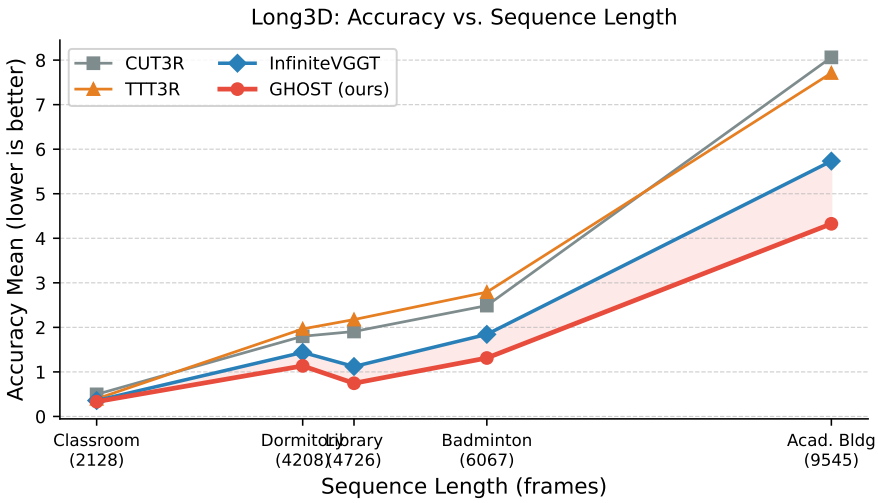


Figure 8: Accuracy Mean (↓) versus sequence length on the Long3D benchmark. The shaded region highlights the gap between GHOST and InfiniteVGGT. GHOST's advantage over key-similarity eviction (InfiniteVGGT) grows with sequence length, confirming that geometry-aware eviction scales more gracefully to very long sequences.

Figure 8 plots Accuracy Mean against sequence length for all methods. Two trends are evident. First, all methods degrade as sequence length increases, reflecting the growing difficulty of retaining a fixed-budget representation of an increasingly long history. Second, the *gap* between GHOST and InfiniteVGGT widens with sequence length: at 2,128 frames (Classroom), GHOST improves Accuracy by 7.0% (0.357→0.332); at 4,726 frames (Library), the improvement reaches 33.5% (1.121→0.745); and at 9,545 frames (Academic Building), 24.6% (5.733→4.325). This superlinear scaling of the benefit confirms the core hypothesis: the longer the sequence, the more severely key-similarity eviction misfires by discarding geometrically distinctive early frames, and the more geometry-aware eviction pays off.

CUT3R and TTT3R exhibit particularly severe degradation at extreme sequence lengths (Academic Building: 8.062 and 7.710 respectively), as their sequential update strategies accumulate drift without a principled mechanism to preserve globally informative historical observations. GHOST's geometry-hierarchical scoring explicitly accounts for viewpoint coverage and depth diversity across the full sequence history, yielding substantially more graceful degradation.

G Additional Evaluations from the Author Response

The following evaluations use the protocols and measurements reported in our author response. They distinguish tested transfer and robustness from possible future extensions.

Cross-backbone transfer. We implemented GHOST scoring in STream3R, a decoder-only causal reconstruction Transformer. On 7-Scenes at 518×392 resolution, stride 5, and 200–300 frames per clip, we hold the checkpoint, frame order, resolution, and precision fixed. We match the measured KV footprint, since token counts are not directly comparable across backbones (Table 13).

Table 13: STream3R transfer on 7-Scenes. The GHOST and Window-22 rows use approximately the same KV memory.

Policy	Acc mean/med↓	Comp mean/med↓	NC mean/med↑	KV GB↓	FPS↑
Full cache	0.044 / 0.013	0.024 / 0.007	0.606 / 0.666	19.75	5.04
Window-8	0.107 / 0.037	0.035 / 0.008	0.587 / 0.635	0.86	13.20
Window-22	0.102 / 0.029	0.029 / 0.006	0.594 / 0.647	2.19	10.59
GHOST (ours)	0.078 / 0.024	0.027 / 0.006	0.598 / 0.656	2.19	10.44

At 2.19 GB, GHOST improves reconstruction metrics relative to Window-22, with slightly lower FPS. This establishes transfer under this specific causal cache interface; it does not establish direct transfer to offline or pairwise models.

Dynamic scenes. We evaluate the unchanged GHOST policy on official dynamic MPI-Sintel sequences using the same StreamVGGT checkpoint, ordered frames, and a fixed matched budget (Table 14). The method's frame score was designed for predominantly static scenes. Separating object motion from camera motion could improve this setting, but is not implemented in the reported method.

Table 14: Dynamic MPI-Sintel evaluation. GHOST and Evict3R have matched KV memory.

Method	AbsRel↓	$\delta < 1.25 \uparrow$	KV GB↓	ms/frame↓
Full cache	0.328	0.649	7.2	145
Evict3R	0.341	0.631	3.5	94
GHOST	0.337	0.635	3.5	91

Matched-budget Evict3R comparison. We compare against the public Evict3R policy using the same StreamVGGT checkpoint, official 7-Scenes test split, ordered 300-frame inputs, resolution 518, precision, and retained-token budget (Table 15).

Table 15: GHOST and Evict3R at matched budgets on 7-Scenes.

Method	Budget	Acc↓	Comp↓	NC↑	KV GB↓
Evict3R	1.2M	0.029	0.024	0.564	7.26
GHOST	1.2M	0.023	0.022	0.573	7.26
Evict3R	0.6M	0.067	0.033	0.552	3.51
GHOST	0.6M	0.041	0.029	0.568	3.51

Online overhead. Synchronized timings after warm-up separate frame/token scoring, budget allocation, and top-K selection. The layer-similarity profile is computed once offline, outside online latency. We define t_{online} as the sum of the three measured components and ρ_{overhead} as its fraction of end-to-end time (Table 16).

Table 16: GHOST online overhead. Times are ms/frame, except the final percentage.

Frames	Frame+token	Allocation	Top-K	Online total	Overhead
100	1.0	< 0.01	7.7	8.7	4.1%
300	0.7	< 0.01	12.6	13.3	5.1%
500	0.6	< 0.01	13.5	14.1	5.3%

Thus, online scoring and selection have a measurable 4.1–5.3% cost in the tested input-length range.

Proxy signals without geometry heads. Under the same checkpoint, frame order, and budget, an image-gradient-saliency plus temporal-recency policy is less accurate than full GHOST (Table 17). This is a tested proxy, not evidence that GHOST transfers to every architecture without geometry outputs.

Table 17: Geometry-free proxy ablation on 7-Scenes.

Policy	Geometry heads	Acc↓	Comp↓	NC↑
Full GHOST	Required	0.023	0.022	0.573
Saliency + recency	Not required	0.034	0.029	0.551

Token-level contributions and special-token retention. The leave-one-signal-out tests use the same 7-Scenes, 300-frame, 1.2M-token protocol as the main ablations (Table 18). The special tokens are the architecture-defined camera token and four register tokens. Their measured retention increases from 67% without the priority boost to 92% with it (Table 19); this is prioritization, not a guarantee of retention.

Table 18: Leave-one-token-signal-out ablation.

Variant	Acc↓	Comp↓	NC↑
Full GHOST	0.023	0.022	0.573
Without visual saliency	0.028	0.025	0.554
Without depth confidence	0.030	0.026	0.549
Without point confidence	0.027	0.025	0.552

Table 19: Measured retention of camera and register tokens on 7-Scenes.

Boost Δ_{boost}	Special-token retention↑	Acc↓	NC↑
0.3	92%	0.023	0.573
0	67%	0.026	0.566

H Evaluation Metric Definitions

Let P and G be reconstructed and ground-truth point sets, respectively. With nearest-neighbor indices $j(i) = \arg\min_j \|p_i - g_j\|_2$ and $i(j) = \arg\min_i \|g_j - p_i\|_2$, Accuracy and Completeness are

$$\text{Acc}(P, G) = \frac{1}{|P|} \sum_{p_i \in P} \min_{g_j \in G} \|p_i - g_j\|_2, \tag{10}$$

$$\text{Comp}(P, G) = \frac{1}{|G|} \sum_{g_j \in G} \min_{p_i \in P} \|g_j - p_i\|_2. \tag{11}$$

For corresponding estimated and ground-truth unit normals, bidirectional Normal Consistency is

$$\text{NC}(P, G) = \frac{1}{2} \left[\frac{1}{|P|} \sum_i \left| \langle n_i^P, n_{j(i)}^G \rangle \right| + \frac{1}{|G|} \sum_j \left| \langle n_j^G, n_{i(j)}^P \rangle \right| \right]. \tag{12}$$

Tables specify mean or median aggregation across evaluated scenes.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction precisely state the three technical contributions of GHOST—hierarchical dual-level importance scoring, special-token privilege mechanism, and cosine-similarity-guided layer-wise budget allocation—and the evaluation scope (7-Scenes, NRGBD, Bonn, and Long3D benchmarks, using StreamVGGT as backbone). All quantitative claims (e.g., up to $1.75\times$ throughput speedup, 49% KV cache reduction, consistent outperformance over InfiniteVGGT) are directly supported by Tables 2–8 in §5.2–5.4.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix E elaborates on the limitations of the GHOST method. It states that GHOST cannot be directly applied to architectures without causal structure and models not bottlenecked by KV cache. Additionally, it points out the challenge of extending GHOST to dynamic scenes, since the current frame-level scoring mechanism is designed for static scenes and has defects in dynamic scenarios.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: GHOST is an algorithmic framework without formal theorems or proofs. All design choices are motivated empirically or by well-established geometric principles, with ablation experiments in §5.3 and Appendix B providing empirical validation.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: §5.1 provides all hyperparameters: B_{total}=1,200,000, τ=0.5, Δ_{boost}=0.3, ϵ_{tb}=10^{-6}, $(w_{\text{cam}}, w_{\text{geo}}, w_{\text{temp}})$=$(0.55, 0.55, 0.25)$, $(w_{\text{sal}}, w_{\text{dc}}, w_{\text{pc}})$=$(0.28, 0.45, 0.35)$, w_f=w_k=0.5. Profiling details (13 sequences, RTX 4090), dataset splits, input-length regimes, and evaluation metrics are specified in §5.1 and Appendix H. Algorithm 1 gives the eviction procedure.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The project code is publicly available at `https://github.com/lokiniuniu/GHOST`. The authors confirm that its instructions and scripts are sufficient to reproduce the reported results; the datasets are credited and can be obtained from their original providers.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

Justification: GHOST requires no additional model training, but its hyperparameters are selected by grid search on five held-out sequences and its layer budgets use an offline profile. Input-length regimes, hardware (RTX 4090, anchor-1 mode), and evaluation protocols are specified in §5.1 and Appendix G.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [N/A]

   Justification: GHOST is fully deterministic (no randomness in eviction decisions). Results are means over fixed test sequences (18 for 7-Scenes, 8 for NRGBD, standard Bonn sequences, 5 scenes for Long3D). Error bars across seeds are not applicable. Robustness is instead evidenced by consistent improvements across all four independent benchmarks, all input lengths, and all evaluation metrics.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: §5.1 specifies the GPU (RTX 4090). Memory footprint and throughput figures are reported in §5.4 (e.g., 3.51 GB at B=600,000 vs. 6.94 GB for InfiniteVGGT; $1.75\times$ speedup). The measured online overhead is reported in Appendix G. Offline cosine-similarity profiling is a separate one-time cost.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This work proposes an algorithmic efficiency improvement for long-sequence 3D reconstruction. It involves no human subjects, no sensitive personal data, and no foreseeable dual-use risks beyond those generally associated with 3D scene reconstruction technology.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: GHOST reduces the memory and compute requirements for long-sequence 3D reconstruction, enabling deployment on lower-memory hardware and benefiting applications in robotics, AR/VR, and autonomous navigation. No specific negative societal impact beyond standard concerns about 3D reconstruction technology is foreseen.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: This work releases only a KV cache eviction algorithm that operates on an existing publicly available model checkpoint. No new model weights, training data, or high-risk assets are introduced.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

Justification: The paper credits 7-Scenes [16], NRGBD [1], Bonn [14], Long3D [33], and the baseline methods. Microsoft Research restricts 7-Scenes to non-commercial use; the original NRGBD provider lists its synthetic scene assets under CC-BY or CC0; the Bonn provider requests citation; InfiniteVGGT code is Apache-2.0. The Long3D dataset page does not list a separate license. The authors confirm that they checked and followed the applicable provider terms for all datasets, pretrained checkpoints, and baseline code.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The GHOST algorithm is fully specified in Algorithm 1 (Appendix A) with all inputs, outputs, and intermediate steps described. The method section (§4) provides the mathematical formulation of all scoring components. The authors confirm that the public code repository documents the method and provides instructions and scripts sufficient to reproduce all reported results.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing or human subjects are involved in this work.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subjects are involved in this work.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

    Answer: [N/A]

    Justification: The authors confirm that no LLM or agent is an important, original, or non-standard component of the core method.