T-LRM: TEMPORAL-AWARE LRM FOR MULTI-VIEW DYNAMIC SCENE RECONSTRUCTION

Anonymous authors
Paper under double-blind review

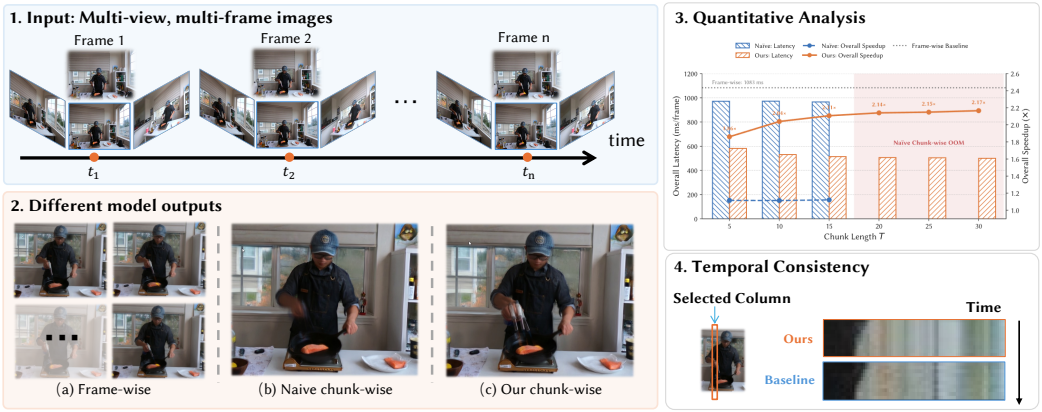


Figure 1: Unlike (a) frame-by-frame multi-view dynamic scene reconstruction, which fails to exploit temporal redundancy, and (b) naïve joint processing of multi-timestamp inputs, which can suffer from cross-temporal interference, our method (c) enables temporal-aware chunk-wise reconstruction of multi-timestamp, multi-view inputs, achieving multi-fold inference acceleration while preserving temporal consistency.

ABSTRACT

Large Reconstruction Models (LRMs) have recently emerged as the dominant paradigm for feed-forward 3D reconstruction. However, existing 3D foundation models are primarily designed for static scenes or monocular dynamic inputs, making them difficult to extend to high-quality multi-view dynamic scene reconstruction. A straightforward frame-by-frame inference strategy is computationally expensive and fails to exploit the strong temporal redundancy inherent in multi-view videos. To address these limitations, we propose T-LRM, a Temporal-aware LRM framework that extends existing 3D foundation models to temporally aware multi-view dynamic reconstruction in a training-free manner. Our key insight is that multi-view dynamic videos contain substantial spatial redundancy across time. By identifying dynamic regions while sharing temporally consistent static tokens, redundant computation can be substantially reduced while avoiding direct interactions between dynamic tokens from different timestamps. Based on this observation, we introduce a Temporal-aware Token Merging strategy that averages static tokens across timestamps for shared processing while preserving dynamic tokens for timestamp-specific processing. The proposed framework is lightweight, plug-and-play, and can be seamlessly integrated into different feed-forward reconstruction architectures without retraining. Extensive experiments on multiple benchmark datasets demonstrate that T-LRM achieves multi-fold inference acceleration while maintaining comparable reconstruction quality.

1 INTRODUCTION

Dynamic scene reconstruction has attracted increasing attention due to its applications in volumetric video, digital twins, and virtual reality. Compared with monocular reconstruction, synchronized multi-view capture provides substantially richer geometric observations, enabling high-fidelity

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

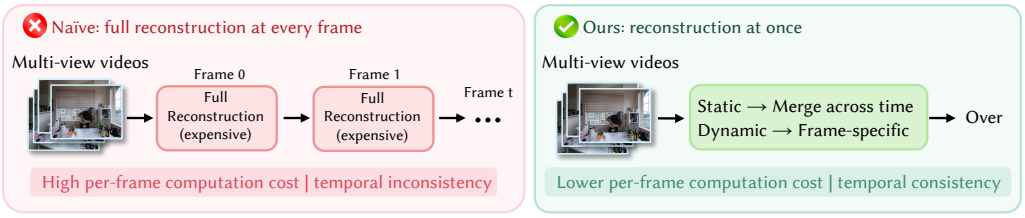


Figure 2: Comparison between frame-by-frame and temporal-aware inference. Our approach shares static regions across timestamps while preserving timestamp-specific global token interaction for dynamic regions, avoiding redundant full reconstruction.

free-viewpoint rendering. However, efficient feed-forward reconstruction from multi-view dynamic videos remains largely unexplored. In this work, we focus on this setting, aiming to efficiently reconstruct dynamic scenes from synchronized multi-view video sequences.

Recent advances in Large Reconstruction Models (LRMs) have significantly improved the efficiency of 3D reconstruction. Unlike optimization-based methods that require per-scene iterative optimization, feed-forward LRMs reconstruct a scene in a single forward pass and have become the dominant paradigm for large-scale reconstruction. Representative works include geometry reconstruction models such as DUSt3R Wang et al. (2024b) and VGGT Wang et al. (2025a), as well as feed-forward novel view synthesis methods including LRM Hong et al. (2023), LVSM Jin et al. (2024), ZipMap Jin et al. (2026), DepthSplat Xu et al. (2025a), and recent long-context reconstruction models. Despite their success, these methods are primarily designed for static scenes, while existing feed-forward dynamic reconstruction methods still largely focus on monocular videos. Efficient reconstruction from multi-view dynamic videos remains largely unexplored. Extending these models from static images to dynamic multi-view videos is non-trivial.

A straightforward solution is to reconstruct each timestamp independently, as shown in Fig. 2. However, such **frame-by-frame inference** repeatedly processes nearly identical scene content, leading to substantial redundant computation while ignoring the strong temporal redundancy inherent in multi-view videos. Another line of work introduces additional temporal modeling networks to aggregate information across frames, but these modules inevitably increase inference cost and model complexity. More importantly, existing methods treat all temporal tokens equally, although only a small fraction of the scene changes over time while the majority remains temporally consistent.

Our key observation is that **temporal computation should be proportional to scene changes rather than sequence length**. Static regions provide consistent geometric evidence across time and therefore require only a single global token interaction (instantiated as either TTT Update or Global Attention), whereas dynamic regions should remain temporally independent to avoid cross-temporal interference. Motivated by this observation, we propose T-LRM, a general temporal-aware framework for feed-forward multi-view dynamic reconstruction. Specifically, we introduce a Temporal-aware Token Merging strategy that identifies temporally consistent regions, shares static tokens across timestamps, and preserves dynamic tokens for timestamp-specific global token interaction. By transforming conventional **time-proportional inference** into **change-proportional inference**, our framework substantially reduces redundant computation while maintaining reconstruction fidelity and temporal consistency.

Importantly, T-LRM is lightweight and training-free. It reorganizes token computation without modifying network architectures or requiring additional training, and is compatible with both TTT-based and attention-based reconstruction models. Extensive experiments demonstrate consistent multi-fold inference acceleration with competitive reconstruction quality and improved temporal consistency.

(1) We propose T-LRM, a temporal-aware framework for efficient feed-forward reconstruction of synchronized multi-view dynamic scenes under fixed camera settings, without retraining.

(2) We introduce a Temporal-aware Token Merging strategy that exploits temporal redundancy by sharing static tokens while preserving timestamp-specific dynamic tokens, substantially reducing redundant computation.

(3) As a lightweight and plug-and-play framework, T-LRM$\tilde{\text{i}}$s compatible with both TTT-based and attention-based reconstruction backbones, achieving multi-fold inference acceleration with competitive reconstruction quality and improved temporal consistency.

2 Related Works

2.1 Feed-Forward 3D Reconstruction

Compared with optimization-based methods such as NeRF Mildenhall et al. (2021) and 3D Gaussian Splatting Kerbl et al. (2023), feed-forward 3D reconstruction has recently emerged as the dominant paradigm due to its ability to reconstruct scenes in a single forward pass. Early works, such as DUSt3R Wang et al. (2024b), directly regress view-consistent point maps from image pairs without requiring camera calibration, while VGGT further extends this paradigm to arbitrary numbers of input views by jointly predicting camera parameters, dense point maps, and 2D correspondences. Building upon these models, a series of follow-up works have explored more efficient and robust feed-forward geometry reconstruction, including MASt3R, Stream3R Lan et al. (2025), Fast3R Yang et al. (2025), Test3R Yuan et al. (2025), Point3R Wu et al. (2025c), DepthAnything3 Lin et al. (2025), YoNoSplatYe et al. (2026), π^3Wang et al. (2025d), StreamVGGTZhuo et al. (2026), AnySplat Jiang et al. (2025b) and TTT3R Chen et al. (2025b).

Feed-forward reconstruction has also achieved remarkable progress in novel view synthesis. Earlier approaches typically relied on convolutional networks to predict explicit scene representations, such as multi-plane images, voxels, or point clouds. More recently, transformer-based Large Reconstruction Models (LRMs) have substantially improved reconstruction quality by directly learning 3D representations from multi-view images. Representative methods include LRM, GS-LRM Zhang et al. (2024b), PixelSplat Charatan et al. (2024), MVSplat Chen et al. (2024), LGM Tang et al. (2024), DepthSplat Xu et al. (2025a), LVSM Jin et al. (2024), Long-LRM Ziwen et al. (2025), TTT-LRM Wang et al. (2026), Rayzer Jiang et al. (2025a); Zhao et al. (2026) and ZipMap Jin et al. (2026), which demonstrate the effectiveness of large-scale feed-forward reconstruction. Despite these advances, most existing methods are designed for static scenes, leaving efficient reconstruction of dynamic multi-view videos largely unexplored.

2.2 Monocular and Multi-view 4D Reconstruction

Dynamic 3D reconstruction can generally be divided into *monocular* and *multi-view* settings.

Optimization-based methods. Optimization-based approaches have achieved remarkable success in both settings. Representative monocular methods, such as D-NeRF, Deformable-3DGS Yang et al. (2024), and SCGS Huang et al. (2023), Shape of Motion Wang et al. (2024a) reconstruct dynamic scenes by jointly optimizing geometry and appearance from a single video. In the multi-view setting, methods including SpaceTimeGS Li et al. (2024), Swift4DWu et al. (2025b), LongVolcapXu et al. (2024), and FreeTimeGS Wang et al. (2025c) leverage synchronized camera observations to achieve high-fidelity reconstruction. Several approaches, such as HexPlane Cao & Johnson (2023), K-Planes Fridovich-Keil et al. (2023), 4DGaussian Wu et al. (2024), and 4DGS Yang et al. (2023), are applicable to both monocular and multi-view scenarios through unified spatio-temporal representations. Overall, optimization-based methods have established strong reconstruction quality, although they usually require scene-specific optimization and therefore suffer from high computational cost.

Learning-based methods. In contrast, learning-based methods reconstruct dynamic scenes in a feed-forward manner, enabling significantly faster inference than optimization-based approaches. Recent methods, including CUT3R Wang et al. (2025b), TTT3R Chen et al. (2025b), Page-4D Zhou et al. (2025), VGGT-4D Hu et al. (2025), 4DGTXu et al. (2025b) and DVGGT Chen et al. (2025a), L4GM Ren et al. (2024), Monst3RZhang et al. (2024a), UFO-4D Hur et al. (2026) have demonstrated promising performance. However, most existing approaches adopt a streaming or frame-by-frame reconstruction paradigm, repeatedly encoding temporally consistent scene content across adjacent frames and thus incurring substantial redundant computation. **In comparison, feed-forward reconstruction from multi-view dynamic videos remains relatively underexplored.** Existing approaches can be broadly categorized into two directions. SplatVoxel Wang et al. (2025e) extends feed-forward static reconstruction to dynamic scenes through history-aware inference, warping historical Gaussian primitives and fusing them with current observations to improve temporal consis-

tency. NoPo4D Balice et al. (2026) introduces an explicit temporal aggregation module to model cross-frame dependencies. These approaches leverage temporal information through historical reconstruction fusion or cross-frame feature aggregation. In contrast, our method explicitly exploits temporal redundancy through a temporal-aware token merging strategy. By separating static and dynamic regions, temporally consistent tokens are shared across timestamps while dynamic tokens are preserved, substantially reducing redundant computation without introducing additional trainable modules.

2.3 Token Merging

Token merging has emerged as an effective training-free strategy for improving the efficiency of Vision Transformers (ViTs) by reducing redundant visual tokens. Pioneered by ToMe Bolya et al. (2023); Bolya & Hoffman (2023), this line of work merges semantically similar tokens based on feature similarity, significantly reducing computational cost while preserving model performance. Subsequent studies have extended token merging to a wide range of applications, including diffusion models, video understanding, vision-language models, and video editing, through techniques such as token unmerging, pruning, and spatio-temporal merging. These methods further exploit the inherent redundancy across spatial and temporal dimensions to accelerate inference with minimal performance degradation. Despite their success, existing token merging approaches are primarily designed for recognition or generation tasks, and their application to feed-forward 3D reconstruction remains largely unexplored, especially in dynamic scene reconstruction.

3 Methods

3.1 Problem Formulation and Complexity Analysis

We consider the established multi-view dynamic reconstruction setting in which a scene is captured by a synchronized and calibrated fixed camera rig, as commonly adopted in volumetric video and free-viewpoint video systems Xu et al. (2023a; 2024); Wu et al. (2025a); Wang et al. (2023). Given a dynamic multi-view image chunk with T timestamps, V calibrated views, and P image tokens per view, the input to a Large Reconstruction Model (LRM) contains $L = T \times V \times P$ tokens. The fixed-camera setting provides spatial alignment at corresponding image locations across timestamps, enabling temporal redundancy to be identified without additional camera-motion compensation. However, many tokens correspond to temporally consistent regions, causing repeated global token interaction on largely unchanged scene content and introducing substantial computational redundancy.

Our key observation is that global token interaction should account for **scene changes** rather than uniformly repeat computation across the entire **sequence**. Instead of uniformly processing all tokens, we first identify temporally stable regions and decouple them from dynamic ones. Within each global interaction block, static tokens are processed once and shared across time, while dynamic tokens retain timestamp-specific global token interaction. This design substantially reduces redundant computation while preserving timestamp-specific scene dynamics.

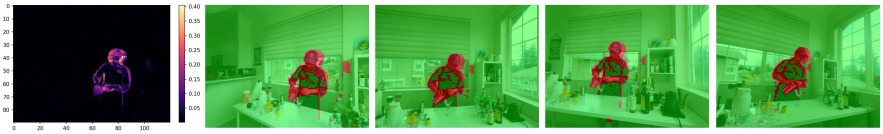
Formally, let S and D denote the numbers of static and dynamic tokens per timestamp. Conventional frame-by-frame inference processes $T(S + D)$ tokens in each global interaction block, whereas our method processes $S + TD$ tokens. The relative reduction in the number of tokens processed is therefore

$$1 - \frac{S + TD}{T(S + D)} = \frac{S(T - 1)}{T(S + D)} \approx \frac{S}{S + D}, \tag{1}$$

where the approximation holds for sufficiently large T. For a given static-token proportion, this reduction increases with the temporal chunk length; for a given chunk length, it increases with the proportion of static scene content. This token-count reduction provides an indicator of computational savings, while the actual speedup depends on the global interaction mechanism and implementation.

3.2 Patch Classification

We first identify temporally stable patches to separate static scene content from dynamic regions. Since the cameras remain fixed throughout the sequence, patches at the same spatial location are



(a) Temporal diff. (b) Dynamic-static patches

Figure 3: Visualization of static and dynamic patch classification. Static regions account for the majority of image patches, leaving only a small fraction as dynamic. This motivates reusing static tokens while updating only dynamic ones, thereby improving inference efficiency.

naturally aligned across timestamps. We therefore use temporal appearance variation as a simple and effective cue for motion detection. The classification is performed at the patch level, consistent with the token granularity of the reconstruction model.

Let $\mathbf{X}_t^p$ denote the p-th patch at timestamp t, where p implicitly represents both its view and spatial location. We measure its temporal variation with respect to the temporal mean:

$$\bar{\mathbf{X}}^p = \frac{1}{T}\sum_{t=1}^{T}\mathbf{X}_t^p, \qquad \Delta^p = \max_t \left\|\mathbf{X}_t^p - \bar{\mathbf{X}}^p\right\|_1. \tag{2}$$

A patch is classified as static if $\Delta^p \leq \tau$ (with a default threshold of $\tau = 0.02$) and dynamic otherwise, where τ controls the sensitivity to temporal changes. We visualize the resulting static and dynamic regions in Fig. 3. After classification, we denote the resulting static and dynamic tokens at timestamp t as $\mathbf{X}_{t,s}$ and $\mathbf{X}_{t,d}$, respectively.

3.3 Temporal-aware Token Merging

Merging. Given the static and dynamic tokens obtained from patch classification, we treat them differently according to their temporal characteristics. Static regions remain largely consistent across timestamps, resulting in highly redundant token representations. We therefore average the static tokens at corresponding spatial locations across timestamps and share the resulting tokens across the entire sequence. In contrast, dynamic regions contain timestamp-specific scene changes and are therefore preserved independently at every timestamp.

Following the notation in Sec. 3.2, we denote the static and dynamic tokens at timestamp t as $\mathbf{X}_{t,s}$ and $\mathbf{X}_{t,d}$, respectively. The temporally shared static tokens are constructed as

$$\mathbf{X}_s = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{X}_{t,s}. \tag{3}$$

Thus, the static tokens are retained only once for shared processing, whereas the dynamic tokens are retained separately for all timestamps. If S and D denote the numbers of static and dynamic tokens per timestamp, respectively, the total number of tokens processed by Global Token Interaction is reduced from $T(S+D)$ to $S+TD$.

Static-shared and dynamic-specific processing. The static and dynamic tokens are subsequently processed according to their temporal characteristics. We denote a generic token processing block as $\mathcal{B}(\cdot)$, which can instantiate either a conventional attention block or a TTT-based fast-weight block.

Instead of repeatedly processing temporally redundant content, the shared static tokens are processed only once:

$$\widetilde{\mathbf{X}}_s = \mathcal{B}(\mathbf{X}_s). \tag{4}$$

The resulting representation is then shared across all timestamps:

$$\widetilde{\mathbf{X}}_s \xrightarrow{\text{Copy}} \widetilde{\mathbf{X}}_{t,s}, \qquad t = 0, \ldots, T-1. \tag{5}$$

In contrast, the dynamic tokens retain their timestamp-specific representations and are processed independently:

$$\widetilde{\mathbf{X}}_{t,d} = \mathcal{B}(\mathbf{X}_{t,d}), \qquad t = 0, \ldots, T-1. \tag{6}$$

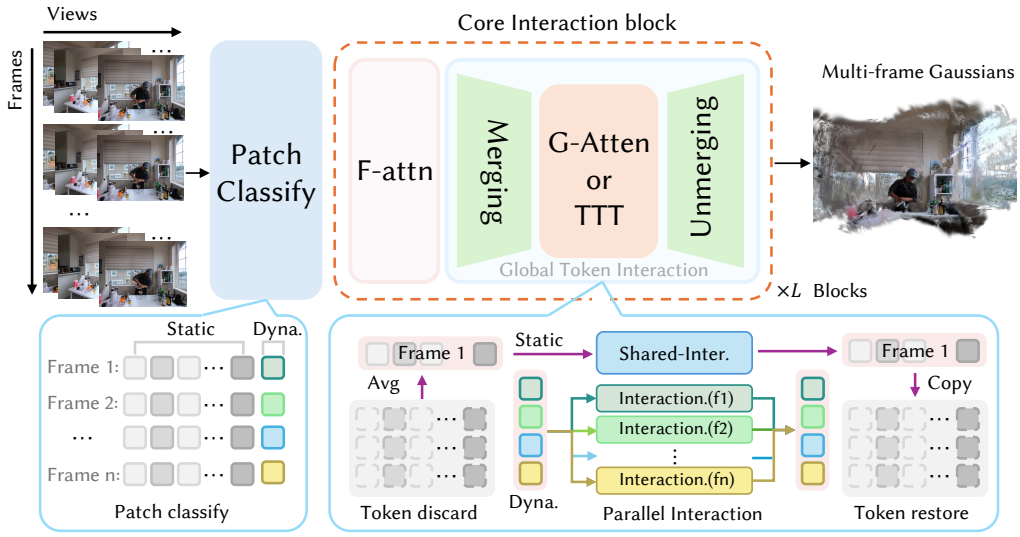


Figure 4: **Overview of T-LRM.** T-LRM supports multi-frame multi-view inputs. After tokenization, we (1) classify patches into static and dynamic regions, (2) merge temporally redundant static tokens after frame attention while retaining timestamp-specific dynamic tokens, (3) perform shared interaction for static tokens and parallel timestamp-wise interaction for dynamic tokens, using either Global Attention or TTT Update, and (4) unmerge the tokens to recover the original layout for dense reconstruction.

Accordingly, no global interaction is performed between the static and dynamic token groups, or between dynamic token groups from different timestamps. This decoupled processing removes redundant and cross-temporal token interactions. Although **removing static–dynamic global interaction** may introduce a small reduction in reconstruction quality, our experiments show that the degradation remains limited while Global Token Interaction is accelerated by approximately $3\times$ (Tab.2).

Here, $\mathcal{B}$ denotes the Global Token Interaction module, which can be instantiated by either attention-based or TTT-based architectures. Detailed implementations for different architectures are provided in the supplementary material(Appendix Sec.B).

Unmerging. After processing, the shared static representation is replicated across timestamps, while the independently processed dynamic representations remain at their corresponding timestamps. Specifically, the reconstructed token sequence at timestamp t is formed by

$$\widetilde{\mathbf{X}}_t = \mathrm{Unmerge}\left(\widetilde{\mathbf{X}}_{t,s}, \widetilde{\mathbf{X}}_{t,d}\right), \qquad t = 0, \ldots, T-1, \tag{7}$$

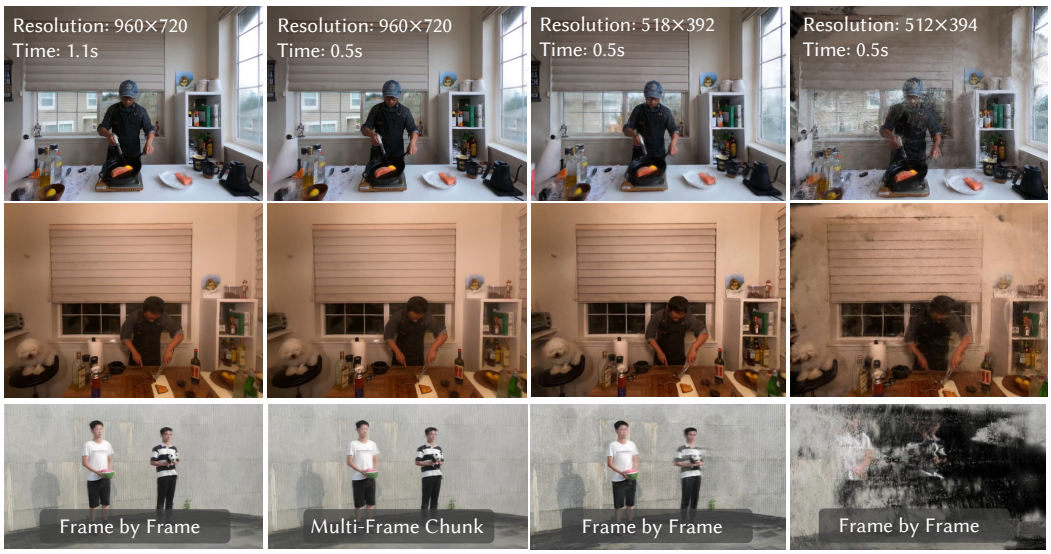
where $\mathrm{Unmerge}(\cdot)$ restores the static and dynamic tokens to their original spatial locations.

Unlike conventional token unmerging methods that recover merged tokens by copying a shared representation to all constituent tokens, our operation only broadcasts temporally shared static representations while preserving timestamp-specific dynamic representations. The original token topology is therefore exactly recovered before the residual connection and Gaussian decoder, requiring no modification to the downstream architecture.

4 Experiments

4.1 Experimental Setup

We assess T-LRMon several standard multi-view dynamic benchmarks, including the classic N3DV Li et al. (2022b) dataset, the weakly textured MeetRoom Li et al. (2022a) dataset, ZJU-SelfCap Xu et al. (2023b) and large-motion settings such as ENeRFLin et al. (2022). Unless noted otherwise, each experiment takes six synchronized views at 960×720 resolution as input. The study is organized around three questions: (1) How Much Acceleration Does the Proposed Framework Achieve? (2) Does Temporal-aware Token Merging Preserve Reconstruction Quality? (3) Can the Proposed Framework Generalize to Different Reconstruction Mode? All runs are performed on a workstation with a single NVIDIA A100 GPU (80GB VRAM).



(a) TTTLRM (b) Ours (c) WorldMirror (d) DepthSplat

Figure 5: **Qualitative comparison.** Our method is built upon temporal chunk-wise input, whereas others follow their default evaluation settings. Our approach achieves inference acceleration with no noticeable degradation in visual quality.

Table 1: **Quantitative comparison on different datasets.** We report PSNR (↑), SSIM (↑), LPIPS (↓), tOF (↓) Time (↓), and Speedup (↑). Six uniformly spaced camera views are used as input for all methods, and all metrics are averaged over all frames.

Dataset	TTTLRM			Ours			WorldMirror			DepthSplat		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM ↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Infer type	Frame by Frame			20 Frames chunk			Frame by Frame			Frame by Frame		
ZJU-SelfCap	20.72	0.715	0.270	20.56	0.710	0.266	19.17	0.603	0.300	17.07	0.562	0.323
MeetRoom	30.14	0.932	0.187	29.94	0.928	0.193	26.91	0.906	0.137	20.04	0.719	0.345
ENeRF	23.99	0.528	0.325	23.73	0.523	0.336	25.11	0.660	0.272	10.79	0.162	0.637
N3DV	29.23	0.896	0.157	29.11	0.894	0.160	27.81	0.873	0.153	22.46	0.730	0.323
Mean	26.02	0.768	0.235	25.84	0.764	0.239	24.75	0.760	0.216	17.59	0.543	0.407
Avg tOF↓	0.282			0.190			0.292			1.471		
Resolution↑	960×720			960×720			518×392			512×394		
Time(s) ↓	1.1			0.5			0.5			0.5		

4.2 Main Experiment

Q1: How Much Acceleration Does the Proposed Framework Achieve?

We evaluate the computational efficiency of our framework by comparing inference time and speedup across different sequence lengths. By sharing static tokens across timestamps while preserving timestamp-specific dynamic tokens, Temporal-aware Token Merging reduces redundant Global Token Interaction.

As shown in Tab. 1 and Fig. 5, all methods use 6 uniformly sampled views. WorldMirror Liu et al. (2025) and DepthSplat are evaluated at their default resolutions, while TTT-LRM and ours use 720×960. The baselines operate frame by frame, whereas ours processes 20-frame chunks to exploit temporal redundancy. Our method maintains reconstruction quality close to TTT-LRM while substantially improving efficiency. As further analyzed in Tab. 2, Temporal-aware Token Merging accelerates Global Token Interaction by approximately $3\times$. Controlled experiments show that frame parallelism provides additional speedup, while the major gain comes from reducing redundant Global Token Interaction.

Q2: Does Temporal-aware Token Merging Preserve Reconstruction Quality?

As shown in Tab. 1 and Fig. 5, we compare our method with the original reconstruction models on multiple benchmark datasets. Quantitative results are reported using PSNR, SSIM, and LPIPS,

Table 2: **Controlled analysis of chunk-wise parallelism and Our Temporal-aware Token Merging. Global Token Interaction** is the primary component accelerated by our strategy (Conduct on N3DV).

Module / Metric	TTT-LRM	TTT-LRM (Naive frame parallel)			Ours (Chunk-wise)					
Chunk size T	1	5	10	15	5	10	15	20	25	30
Frame-Attention (ms)	403	309	303	306	305	299	303	309	316	311
Global Token Interaction (ms)	680	662	660	665	277	232	211	197	188	189
GTI Speedup ↑	1.00×	1.03×	1.03×	1.02×	2.45×	2.93×	3.22×	3.45×	**3.62**×	3.60×
Overall (ms/frame) ↓	1083	971	972	966	582	531	514	506	504	500
Overall Speedup ↑	1.00×	1.12×	1.11×	1.12×	1.86×	2.03×	2.11×	2.14×	2.15×	**2.17**×
PSNR ↑	29.23	29.23	29.23	29.23	29.16	29.13	29.11	29.12	29.09	29.06
Peak Memory (GB) ↓	9	36	57	76	13	17	25	35	46	57

while qualitative comparisons demonstrate reconstruction fidelity under challenging motions and long video sequences. Despite significantly reducing computational cost, our method maintains comparable reconstruction quality while improving temporal consistency.

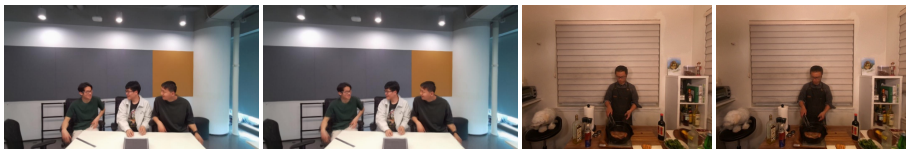
Temporal consistency. Beyond frame-wise reconstruction quality, we further evaluate temporal consistency using the temporal optical-flow error (tOF), following TecoGAN Chu et al. (2020). Let $\hat{I}_t$ and I_t denote the rendered and ground-truth images at timestamp t from the same evaluation viewpoint, respectively. Given an optical flow estimator $\mathcal{F}$, tOF measures the discrepancy between the estimated inter-frame motion of the reconstructed and ground-truth sequences:

$$\text{tOF} = \frac{1}{|\mathcal{P}||\Omega|} \sum_{t \in \mathcal{P}} \sum_{\mathbf{x} \in \Omega} \left\| \mathcal{F}(\hat{I}_{t-1}, \hat{I}_t)(\mathbf{x}) - \mathcal{F}(I_{t-1}, I_t)(\mathbf{x}) \right\|_2, \tag{8}$$

where $\mathcal{P}$ denotes the set of time indices corresponding to the evaluated consecutive-frame pairs $(t-1, t)$, and Ω denotes the image domain. A lower tOF indicates closer agreement between the estimated inter-frame motion of the reconstructed and ground-truth sequences. As shown in Tab. 1, our method reduces tOF from 0.282 for frame-wise TTT-LRM to 0.190.

Table 3: Effect of the proposed Temporal-aware Token Merging on WorldMirror and TTT-LRM, evaluated on the N3DV.

Method	PSNR↑	SSIM↑	LPIPS↓	Time(s)↓	Speedup↑
WorldMirror	27.81	0.873	0.153	0.51	1.00×
WorldMirror + Ours	27.69	0.871	0.154	0.23	2.22×
TTT-LRM	29.23	0.896	0.157	1.12	1.00×
TTT-LRM + Ours	29.10	0.894	0.160	0.54	2.07×



(a) TTTLRM (b) TTTLRM+Ours (c) WorldMirror (d) WorldMirror+Ours

Figure 6: **Model sweep.** Visualization of applying our method to different feed-forward reconstruction models, demonstrating its generality across existing approaches.

Q3: Can the Proposed Framework Generalize to Different Reconstruction Models?

To evaluate the applicability of the proposed framework across different reconstruction architectures, as shown in Tab. 3 and Fig. 6, we integrate our Temporal-aware Token Merging into two recent feed-forward reconstruction models without retraining or modifying their network architectures. Specifically, we evaluate **WorldMirror**, which employs Global Attention for global token interaction, and **TTT-LRM**, which performs global token interaction through TTT Update during inference. These two models cover two distinct designs for global token interaction, and the results show that our framework can be effectively applied to both attention-based and TTT-based reconstruction models.

4.3 Ablation Studies

Effect of the input chunk length. In Tab. 2, we study the effect of the input chunk length T. As T increases, the shared static tokens are reused across more timestamps, progressively reducing the amortized cost of Global Token Interaction. The GTI speedup increases from $2.45\times$ at $T = 5$ to up to $3.62\times$, while the overall speedup reaches $2.17\times$ and PSNR remains close to the frame-wise baseline. The efficiency gain gradually saturates for longer chunks, as the remaining computation is dominated by timestamp-specific dynamic tokens and Frame-Attention.

Source of acceleration. We further disentangle the acceleration introduced by frame parallelism from that of our Temporal-aware Token Merging in Tab. 2. Naive Frame Parallelism reduces Frame-Attention time through parallel execution, but leaves Global Token Interaction nearly unchanged, resulting in only a modest overall speedup. In contrast, our method substantially accelerates Global Token Interaction and achieves up to $2.17\times$ overall speedup. These results demonstrate that frame parallelism provides additional efficiency gains, while the major acceleration comes from reducing redundant Global Token Interaction.

Effect of the static ratio. In Tab. 4, we investigate the influence of the static ratio by varying the motion classification threshold τ. A smaller threshold classifies more patches as dynamic, reducing the amount of temporal token merging and gradually degenerating to the original reconstruction model. Conversely, a larger threshold increases the proportion of shared static tokens, resulting in higher computational efficiency while potentially introducing slight quality degradation near motion boundaries. This experiment reveals the trade-off between reconstruction quality and inference efficiency.

Table 4: Ablation study on the static classification threshold τ, conducted on the *coffee_martini* scene from N3DV.

τ	Static Ratio	PSNR↑	Time (ms/frame)↓
0.20	0.995	25.30	418.58
0.10	0.963	25.97	480.54
0.02	0.896	26.88	499.89
0.01	0.857	26.59	509.82
0.008	0.783	26.46	544.52
0.006	0.741	26.44	563.34

5 Discussion and Limitations

Our acceleration relies on temporal redundancy through **reusing tokens in static regions**. As analyzed in Eq. 1, the reduction in Global Token Interaction increases with the proportion of static regions, i.e., $S/(S + D)$. Consequently, the acceleration becomes less pronounced in scenes dominated by large-scale motion, where fewer tokens can be safely reused across time. In such cases, more tokens require timestamp-specific Global Token Interaction to faithfully capture dynamic geometry and appearance. Developing more adaptive token reuse strategies for highly dynamic scenes is an interesting direction for future work.

Following prior dynamic reconstruction methods Xu et al. (2024); Wang et al. (2023); Xu et al. (2023a), our framework assumes synchronized and fixed camera rigs. This setting provides spatial correspondence across timestamps, enabling efficient static/dynamic token classification. Camera motion or calibration drift may break such correspondence and require additional alignment. Moreover, illumination changes, shadows, and specular effects may be classified as dynamic, reducing the amount of temporal reuse.

6 Conclusion

We presented T-LRM, a temporal-aware framework that extends feed-forward 3D reconstruction models to efficient multi-view dynamic reconstruction. Our Temporal-aware Token Merging exploits temporal redundancy by sharing temporally consistent static tokens across timestamps while preserving timestamp-specific dynamic tokens, substantially reducing redundant computation. As a lightweight, training-free, and plug-and-play framework, T-LRM can be integrated into different feed-forward reconstruction architectures without retraining or architectural modifications. Extensive experiments demonstrate multi-fold inference acceleration while maintaining competitive reconstruction quality and temporal consistency. We hope this work provides a simple yet effective paradigm for efficient dynamic reconstruction with future 3D models.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation. We did not use generative AI tools for any other tasks requiring disclosure, including developing the research methodology, designing experiments, implementing the proposed method, or interpreting results; the remaining required disclosure tasks are not applicable to this work.

Additionally, we used generative AI tools to assist with drafting and editing parts of the manuscript, improving its readability, and writing a small number of utility functions. The core method and implementation were developed by the authors. All AI-assisted text and code were reviewed and verified by the authors. We take full responsibility for the final content of this work, including text and code produced with the aid of generative AI.

REFERENCES

Matteo Balice, Yanik Kunzi, Chenyangguang Zhang, Matteo Matteucci, Marc Pollefeys, and Sunghwan Hong. No pose, no problem in 4d: Feed-forward dynamic gaussians from unposed multi-view videos, 2026. URL https://arxiv.org/abs/2605.22190.

Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion, 2023. URL https://arxiv.org/abs/2303.17604.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *ICLR*, 2023.

Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 130–141, 2023.

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19457–19467, 2024.

Xiaoxue Chen, Ziyi Xiong, Yuantao Chen, Gen Li, Nan Wang, Hongcheng Luo, Long Chen, Haiyang Sun, BING WANG, Guang Chen, et al. Feedforward 4d reconstruction for dynamic driving scenes using unposed images. 2025a.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Ttt3r: 3d reconstruction as test-time training. *arXiv preprint arXiv:2509.26645*, 2025b.

Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European Conference on Computer Vision*, pp. 370–386. Springer, 2024.

Mengyu Chu, You Xie, Jonas Mayer, Laura Leal-Taixé, and Nils Thuerey. Learning temporal coherence via self-supervision for gan-based video generation. *ACM Transactions on Graphics (TOG)*, 39(4):75–1, 2020.

Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12479–12488, 2023.

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction, 2025. URL https://arxiv.org/abs/2511.19971.

Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. *arXiv preprint arXiv:2312.14937*, 2023.

Junhwa Hur, Charles Herrmann, Songyou Peng, Philipp Henzler, Zeyu Ma, Todd Zickler, and Deqing Sun. Ufo-4d: Unposed feedforward 4d reconstruction from two images. *arXiv preprint arXiv:2602.24290*, 2026.

Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, et al. Rayzer: A self-supervised large view synthesis model. *arXiv preprint arXiv:2505.00702*, 2025a.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM Transactions on Graphics*, 44(6):1–16, 2025b.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. *arXiv preprint arXiv:2410.17242*, 2024.

Haian Jin, Rundi Wu, Tianyuan Zhang, Ruiqi Gao, Jonathan T. Barron, Noah Snavely, and Aleksander Holynski. Zipmap: Linear-time stateful 3d reconstruction via test-time training, 2026. URL https://arxiv.org/abs/2603.04385.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139–1, 2023.

Yushi Lan, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Shuai Yang, Bo Dai, Chen Change Loy, and Xingang Pan. Stream3r: Scalable sequential 3d reconstruction with causal transformer. *arXiv preprint arXiv:2508.10893*, 2025.

Lingzhi Li, Zhen Shen, Zhongshu Wang, Li Shen, and Ping Tan. Streaming radiance fields for 3d video synthesis. *Advances in Neural Information Processing Systems*, 35:13485–13498, 2022a.

Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5521–5531, 2022b.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8508–8520, 2024.

Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia Conference Proceedings*, 2022.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Yifan Liu, Zhiyuan Min, Zhenwei Wang, Junta Wu, Tengfei Wang, Yixuan Yuan, Yawei Luo, and Chunchao Guo. Worldmirror: Universal 3d world reconstruction with any-prior prompting. *arXiv preprint arXiv:2510.10726*, 2025.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Jiawei Ren, Kevin Xie, Ashkan Mirzaei, Hanxue Liang, Xiaohui Zeng, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung W Kim, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024.

Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024.

Chen Wang, Hao Tan, Wang Yifan, Zhiqin Chen, Yuheng Liu, Kalyan Sunkavalli, Sai Bi, Lingjie Liu, and Yiwei Hu. tttlrm: Test-time training for long context and autoregressive 3d reconstruction. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2026.

Feng Wang, Sinan Tan, Xinghang Li, Zeyue Tian, Yafei Song, and Huaping Liu. Mixed neural voxels for fast multi-view video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19706–19716, 2023.

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

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. 2024a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proc. of CVPR*, pp. 10510–10522, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proc. of CVPR*, pp. 20697–20709, 2024b.

Yifan Wang, Peishan Yang, Zhen Xu, Jiaming Sun, Zhanhua Zhang, Yong Chen, Hujun Bao, Sida Peng, and Xiaowei Zhou. Freetimegs: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In *CVPR*, 2025c. URL https://zju3dv.github.io/freetimegs.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Scalable permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025d.

Yiming Wang, Lucy Chai, Xuan Luo, Michael Niemeyer, Manuel Lagunas, Stephen Lombardi, Siyu Tang, and Tiancheng Sun. Splatvoxel: History-aware novel view streaming without temporal training, 2025e. URL https://arxiv.org/abs/2503.14698.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20310–20320, 2024.

Jiahao Wu, Rui Peng, Jianbo Jiao, Jiayu Yang, Luyang Tang, Kaiqiang Xiong, Jie Liang, Jinbo Yan, Runling Liu, and Ronggang Wang. Localdygs: Multi-view global dynamic scene modeling via adaptive local implicit feature decoupling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9519–9529, 2025a.

Jiahao Wu, Rui Peng, Zhiyan Wang, Lu Xiao, Luyang Tang, Jinbo Yan, Kaiqiang Xiong, and Ronggang Wang. Swift4d: Adaptive divide-and-conquer gaussian splatting for compact and efficient reconstruction of dynamic scene. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=c1RhJVTPwT.

Yuqi Wu, Wenzhao Zheng, Jie Zhou, and Jiwen Lu. Point3r: Streaming 3d reconstruction with explicit spatial pointer memory, 2025c. URL https://arxiv.org/abs/2507.02863.

Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16453–16463, 2025a.

Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 4k4d: Real-time 4d view synthesis at 4k resolution. 2023a.

Zhen Xu, Tao Xie, Sida Peng, Haotong Lin, Qing Shuai, Zhiyuan Yu, Guangzhao He, Jiaming Sun, Hujun Bao, and Xiaowei Zhou. Easyvolcap: Accelerating neural volumetric video research. 2023b.

Zhen Xu, Yinghao Xu, Zhiyuan Yu, Sida Peng, Jiaming Sun, Hujun Bao, and Xiaowei Zhou. Representing long volumetric video with temporal gaussian hierarchy. *ACM Transactions on Graphics*, 43(6), November 2024. URL https://zju3dv.github.io/longvolcap.

Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4dgt: Learning a 4d gaussian transformer using real-world monocular videos. *Advances in neural information processing systems*, 2025b.

Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21924–21935, 2025.

Zeyu Yang, Hongye Yang, Zijie Pan, Xiatian Zhu, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20331–20341, 2024.

Botao Ye, Boqi Chen, Haofei Xu, Daniel Barath, and Marc Pollefeys. Yonosplat: You only need one model for feedforward 3d gaussian splatting. In *International Conference on Learning Representations (ICLR)*, 2026.

Yuheng Yuan, Qiuhong Shen, Shizun Wang, Xingyi Yang, and Xinchao Wang. Test3r: Learning to reconstruct 3d at test time. *arXiv preprint arXiv:2506.13750*, 5, 2025.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arxiv:2410.03825*, 2024a.

Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2024b.

Qitao Zhao, Hao Tan, Qianqian Wang, Sai Bi, Kai Zhang, Kalyan Sunkavalli, Shubham Tulsiani, and Hanwen Jiang. E-rayzer: Self-supervised 3d reconstruction as spatial visual pre-training. In *CVPR*, 2026.

Kaichen Zhou, Yuhan Wang, Grace Chen, Gaspard Beaudouin, Fangneng Zhan, Paul Pu Liang, and Mengyu Wang. Page-4d: Vggt-4d perception via disentangled pose and geometry estimation. In *The Fourteenth International Conference on Learning Representations*, 2025.

Dong Zhuo, Wenzhao Zheng, Jiahe Guo, Yuqi Wu, Jie Zhou, and Jiwen Lu. Streaming 4d visual geometry transformer, 2026. URL https://arxiv.org/abs/2507.11539.

Chen Ziwen, Hao Tan, Kai Zhang, Sai Bi, Fujun Luan, Yicong Hong, Li Fuxin, and Zexiang Xu. Long-lrm: Long-sequence large reconstruction model for wide-coverage gaussian splats. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

A Appendix

Our goal is to reduce redundant computation in feed-forward multi-view dynamic reconstruction by exploiting temporal consistency. Given a multi-view video chunk containing T timestamps and V synchronized views, the input images are first converted into patch tokens following the original reconstruction backbone. Different from conventional frame-by-frame inference that independently processes each timestamp, our method reorganizes tokens according to their temporal behaviors before global temporal reasoning.

The key observation is that most regions in multi-view dynamic videos remain unchanged over time, while only a small portion corresponds to moving objects. Therefore, we divide input tokens into two categories: **Static tokens**, which represent temporally invariant scene structures and can be shared across timestamps. **Dynamic tokens**, which correspond to changing regions and require timestamp-specific processing. Based on this observation, our framework consists of three steps:

(1) Temporal-aware token classification: identify static and dynamic patches based on temporal appearance variation.

(2) Token merging and decoupled adaptation: reuse static tokens across timestamps while preserving independent dynamic tokens.

(3) Token unmerging: restore the original token layout required by the reconstruction decoder.

The overall process converts conventional time-proportional computation into change-proportional computation, where only changing regions require repeated temporal adaptation.

B Implementation on Different Reconstruction Architectures

This section describes how Temporal-aware Token Merging is integrated into conventional attention-based and TTT-based reconstruction architectures. In both cases, our method is inserted into the Global Token Interaction stage without introducing additional trainable modules or modifying the downstream decoder.

Frame-Attention. Frame-Attention is applied before Temporal-aware Token Merging and follows the original implementation of each reconstruction backbone. It processes the multi-view tokens within each timestamp to preserve complete spatial and cross-view context. We do not reduce or merge tokens during this stage. After Frame-Attention, the resulting tokens are classified into temporally static and dynamic groups and are passed to the modified Global Token Interaction stage described below.

Temporal Token Organization. Let $\mathbf{X}_{t,s}$ and $\mathbf{X}_{t,d}$ denote the static and dynamic tokens after Frame-Attention at timestamp t, respectively. We average the static tokens at corresponding spatial locations across the temporal chunk:

$$\mathbf{X}_s = \frac{1}{T}\sum_{t=0}^{T-1}\mathbf{X}_{t,s}. \tag{9}$$

The resulting $\mathbf{X}_s$ is used as the shared static token group, while each $\mathbf{X}_{t,d}$ is retained as a timestamp-specific dynamic token group. If each timestamp contains S static tokens and D dynamic tokens, this organization reduces the total number of tokens processed by Global Token Interaction from $T(S+D)$ to $S+TD$.

Conventional Global Attention. For attention-based reconstruction models, the original Global Attention jointly processes all input tokens. In our implementation, the shared static tokens and timestamp-specific dynamic tokens are processed as independent groups.

The shared static representation is computed once:

$$\widetilde{\mathbf{X}}_s = \mathrm{Attn}\left(\mathbf{X}_s, \mathbf{X}_s, \mathbf{X}_s\right). \tag{10}$$

For each timestamp, Global Attention is applied independently to its dynamic tokens:

$$\widetilde{\mathbf{X}}_{t,d} = \mathrm{Attn}\left(\mathbf{X}_{t,d}, \mathbf{X}_{t,d}, \mathbf{X}_{t,d}\right), \qquad t = 0, \ldots, T-1. \tag{11}$$

Therefore, no Global Attention is performed between the shared static group and the dynamic groups, or between dynamic groups from different timestamps. This separation prevents timestamp-specific dynamic tokens from directly interacting across time and avoids repeatedly processing temporally redundant static content.

After Global Attention, $\widetilde{\mathbf{X}}_s$ is copied to all timestamps and combined with the corresponding dynamic representations:

$$\widetilde{\mathbf{X}}_t = \mathrm{Unmerge}\left(\widetilde{\mathbf{X}}_s, \widetilde{\mathbf{X}}_{t,d}\right). \tag{12}$$

The unmerged sequence restores the original spatial token layout before the residual connection and downstream decoder.

For the attention operation itself, conventional frame-wise processing has a quadratic interaction cost proportional to

$$T(S+D)^2, \tag{13}$$

whereas the separated processing used by our method has a cost proportional to

$$S^2 + TD^2. \tag{14}$$

The exact runtime reduction also depends on feature projections, memory operations, and the underlying attention implementation.

TTT Fast-Weight Interaction. For TTT-based reconstruction models, Global Token Interaction is implemented through fast-weight adaptation rather than explicit pairwise attention. We apply the same static-shared and dynamic-specific processing principle to the TTT block.

For a token group $\mathbf{X}$, the TTT interaction consists of an UPDATE operation followed by an APPLY operation:

$$\mathcal{B}_{\mathrm{TTT}}(\mathbf{X}) = \textsc{Apply}\left(\textsc{Update}(\mathbf{X}), \mathbf{X}\right). \tag{15}$$

The shared static tokens pass through this interaction only once:

$$\widetilde{\mathbf{X}}_s = \mathcal{B}_{\mathrm{TTT}}\left(\mathbf{X}_s\right). \tag{16}$$

The dynamic tokens at each timestamp are processed independently:

$$\widetilde{\mathbf{X}}_{t,d} = \mathcal{B}_{\mathrm{TTT}}\left(\mathbf{X}_{t,d}\right), \qquad t = 0, \ldots, T-1. \tag{17}$$

Accordingly, the fast-weight UPDATE and APPLY operations are independently performed for the shared static group and for each timestamp-specific dynamic group. No fast-weight state is propagated between the static and dynamic groups or across different timestamps.

The processed static representation is then copied across timestamps and unmerged with the corresponding dynamic representations:

$$\widetilde{\mathbf{X}}_t = \mathrm{Unmerge}\left(\widetilde{\mathbf{X}}_s, \widetilde{\mathbf{X}}_{t,d}\right). \tag{18}$$

This operation restores the original token topology before the residual connection and Gaussian decoder.

Efficiency–Quality Trade-off. Separating the token groups removes Global Token Interaction between static and dynamic regions. This may lead to a small reduction in reconstruction quality because some global contextual exchange is omitted. However, Frame-Attention is retained without modification before token merging, allowing each token to preserve within-timestamp multi-view context. Empirically, the resulting quality degradation remains limited, while the cost of Global Token Interaction is reduced by approximately $3\times$.

Summary. The attention-based and TTT-based implementations use different global interaction mechanisms, but follow the same processing principle. Temporally redundant static tokens are averaged and processed once, while dynamic tokens are preserved and processed independently for each timestamp. The resulting representations are subsequently unmerged to restore the original token layout. This design allocates repeated Global Token Interaction primarily to scene changes and enables efficient integration into both reconstruction architectures without additional training.

C Additional Experimental Details

C.1 Dataset Coverage and Evaluation Protocol

We evaluate our method on N3DV, MeetRoom, ZJU-SelfCap, and ENeRF. For each dataset, we evaluate all frames from all scenes, without selecting a subset of scenes or temporally subsampling the evaluation sequences. Camera view 0 is consistently used as the held-out evaluation viewpoint across all scenes. The reconstructed scene at each timestamp is rendered from this viewpoint and compared with the corresponding ground-truth image.

Six input camera views are sampled uniformly from the available input viewpoints, and synchronized images from these views are used to reconstruct each timestamp. The same input-view selection is used for the frame-wise TTT-LRM baseline and our method, ensuring that their comparison uses the same image observations.

C.2 Image Preprocessing

For TTT-LRM and our method, input images are cropped and resized to a width of 960 pixels and a height of 720 pixels. Both methods use the same preprocessing procedure, so that their quality and efficiency comparison is not affected by differences in input resolution or cropping.

C.3 Shared Parameters Across Scenes

Unless explicitly varied in an ablation study, we use a single parameter configuration across all scenes and datasets. In particular, the static/dynamic classification threshold τ is shared across scenes and datasets, rather than adjusted separately for each test sequence. The main comparison uses temporal chunks of 20 frames, while the chunk-length ablation varies this setting as reported in the main paper.

This shared configuration evaluates whether the proposed temporal token reuse strategy remains effective across different scenes without scene-specific parameter adjustment. The static-token ratio is determined by the input sequence and the shared threshold, and may therefore vary across scenes even when the parameter configuration is unchanged.

C.4 Backbone Comparisons

Our primary controlled comparison is between frame-wise TTT-LRM and TTT-LRM equipped with our temporal-aware token merging strategy. Both use the same six input views and the same 960×720 input resolution. This comparison evaluates the effect of temporal token reuse on reconstruction quality and inference efficiency within the same backbone.

The cross-model comparison additionally includes WorldMirror and DepthSplat at their respective default input resolutions, as reported in the main paper. These results characterize the performance of the evaluated model configurations; they do not isolate the effect of the backbone architecture under a common input resolution. Differences in preprocessing, pretrained models, and the use of camera information should therefore be considered when interpreting the cross-model results.

We additionally evaluate WorldMirror with and without our method on N3DV to examine applicability beyond the TTT-based backbone. Together, the TTT-LRM and WorldMirror experiments provide evidence of applicability to both TTT-based and attention-based global interaction mechanisms.

C.5 Static-Token Ratio Ablation

The static-token ratio ablation varies the classification threshold τ while examining the resulting reconstruction quality and processing time. The reported static-token ratio is an outcome of patch classification, rather than a directly specified token-retention budget. Reporting both τ and the resulting static-token ratio is therefore necessary to reproduce this experiment.

At a static-token ratio of 0.896, the method achieves 26.88 dB PSNR with a processing time of 499.89 ms per frame. Increasing the static-token ratio to 0.995 reduces the processing time to 418.58 ms per frame, while PSNR decreases to 25.30 dB. These results illustrate the quality cost of aggressive temporal token sharing.

C.6 Metric Reporting

We report PSNR, SSIM, and LPIPS for reconstruction quality and tOF for temporal motion consistency. Reconstruction metrics are evaluated on the rendered images from camera view 0 against the corresponding ground-truth images. The tOF metric compares optical flow estimated from consecutive reconstructed frames with optical flow estimated from the corresponding ground-truth frames.