Evaluating Attention Reuse in Dynamic 3D Gaussian Reconstruction

M. Uzzwal Vanshik Reddy[1], T. N. Rickesh[2], and Sowmya Kamath S.[1]

[1] Department of Information Technology,
National Institute of Technology Karnataka, Surathkal (NITK),
Srinivasnagar P.O., Mangaluru 575025, India
{uzzwalvansh.221it043, sowmyakamath}@nitk.edu.in
[2] Nanyang Technological University,
50 Nanyang Ave, Singapore 639798
rickesh001@e.ntu.edu.sg

Abstract. Vision Transformers play a central role in recent feed-forward approaches to dynamic 3D scene reconstruction based on 3D Gaussian Splatting. While self-attention is theoretically expensive, modern implementations employ highly optimized fused kernels that reduce its practical cost. This raises an open question: can attention reuse mechanisms, originally proposed to reduce quadratic attention complexity, provide meaningful benefits in large spatiotemporal reconstruction models? We investigate this question in the context of STORM, a feed-forward framework for dynamic 3D Gaussian reconstruction. We integrate Less-Attention layers into the STORM-B/8 Transformer backbone and evaluate across different reuse ratios. Our study examines runtime performance, memory usage, parameter growth and reconstruction quality under both naïve and optimized implementations. Our results demonstrate that attention reuse does not provide training time improvements over optimized full attention, even when implemented with careful engineering optimizations, which can be attributed to the additional memory access overhead introduced by reuse mechanisms and the high efficiency of fused attention kernels on modern GPU architectures. Consistent improvements in reconstruction quality at moderate reuse ratios were observed, suggesting that attention reuse introduces a beneficial inductive bias that stabilizes spatiotemporal feature aggregation and primarily influences representational behavior rather than computational efficiency.

Keywords: Dynamic 3D Reconstruction · 3D Gaussian Splatting · Vision Transformers · Attention Reuse · Spatiotemporal Modeling · Hardware-Aware Evaluation · Efficient Attention

1 Introduction

Reconstructing dynamic 3D scenes from sparse multi-view imagery remains a challenge in the computer vision domain, with applications spanning autonomous driving, robotics, augmented reality and simulation. Recent advances in feed-forward reconstruction have demonstrated that high-quality dynamic scenes can

be modeled without per-scene optimization by directly predicting spatiotemporal 3D Gaussian representations from image sequences [1]. These approaches offer significant advantages in scalability and inference speed, enabling real-time or near-real-time reconstruction in complex dynamic environments.

A key enabler of such feed-forward models is the Vision Transformer, aggregating information across spatial locations, camera views and timesteps. In particular, Transformer-based architectures allow dense global interactions that are critical for multi-view geometric reasoning and temporal consistency. However, self-attention scales quadratically with sequence length, a recognized bottleneck in large-scale vision models [2, **?**]. Motivated by this concern, a large body of prior work has explored alternatives to standard self-attention, including low-rank approximations [18], sparse attention patterns [2] and kernel-based methods [6] More recently, attention reuse mechanisms have been proposed [8]based on the observation that attention maps tend to stabilize across Transformer depth. Instead of recomputing query-key interactions at every layer, these methods reuse and transform attention maps computed in earlier layers, aiming to reduce redundant computation while preserving global token interactions. From an asymptotic complexity perspective, such approaches appear attractive, particularly for deep Transformers operating on long sequences. At the same time, the practical cost model of self-attention has fundamentally changed. Modern deep learning frameworks increasingly rely on highly optimized fused attention kernels, such as FlashAttention-style implementations [4].These implementations are explicitly designed to minimize memory traffic and maximize hardware utilization, often rendering theoretical FLOP-based complexity analyses insufficient to predict real-world performance. As a result, it is no longer clear whether attention reuse or approximation mechanisms provide tangible benefits when deployed within large, optimized spatiotemporal vision systems.

In this work, we investigate this question in the context of STORM [19], a feed-forward framework for dynamic 3D Gaussian reconstruction that employs a deep Vision Transformer backbone to jointly reason over space, time and multiple views. STORM represents a challenging and practically relevant setting: its Transformer processes thousands of spatiotemporal tokens per forward pass, With so many tokens being processed it is a good model for testing whether attention reuse can meaningfully improve computational efficiency or reconstruction performance in modern reconstruction pipelines. We integrate Less-Attention layers into the STORM Transformer backbone and conduct a systematic evaluation across multiple attention reuse ratios. We examine both naïve implementations and carefully optimized variants designed to eliminate avoidable overheads, allowing us to disentangle algorithmic limitations from implementation artifacts. Our evaluation considers runtime, memory consumption, parameter scaling and reconstruction quality measured using standard photometric metrics. The main contributions of this work as listed below.

– Hardware aware evaluation of attention reuse within a large-scale feed-forward dynamic 3D Gaussian reconstruction framework, using STORM.

– We integrate the Less-Attention mechanism into the STORM-B/8 Transformer backbone and provide a systematic study across multiple reuse ratios.
– We demonstrate that attention reuse does not yield run time(wall-clock) speedups over optimized fused full-attention kernels.

The rest of this article is organized as follows. Section 2 reviews related work in dynamic 3D reconstruction, Transformer-based feed-forward modeling and efficient attention mechanisms. Section 3 presents our methodology, including the formulation of self-attention in STORM, the integration of the Less-Attention mechanism, dataset specifics and implementation optimizations. Section 4 reports experimental results, analyzing memory usage, runtime performance, reconstruction quality and training dynamics. Section 5 discusses the computational and representational implications of attention reuse in spatiotemporal vision systems. Finally, Section 6 concludes the paper and outlines directions for future research.

2 Related Work

Dynamic 3D scene reconstruction has been extensively studied in the context of neural radiance fields (NeRFs) that model time-varying geometry and appearance. Early approaches such as D-NeRF [14], NSFF [10], Nerfies [12] and HyperNeRF [13] introduce deformation fields or canonical representations to account for scene dynamics. While effective, these methods typically require dense temporal sampling, per-scene optimization, or explicit motion supervision, which limits scalability to long sequences or large datasets. The introduction of 3D Gaussian Splatting [7] enabled real-time rendering and significantly faster optimization, motivating several extensions to dynamic settings. Recent works such as Deformable 3D Gaussians [11], 4D Gaussian Splatting [9] and GaussianFlow [5] incorporate motion models or flow-based formulations to handle temporal variation. However, many of these approaches still rely on scene-specific optimization or dense supervision. STORM [19] advances this line of work by introducing a fully feed-forward, self-supervised framework that predicts spatiotemporal 3D Gaussian parameters directly from sparse multi-view video. By eliminating per-scene optimization and explicit motion labels, STORM enables scalable dynamic reconstruction, but relies on a deep Transformer backbone that aggregates information across large spatiotemporal token sequences.

Transformers have become a central component in feed-forward 3D reconstruction due to their ability to model long-range interactions across spatial locations and camera views. Early works such as PixelNeRF [20] and IBRNet [17] employ attention mechanisms to fuse multi-view image features for generalizable radiance field prediction. Subsequent methods extend Transformer-based reasoning to object-centric and scene-level reconstruction tasks. More recently, Transformer architectures have been adopted in Gaussian-based feed-forward reconstruction models. Large Gaussian models such as LGM [16] and GS-LRM [22] demonstrate that Transformers can directly regress complete 3D Gaussian representations

from image sets, enabling fast inference without per-scene optimization. In dynamic settings, Transformers provide a unified mechanism for joint spatial and temporal reasoning, but their reliance on dense self-attention raises concerns about computational efficiency as the number of tokens grows.

A large body of prior work aims to reduce the quadratic complexity of self-attention. Linformer [18] approximates attention using low-rank projections, while Performer [3] employs kernel-based methods to achieve linear complexity. Sparse and block-based attention mechanisms, such as Sparse Transformers [2] and BigBird [21], restrict attention to structured patterns to reduce computation. While these approaches are effective in language modeling and image classification, they often modify the structure or distribution of attention. In contrast, dynamic 3D reconstruction typically requires dense global interactions across views and timesteps, making it challenging to impose sparsity or locality without degrading geometric consistency or reconstruction quality. In parallel, recent work on optimized attention kernels has significantly altered the practical cost model of self-attention. FlashAttention [4] and its successors demonstrate that carefully fused GPU kernels can dramatically reduce memory traffic and improve hardware utilization, making full attention substantially faster in practice than suggested by asymptotic complexity alone.

An alternative approach to improving attention efficiency focuses on reusing attention maps across Transformer layers. The Less-Attention framework [23] proposes to predict attention matrices in deeper layers by applying learnable transformations to attention maps computed earlier in the network, rather than recomputing query–key interactions at every layer. This approach is motivated by empirical observations that attention patterns often stabilize with depth in Vision Transformers. Unlike approximation-based methods, attention reuse preserves dense global interactions between tokens and does not impose sparsity. However, its effectiveness relies on two key assumptions: first, that transforming existing attention maps is computationally cheaper than recomputing attention; and second, that the overhead of manipulating dense attention matrices does not dominate runtime or memory usage. Prior evaluations of attention reuse have primarily focused on image classification benchmarks and moderate sequence lengths, leaving its behavior in large spatiotemporal models with hardware-optimized attention kernels largely unexplored.

To the best of our knowledge, no prior work has systematically evaluated attention reuse within feed-forward dynamic 3D reconstruction frameworks or analyzed its interaction with modern fused attention implementations. Our work addresses this gap by providing a hardware-aware empirical study of attention reuse in STORM. By providing a hardware-aware analysis of both computational and representational effects, we aim to highlight the importance of evaluating efficient attention mechanisms within realistic, optimized system settings and offer insights for the design of future Transformer architectures for large-scale spatiotemporal vision tasks.

3 Methodology

We evaluate the computational and representational effects of attention reuse in large spatiotemporal Transformer architectures for dynamic 3D reconstruction. We base our study on the STORM framework, a feed-forward model for dynamic 3D Gaussian reconstruction and integrate the *Less-Attention mechanism* into its Vision Transformer backbone. Specifically, we replace selected standard self-attention layers with reuse-enabled layers that utilize attention maps from prior layers, while preserving the original tokenization and value projection pathways.

3.1 Dataset Specifics

All experiments were conducted on a subset of the Waymo Open Dataset [15] following the STORM data protocol. We utilize clips, each 2.0s long at 10 FPS and having 3 camera-views for a total of 20 frames in each clip. For experiments with 2 context frames (ctx) the input is the 1st and 10th frames and for experiments with 4 ctx the input is 1st, 5th, 10th and 15th frames. 2 of the remaining frames are used for prediction as target frames (tgt). Input frames are resized to 160×240 and each training sample consists of two context frames and two target frames with up to three camera views. Models are trained for 150,000 iterations and evaluated on a fixed validation set of 720 samples.

3.2 Self-Attention in STORM

STORM employs multi-head self-attention (MHSA) as its primary mechanism for aggregating information across spatial locations, camera viewpoints and temporal frames within a unified token sequence. Tokens are constructed by flattening multi-view features across cameras and timesteps. Within each Transformer layer, MHSA enables every token to attend globally to all others, This dense global interaction plays a central role in producing coherent dynamic 3D Gaussian representations but also leads to quadratic complexity with respect to the spatiotemporal token count, motivating the exploration of more efficient attention mechanisms. Given an input token sequence, $\mathbf{X} \in \mathbb{R}^{N \times D}$, where N denotes the sequence length and D the hidden dimension, each attention head computes queries, keys and values as per Eq. (1), with learnable projection matrices as shown in Eq. (2). The attention output is defined as shown in Eq. (3).

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{X}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{X}\mathbf{W}_V \tag{1}$$

$$\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{D \times d} \tag{2}$$

$$\mathrm{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V} \tag{3}$$

The dominant computational cost arises from the matrix multiplication $\mathbf{QK}^\top$ and the subsequent multiplication with $\mathbf{V}$, resulting in a theoretical complexity of $\mathcal{O}(N^2 d)$ per attention head. In practice, STORM relies on fused scaled dot-product attention kernels that substantially reduce memory traffic and improve hardware utilization.

3.3 Attention Reuse via Less-Attention

Less-Attention reduces redundant attention computation across Transformer layers by explicitly reusing attention maps produced in preceding layers, rather than recomputing full query–key interactions at every depth. Attention patterns in deep Transformers tend to stabilize with depth, motivating reuse across layers. Instead of forming new queries and keys, Less-Attention propagates the attention matrix from layer $l-1$ to layer l through lightweight learnable transformations that adapt the reused attention to the current layer's feature space. The resulting attention weights are then applied to newly computed value projections, preserving dense global token interactions while reducing the computational cost associated with quadratic attention.

Let $\mathbf{A}^{(l-1)} \in \mathbb{R}^{N\times N}$ denote the attention matrix produced at layer $l-1$. Instead of recomputing queries and keys at layer l, the attention matrix is estimated as per Eq. (4), where, Θ_l and Ψ_l are learnable linear transformations applied to the attention matrix. The final attention weights are obtained via softmax normalization as per Eq. (5). The output of the attention layer is then computed as per Eq. (6), where, the value projections $\mathbf{V}^{(l)}$ are computed as in standard self-attention. This formulation preserves dense global token interactions while eliminating explicit query–key recomputation in reused layers.

$$\hat{\mathbf{A}}^{(l)} = \Psi_l\Big(\Theta_l\Big(\mathbf{A}^{(l-1)}\Big)\Big) \tag{4}$$

$$\mathbf{A}^{(l)} = \mathrm{softmax}\Big(\hat{\mathbf{A}}^{(l)}\Big) \tag{5}$$

$$\mathbf{O}^{(l)} = \mathbf{A}^{(l)}\mathbf{V}^{(l)} \tag{6}$$

While Less-Attention removes the explicit $\mathbf{QK}^\top$ operation, it introduces dense transformations over the attention matrix. A full-rank implementation of Θ_l and Ψ_l incurs a complexity of $\mathcal{O}(N^3)$, which is strictly worse than standard self-attention for typical Transformer configurations. To mitigate this, Less-Attention proposes low-rank parameterizations, where $\Theta_l \approx \mathbf{U}_l\mathbf{V}_l^\top, \quad \mathbf{U}_l, \mathbf{V}_l \in \mathbb{R}^{N\times r}, \quad r \ll N$, reducing the complexity to $\mathcal{O}(N^2 r)$. In theory, when $r \ll d$, attention reuse can be computationally cheaper than standard self-attention. However, this analysis does not account for memory access patterns or hardware-level optimizations. Since Less-Attention operates directly on dense $N \times N$ attention matrices, its runtime is highly sensitive to memory bandwidth, particularly for moderate sequence lengths common in spatiotemporal reconstruction.

Next, the Less-Attention mechanism is integrated into the STORM-B/8 Transformer backbone while keeping the total depth fixed at 12 layers. The first k layers use standard full attention and the remaining layers reuse attention maps computed at layer k. We evaluate reuse ratios of 0%, 33%, 50% and 67%, these ratios correspond to 0, 4, 6 and 8 reused layers out of 12, chosen as natural integer breakpoints in the fixed-depth backbone. All other architectural components, including value projections, feed-forward networks, residual connections and decoders, remain unchanged. This design ensures that observed differences can be attributed solely to the attention mechanism.

3.4 Optimized Implementation of Less-Attention

We implement an optimized version of Less-Attention to isolate algorithmic effects from engineering artifacts. Firstly, all transformation parameters associated with attention reuse are pre-allocated at model initialization for a fixed maximum sequence length. During the forward pass, transformation weights are sliced to the active sequence length using tensor views, which incurs negligible overhead. This design avoids dynamic module creation, repeated weight initialization and runtime GPU memory allocation. Next, instead of invoking transformation layers through `nn.Module` calls, we apply attention transformations using direct linear operations. Specifically, we employ optimized matrix multiplication routines to apply the attention transformations, ensuring that computation is dispatched directly to highly optimized GPU kernels without Python-level overhead.

By rewriting the attention transformation as a sequence of matrix multiplications, we avoid intermediate memory copies and reduce kernel launch overhead, improving memory locality and execution efficiency. Finally, we evaluate both full-rank and low-rank parameterizations of the attention transformation matrices. While low-rank variants reduce parameter count and arithmetic complexity, they continue to operate on dense attention matrices and therefore remain sensitive to memory bandwidth constraints. Together, these optimizations provide a best-case estimate of Less-Attention performance within the PyTorch execution model. As a result, observed differences between standard attention and attention reuse can be attributed to fundamental algorithmic and hardware-level trade-offs rather than implementation inefficiencies.

4 Experiments and Results

All our experiments were performed on a single NVIDIA RTX A6000 GPU. The full-attention baselines use fused scaled dot-product kernels throughout, for a fair comparison. All experimented models were trained for 150,000 iterations under identical experimental settings.

4.1 Memory Usage and Parameter Scaling

Table 1 reports peak GPU memory usage and parameter counts. As the reuse ratio increases, baseline parameter counts decrease slightly due to the removal

of query–key projections in reused layers. However, optimized Less-Attention variants introduce a substantial increase in parameter count, growing to over 1.29B parameters at 67% reuse. This growth comes from dense attention transformation matrices that scale quadratically with token count. These results show that attention reuse trades projection parameters increasing memory pressure.

4.2 Runtime Performance

Table 2 summarizes per-iteration runtime and projected total training time for 150k iterations. Optimized implementations reduce runtime by up to 4.01× over naïve baselines. However, despite these improvements, all Less-Attention configurations remain slower than the full-attention baseline in absolute wall-clock time. As can be seen from the tabulated results, the optimized 50% reuse configuration requires 0.94 s per iteration, compared to 0.33 s for the full-attention model. The cost of dense attention matrix transformations outweighs the savings from skipping query–key recomputation.

Table 1: Peak GPU memory usage and parameter count for different attention reuse ratios.

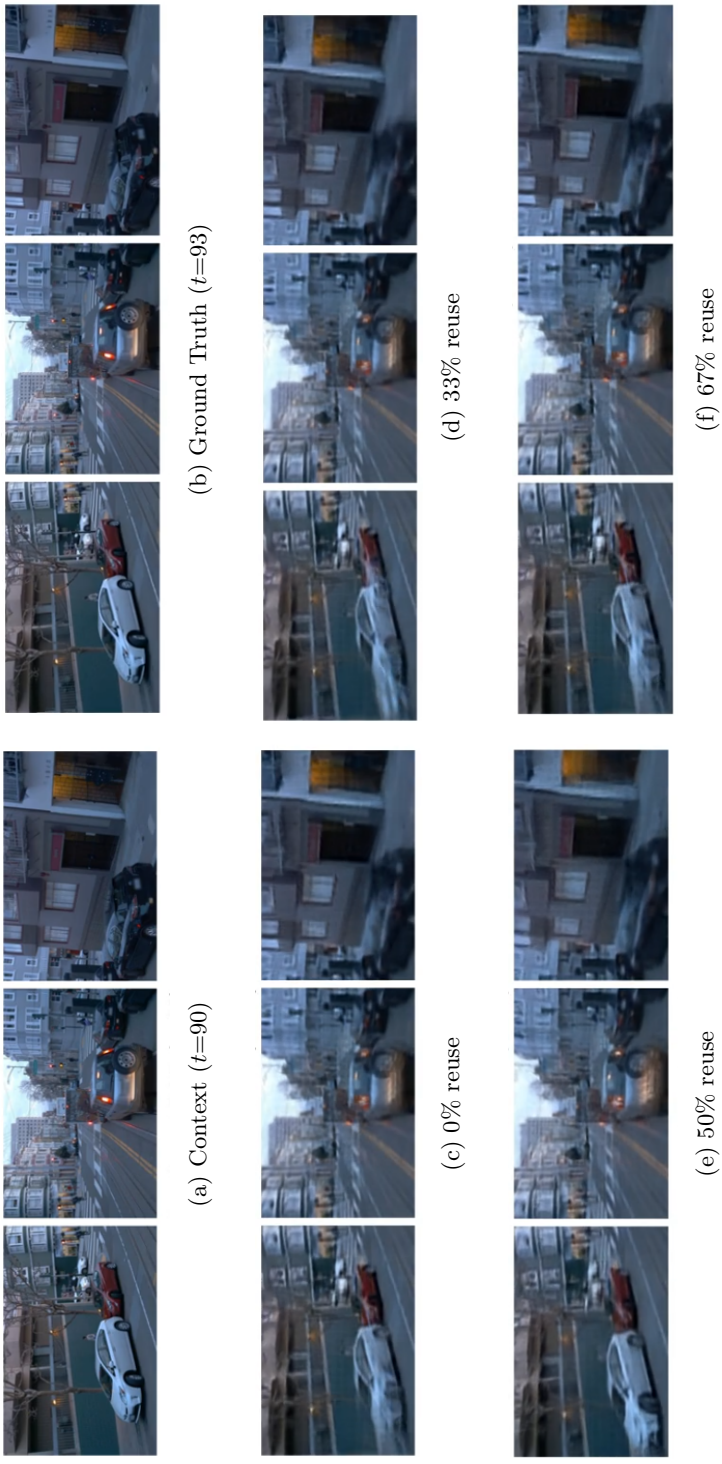
Reuse	**Base Mem**	**Opt Mem**	**Base Params**	**Opt Params**
0%	3.8 GB	19.3 GB	99.66M	99.66M
33%	20.0 GB	28.8 GB	94.94M	631.87M
50%	21.1 GB	34.3 GB	92.58M	897.98M
67%	21.0 GB	41.7 GB	89.03M	1297.13M

Table 2: Runtime comparison across attention reuse ratios. Speed-up is measured relative to the corresponding baseline implementation.

Reuse	**Base** *(s/iter)*	**Opt** *(s/iter)*	**Base Total**	**Opt Total**	**Speed-up**
0%	0.7127	0.3347	29.7 h	14.0 h	2.13×
33%	2.7430	1.3308	114.3 h	55.5 h	2.06×
50%	3.7754	0.9422	157.3 h	39.3 h	4.01×
67%	5.3106	1.3045	182.0 h	54.4 h	3.21×

4.3 Reconstruction Quality

Fig. 1 depicts a qualitative comparison of reconstruction quality across attention reuse ratios. In particular, moderate reuse (50%) produces sharper details and improved dynamic consistency when compared against the ground truth, whereas excessive reuse (67%) slightly constrains representational flexibility, reducing adaptability in dynamic regions. Table 3 reports final reconstruction quality on the validation set. Moderate attention reuse improves reconstruction quality, with



(a) Context (t=90) (b) Ground Truth (t=93)

(c) 0% reuse (d) 33% reuse

(e) 50% reuse (f) 67% reuse

Fig. 1: Qualitative comparison of reconstruction quality across attention reuse ratios. Given the context frame at t=90, predictions at t=93 are compared across reuse settings. Moderate reuse (50%) produces sharper details, whereas excessive reuse (67%) reduces flexibility.

the 50% reuse configuration achieving the highest PSNR (27.07) and SSIM (0.777). Both lower (33%) and higher (67%) reuse ratios lead to reduced performance, indicating a non-monotonic relationship between reuse ratio and reconstruction quality. Moderate attention reuse can introduce a beneficial inductive bias by stabilizing spatiotemporal attention patterns, while excessive reuse constrains representational flexibility in deeper layers.

Table 3: Final reconstruction quality on the validation set.

Less-Attention	**Final PSNR**	**Final SSIM**
0%	26.8205	0.7701
33%	26.5433	0.7544
50%	**27.0684**	**0.7771**
67%	26.7903	0.7583

4.4 Training Dynamics

Fig. 2 shows training PSNR over the full 150k iterations. All configurations exhibit similar convergence behavior during early training, indicating that attention reuse does not destabilize optimization. As training progresses, the 50% reuse model consistently achieves higher PSNR, while the 67% reuse configuration shows increased variance and lower average performance. Although this reflects training behavior rather than generalization, moderate attention reuse thus acts as structural regularization, improving reconstruction fidelity without accelerating convergence

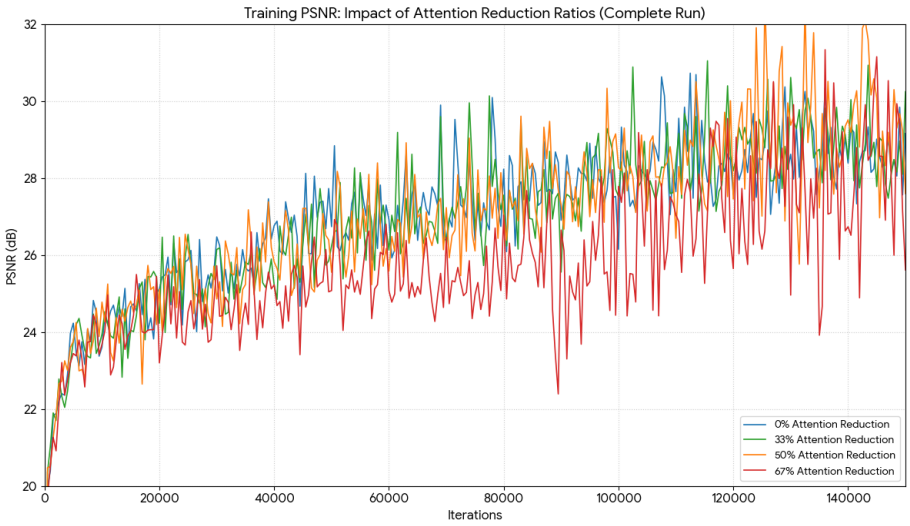


Fig. 2: Training PSNR over 150k iterations for different attention reuse ratios.

5 Discussion

Our experiments reveal a clear separation between the computational and representational effects of attention reuse in spatiotemporal 3D reconstruction. Despite its theoretical appeal, attention reuse does not yield wall-clock speedups over optimized full attention in STORM, even after substantial implementation-level optimizations. This behavior can be attributed to unfavorable memory access patterns and the dominance of fused full-attention kernels on modern GPUs. Surprisingly, however, we observe consistent improvements in reconstruction quality at moderate attention reuse ratios, indicating that attention reuse alters the inductive bias of the Transformer in a beneficial way. In particular, reusing attention maps across layers appears to stabilize spatiotemporal aggregation, leading to improved photometric fidelity without sacrificing global context. Our results show that attention reuse in dynamic 3D reconstruction acts primarily as a structural regularizer rather than a practical acceleration mechanism.

Why Attention Reuse Does Not Improve Training Time. Attention reuse fails to deliver runtime speedups over standard self-attention in STORM, for two reasons. First, modern full-attention implementations rely on fused kernels that minimize memory traffic and maximize on-chip data reuse. Second, Less-Attention operates on dense $N \times N$ matrices, and even with low-rank parameterizations, the resulting memory bandwidth demands dominate execution time at the sequence lengths encountered in STORM Consequently, the theoretical reduction in arithmetic complexity does not translate into improved runtime under realistic hardware constraints. The absence of speedups reflects a fundamental mismatch between the operational regime of attention reuse and the execution characteristics of modern GPUs, rather than shortcomings in implementation.

Why Moderate Attention Reuse Improves Reconstruction Quality. Attention reuse consistently improves reconstruction quality at moderate reuse ratios. The 50% reuse configuration achieves the highest PSNR and SSIM across all evaluated settings. Reusing attention maps encourages stable spatiotemporal correspondence patterns, reducing layer-wise variability in attention allocation. In dynamic multi-view reconstruction, it promotes aggregation of geometric and temporal information across views and timesteps. From this perspective, attention reuse acts as a form of structural regularization, analogous to weight tying or parameter sharing in recurrent or lightweight Transformer architectures. The non-monotonic performance trend further supports this interpretation. While moderate reuse stabilizes attention dynamics, excessive reuse constrains representational flexibility in deeper layers, limiting the model's ability to adapt attention patterns to increasingly abstract features.

Training Dynamics and Optimization Behavior. Attention reuse does not adversely affect optimization stability. However, higher reuse ratios introduce increased variance in training performance, particularly at later stages, consistent with reduced expressive capacity. The alignment between training and validation

metrics confirms that quality gains reflect sustained representational improvement rather than faster convergence.

Implications for Efficient Transformer Design. Our findings highlight the limitations of evaluating efficient attention mechanisms solely through asymptotic complexity analysis. In practice, the effectiveness of such methods depends critically on sequence length, feature dimensionality and hardware execution characteristics.For models with optimized full-attention kernels, manipulating dense attention matrices incurs memory overheads that outweigh theoretical benefits. Future architectures could explicitly decouple these roles, combining stable attention reuse for regularization with selective recomputation to preserve expressiveness.

6 Conclusion and Future Work

In this work, we presented a hardware-aware evaluation of attention reuse in STORM, a feed-forward framework for dynamic 3D Gaussian reconstruction. Our results show that attention reuse does not provide training time speedups over optimized full attention. This behavior arises from the dominance of fused attention kernels and the high memory bandwidth demands of dense attention matrix transformations. We observe consistent improvements in reconstruction quality at moderate reuse ratios, indicating that attention reuse can act as a form of structural regularization that stabilizes spatiotemporal attention dynamics.

Our findings highlight a fundamental disconnect between theoretical complexity analyses and practical performance in modern Transformer architectures. They suggest that efficient attention mechanisms should be evaluated not only for computational savings, but also for their impact on representation learning. The current study focuses on a single dynamic reconstruction framework and evaluates attention reuse within the constraints of existing deep learning frameworks. Custom CUDA kernels that could fuse attention transformations and value aggregation are not explored. Also, regimes with substantially longer temporal sequences where attention reuse may become computationally advantageous are not considered. Future work should explore kernel fusion, lower-rank parameterizations and longer temporal sequences to better characterize the limits of attention reuse..

References

1. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Vedaldi, A., Cham, T.J., Cai, J.: MVSplat: Efficient feed-forward 3D gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision, ECCV (2024)
2. Child, R., Gray, S., Radford, A., Sutskever, I.: Generating long sequences with sparse transformers. In: Advances in Neural Information Processing Systems, NeurIPS (2019)

3. Choromanski, K., Likhosherstov, V., Dohan, D., Song, X., Davis, J., Sarlos, T., Hawkins, P., Belanger, D., Colwell, L., Weller, A.: Rethinking attention with Performers. In: International Conference on Learning Representations, ICLR (2021)
4. Dao, T., Fu, D.Y., Ermon, S., Rudra, A., Ré, C.: FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In: Advances in Neural Information Processing Systems, NeurIPS (2022)
5. Gao, Q., Xu, Q., Cao, Z., Mildenhall, B., Ma, W., Chen, L., Tang, D., Neumann, U.: GaussianFlow: Splatting gaussian dynamics for 4D content creation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 20815–20825 (2024)
6. Katharopoulos, A., Vyas, A., Pappas, N., Fleuret, F.: Transformers are RNNs: Fast autoregressive transformers with linear attention. In: International Conference on Machine Learning, ICML. pp. 5156–5165. PMLR (2020)
7. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (TOG) **42**(4), 1–14 (2023)
8. Li, X., Xu, Y., Sun, J., Huang, Z., Zhang, Z.: Reusing attention in transformer layers. arXiv preprint arXiv:2301.01851 (2023), https://arxiv.org/abs/2301.01851
9. Li, Z., Chen, Z., Li, Z., Xu, Y.: 4D gaussian splatting for real-time dynamic scene rendering. arXiv preprint arXiv:2310.08528 (2023), https://arxiv.org/abs/2310.08528
10. Li, Z., Niklaus, S., Snavely, N., Wang, O.: Neural scene flow fields for space-time view synthesis of dynamic scenes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 6038–6048 (2021)
11. Liang, Z., Huang, J.W., Li, W.B., Wang, L.Z., Lu, L., Cheng, Y.C., Chen, B.Q.: Deformable 3D gaussians for high-fidelity dynamic scene reconstruction. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 21323–21332 (2024)
12. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. In: IEEE/CVF International Conference on Computer Vision, ICCV. pp. 5865–5874 (2021)
13. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: HyperNeRF: A higher-dimensional representation for topologically varying neural radiance fields. ACM Transactions on Graphics (TOG) **40**(6), 1–12 (2021)
14. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-NeRF: Neural radiance fields for dynamic scenes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 10318–10327 (2021)
15. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in perception for autonomous driving: Waymo open dataset. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR (2020)
16. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: LGM: Large multi-view gaussian model for high-resolution 3D content creation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 8510–8519 (2024)
17. Wang, Q., Wang, Z., Cuenco, K., Pyros, C., Vichare, R., Huang, J.B.: IBRNet: Learning-based view synthesis from unstructured imagery. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 4689–4699 (2021)

18. Wang, S., Li, B.Z., Khabsa, M., Ma, H., Ma, H.: Linformer: Self-attention with linear complexity. In: Advances in Neural Information Processing Systems, NeurIPS. vol. 33, pp. 13060–13070 (2020)
19. Yang, J., Huang, J., Chen, Y., Wang, Y., Li, B., You, Y., Igl, M., Sharma, A., Karkus, P., Xu, D., Ivanovic, B., Wang, Y., Pavone, M.: STORM: Spatio-temporal reconstruction model for large-scale outdoor scenes. In: International Conference on Learning Representations. vol. 2025, pp. 50446–50465 (2025)
20. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelNeRF: Neural radiance fields from one or few images. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR. pp. 4578–4587 (2021)
21. Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., Ahmed, A.: Big Bird: Transformers for longer sequences. In: Advances in Neural Information Processing Systems, NeurIPS (2020)
22. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: GS-LRM: Large reconstruction model for 3D gaussian splatting. arXiv preprint arXiv:2404.19702 (2024), https://arxiv.org/abs/2404.19702
23. Zhang, S., Liu, H., Lin, S., He, K.: You only need less attention at each stage in vision transformers. arXiv preprint arXiv:2406.00427 (2024), https://arxiv.org/abs/2406.00427