

Block-Sparse Global Attention for Efficient Multi-View Geometry Transformers

Chung-Shien Brian Wang*

Christian Schmidt*

Jens Piekenbrinck

Bastian Leibe

Computer Vision Group
 RWTH Aachen University

<https://vision.rwth-aachen.de/sparse-vggt>

Abstract

Efficient and accurate feed-forward multi-view reconstruction has long been an important task in computer vision. Recent transformer-based models like VGGT, π^3 and MapAnything have demonstrated remarkable performance with relatively simple architectures. However, their scalability is fundamentally constrained by the quadratic complexity of global attention, which imposes a significant runtime bottleneck when processing large image sets. In this work, we empirically analyze the global attention matrix of these models and observe that the probability mass concentrates on a small subset of patch-patch interactions corresponding to cross-view geometric correspondences. Building on this insight and inspired by recent advances in large language models, we propose a training-free, block-sparse replacement for dense global attention, implemented with highly optimized kernels. Our method accelerates inference by more than $3\times$ while maintaining comparable task performance. Evaluations on a comprehensive suite of multi-view benchmarks demonstrate that our approach seamlessly integrates into existing global attention-based architectures such as VGGT, π^3 , and MapAnything, while substantially improving scalability to large image collections.

1. Introduction

Reconstruction of 3D geometry and camera motion from multiple images is a central, long-standing problem in computer vision that has a broad impact across domains such as autonomous driving, embodied agents, AR/VR or photogrammetry. Classical pipelines approach this task using optimization-based explicit geometric modeling and iterative refinement, most notably bundle adjustment, yielding reliable results at the cost of significant runtime overhead [28, 31]. More recently, feed-forward models [24, 38, 43] have narrowed the gap to structure-from-motion (SfM)

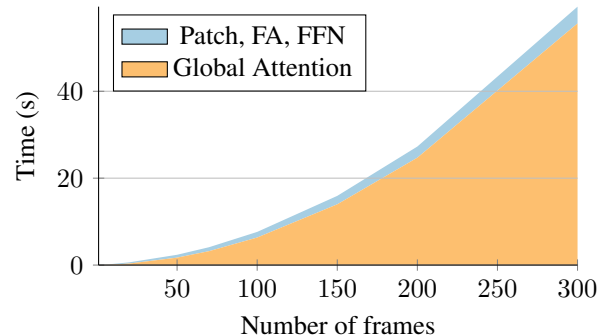


Figure 1. **Runtime of VGGT’s forward pass.** FA denotes frame-wise attention. As the number of input frames increases, global attention dominates the computational cost (measured with FlashAttention2 [8] on an H100 GPU at resolution 518^2). We propose to adapt a block-sparse attention method that considerably reduces the cost of Global Attention while preserving result quality.

systems such as COLMAP [31]. A prevalent concept in such models, originating with DUST3R [43], is to estimate 3D geometry and relative cameras from image-pairs and subsequently consolidate predictions across multiple views.

The recently proposed *Visual Geometry Grounded Transformer (VGGT)* [41] is a particularly simple and effective multi-view geometry estimation model, achieving state-of-the-art performance on reconstruction, pointmap estimation and point tracking. A key architectural design element of VGGT is the use of global attention blocks in the decoder, which enables holistic, scene-level reasoning in a single forward pass. However, the complexity of global attention grows quadratically with the number of input images, thereby becoming the dominant computational contributor even at moderate sequence lengths, as illustrated in Fig. 1, which limits scalability to larger image collections.

By inspecting the global attention maps during a forward pass of the model, visualized in Fig. 3, we observe that each token only attends to a small subset of tokens, leading to sparse attention matrices. This observation motivates our systematic analysis of these sparsity patterns, suggesting that a substantial portion of the computational complex-

*Equal contribution.

ity of the global attention layers can potentially be avoided without degrading task performance.

Building on these findings and inspired by recent advances in large language models [15], we propose a novel adaptive block-sparse attention mechanism that approximates the full attention map and computes attention using block-sparse kernels whose cost scales with the number of active blocks rather than the full quadratic size, ultimately accelerating inference. The additional module is lightweight and does not require any optimization, avoiding the need of extra annotations or backward passes through the original VGGT model. With extensive experiments across multiple datasets, we empirically demonstrate that our method achieves task performance on par with the original VGGT model, while substantially accelerating end-to-end inference speed more than $3\times$.

To further demonstrate the versatility of our approach, we apply the proposed block-sparse attention mechanism to the recent π^3 variant of VGGT, designed for permutation-invariant geometry estimation [44], and to MapAnything [21], which differs architecturally from VGGT yet similarly employs global cross-view attention. We observe that both models exhibit similar accuracy–efficiency trade-offs, underscoring the generality of our method.

Overall, our contributions are threefold: (1) an analysis of sparsity-patterns in global attention layers of multi-view geometry estimation transformer models; (2) a novel adaptive block-sparse attention mechanism to accelerate inference speed without requiring optimization; and (3) comprehensive empirical evaluations demonstrating matched accuracy with significantly reduced compute.

2. Related Work

2.1. Feed-Forward Multi-View Reconstruction

Traditional multi-view reconstruction pipelines follow the *Structure from Motion (SfM)* paradigm, jointly estimating scene geometry and camera poses [16, 28, 31, 32]. Typical implementations perform feature extraction and matching, triangulation, and bundle adjustment. An additional dense depth estimation stage is necessary if a dense reconstruction is desired. COLMAP [31] remains a widely adapted pipeline and is frequently employed as baseline or used to produce ground-truth annotations for datasets.

Learning-based approaches for multi-view reconstruction aim to enhance or replace these pipelines. A prominent line of work employs a cross-attention decoder to match two input frames [12, 18, 24, 43]. Processing multiple views requires additional global alignment post-processing that consolidates pairwise predictions into a single, consistent scene representation [12, 24, 43]. This consolidation is computationally expensive and introduces a separate source of error.

To mitigate the cost of globally aligning image-pairs,

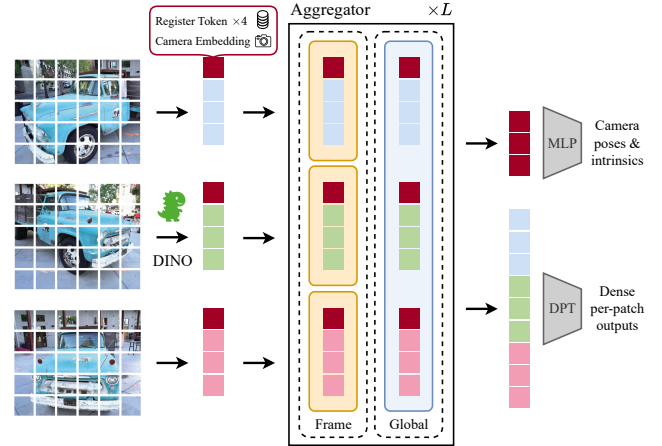


Figure 2. **Architecture overview of VGGT [41].** The key component is the Aggregator consisting of $L = 24$ alternating attention blocks (first frame-wise attention, then global attention over all frames). Each input frame is augmented with five learned embedding vectors: one camera token and four register tokens. After the Aggregator, VGGT regresses camera poses from the camera tokens using a light-weight MLP head, and dense outputs (point maps, depth, point tracks) using DPT heads [29].

several derivative works incrementally extend the reconstruction [13, 40, 42] by iteratively applying two-view models. Some maintain a persistent latent representation to accumulate information [40, 42], while Light3r-SfM [13] directly updates a global reconstruction and shows improved performance on large scenes. Despite their practicality, iterative schemes may not perform as well as methods that consider all views simultaneously. Incremental updates are order-sensitive, prone to error accumulation and drift, and require revisiting past states or additional optimization to enforce long-range, cross-view consistency. In contrast, globally conditioned models can exploit holistic context to align distant views and resolve ambiguities in a single forward pass, albeit at higher computational cost.

2.2. Global Attention for Fast 3D Processing

In contrast to two-view models, architectures with global attention mechanisms are able to reason jointly over all views at once, enabling scene-level consistency [3, 41, 44, 45]. Fast3r [45] combines a per-frame feature encoder with a decoder composed exclusively of global attention blocks, hence lacking frame-wise attention blocks. In addition, it predicts both local and global pointmaps for every view, a design inherited from two-view models.

VGGT [41] simplifies reconstruction by predicting all pointmaps in the coordinate system of a reference view, *i.e.* the first view. Its decoder alternates between frame-wise and global attention, allowing per-frame features to be adapted before the next round of global matching. For reference, we provide an overview of the VGGT architecture in Fig. 2. Since VGGT predicts all pointmaps with

respect to the reference view and employs camera embeddings, its predictions are sensitive to input ordering. To alleviate this issue, π^3 [44] removes the camera embeddings to attain permutation invariance while retaining per-frame register tokens akin to VGGT. The authors report slightly improved reconstruction quality over VGGT on most benchmarks, with notably reduced variance across input permutations. MapAnything [21] adopts a distinct transformer design for metric 3D reconstruction capable of accommodating optional geometric inputs such as intrinsics, poses, or partial depth alongside RGB images.

Despite their architectural differences, these methods share a reliance on global attention, whose time complexity scales quadratically with the number of tokens, which motivates our approach to exploit the sparsity of global attention to preserve reconstruction performance while reducing computational overhead.

2.3. Sparse Attention

Attention is the foundation of many state-of-the-art neural architectures, particularly large language models (LLMs). Recent research has focused on reducing the quadratic complexity of dense attention by exploiting structure within the attention maps [4, 5, 15, 49]. LLM-oriented sparse attention methods [20, 23, 26, 47] rely on autoregressive assumptions, like causal masking or key-value caching during inference, that do not directly transfer to attention over continuous, two-dimensional token grids used in vision models. Other works [17, 50] leverage temporal or spatial continuity, assumptions that do not hold for global attention in multi-view vision transformers.

In contrast to weight sparsity and pruning approaches [14, 37, 46] that compress model weights, our method targets attention sparsity, *i.e.* sparsity emerging in the attention maps themselves. Combining these two forms of sparsity could potentially lead to further acceleration.

Chen *et al.* [4] approximate attention by combining sparse and low-rank representations, but their irregular memory access pattern is inefficient on block-oriented accelerators. A more generally applicable approach is block-sparse attention [5], which combines the hardware-friendly, block-wise computation of FlashAttention [9] with a sparse block mask. While PixelatedButterfly [5] supports both weight and attention sparsity, it relies on static patterns and learned parameters that require optimization. SeerAttention [15] introduces a learned gating mechanism, which requires optimization, with a top- k block selection strategy. SpargeAttention [49] recently showcased the effectiveness of block-sparse attention across diverse models and modalities. It is training-free, combining self-similarity-based and CDF-thresholded block selection with a sparse online softmax computation. To increase self-similarity and enable even sparser block selections, it applies Hilbert-curve per-

mutations to spatially reorder tokens prior to attention.

Building on the block-sparse paradigm, we leverage the attention kernel of SpargeAttention [49] to accelerate inference of VGGT [41], π^3 [44] and MapAnything [21]. Unlike SpargeAttention, we omit self-similarity-based selection, Hilbert-curve permutation and the sparse online softmax. Instead, we combine CDF-based and top- k thresholding for block selection and retain dense attention for special tokens. While Zhang *et al.* [49] reported results with sequences up to 128K tokens and 54% sparsity under causal attention, our largest configuration processes sequences exceeding 512K tokens at a sparsity ratio above 75%. We achieve speed-ups of more than $3\times$ while preserving task performances across all models.

3. Analyzing VGGT’s Attention Patterns

Traditional multi-view reconstruction pipelines rely heavily on hand-crafted features, *e.g.* SIFT [25], in order to establish sparse correspondences that drive structure and camera estimation [31, 32]. Our analysis of the global attention in VGGT reveals a strong concentration on a limited set of token pairs, resulting in sparse patterns that are analogous to correspondence matrices and that mirror geometric matching. The key idea of our approach is to exploit these sparsity patterns to improve the efficiency of global attention.

3.1. Background: VGGT’s Architecture

An overview of the VGGT architecture is illustrated in Fig. 2. We briefly recapitulate the most important parts of the architecture for completeness.

VGGT consists of a patchifier P , a feature aggregator A , and several task-specific heads $H_1, \dots, H_T$. The patchifier P independently transforms each input image I into a set of patch tokens $T = P(I)$. In VGGT, P is implemented as a pre-trained DINOv2-Large [10, 27] with a patch size of 14×14 pixels [41]. For every frame, VGGT concatenates five special tokens to the patch tokens: a camera token and four register tokens [10]. To distinguish between the reference view and auxiliary frames, VGGT maintains one set of special tokens for the first frame and another set for all other frames. The aggregator A is a series of transformer blocks alternating between global and frame-wise attention. Global attention computes full self-attention over the union of all patch and special tokens across all frames, which enables scene-level reasoning. Frame-wise attention performs self-attention between the patch and special tokens of a single frame, which supports per-frame adaptation. Both frame-wise and global attention layers are followed by two-layer per-token MLPs, as is common in transformer architectures [11]. After the final aggregator layer, the register tokens are discarded and the remaining tokens are passed to the task-specific heads. VGGT employs a lightweight head for pose regression and DPT [29] heads for dense prediction

tasks, *i.e.* depth and pointmaps. Our analysis focuses on the global attention layers inside the aggregator.

The follow-up model π^3 [44] closely matches VGGT’s architecture with just a few modifications to promote permutation invariance. Architecturally, π^3 uses fewer alternating attention blocks in the aggregator, 18 instead of 24, replaces DPT with plain Transformer heads for dense prediction tasks and removes camera embeddings while retaining per-frame register tokens. Methodologically, π^3 supervises relative poses between all frames rather than coordinates tied to the reference view. The authors show that these modifications improve the model’s task performance and reduce sensitivity to input permutations [44].

3.2. Visualizing Attention Maps

We visualize the full post-softmax attention map for a global attention block in a middle decoder layer, layer 15, in Fig. 3. Only a small fraction of entries carry non-negligible probability mass, indicating pronounced sparsity. To study how this behavior evolves with layer depth, Fig. 4 (right) plots the average and the maximum activation of different parts of the attention map against the layer index. We observe a peak in the maximum activation values in the middle of the aggregator. This effect is concentrated in patch-patch interactions, whereas attention involving special tokens remains stable across layers. The sparse, highly activated entries align with geometrically meaningful correspondences between frames, as illustrated in Fig. 4 (left).

We hypothesize that the model learns to perform an exhaustive correspondence search over the layers of the aggregator. More concretely, the global attention proposes cross-view correspondences and exchanges information across frames, while the frame-wise attention adapts patch token features to improve matches in the subsequent global layers. Fig. 4 (right) shows that the mid-aggregator layers exhibit the strongest selective attention and appear to contribute disproportionately to multi-view reasoning.

3.3. Are Some Layers More Important?

To further test our hypothesis that the mid-layers of the aggregator are most important for task performance, we conduct a layer-drop ablation in which selected aggregator blocks are skipped at inference. We show the ablation results in Fig. 5 and show additional results in the supplementary material. The figure shows that the model is comparatively robust when early or late aggregator layers are skipped. In contrast, omitting a single middle layer leads to a pronounced performance drop. This behaviour supports our hypothesis that the middle of the aggregator concentrates the essential cross-view information exchange and motivates our adaptive block-selection strategy.

3.4. Consequences of this Analysis

Our analysis yields three take-away messages. First, global attention is highly sparse. Second, the variation in attention entries is dominated by patch–patch interactions, while special token interactions remain stable across depth. Lastly, mid-stack layers carry the key cross-view integration, as confirmed by the layer-drop ablation. These findings indicate that dense quadratic attention is not necessary throughout the stack, but that considerable efficiency gains can be realized by exploiting sparsity, in particular in the important middle layers. Guided by this analysis, we propose to replace the dense global attention layers with an adaptive block-sparse variant. This reduces the time complexity with minimal accuracy loss and integrates into VGGT and π^3 without changes to their encoders or task heads.

4. Method

In this section, we describe our method for block-sparse attention in the global attention layer in VGGT (Sec. 4.1) and how we handle camera and register tokens (Sec. 4.2). Note that while we utilize the implementation of SpargeAttention [49], block-sparse attention in itself is independent of that particular work and could, *e.g.*, be implemented on top of FlashAttention [8, 9, 51] to yield similar results. We provide a theoretical analysis of the achievable speed-up for a certain architecture in the appendix.

4.1. Block-Sparse Attention

Self-Attention. Self-Attention is the cornerstone of many modern neural networks, as it enables dynamic, global interactions between input elements. Given input tokens $\mathbf{X} \in \mathbb{R}^{n \times d}$, queries $\mathbf{Q}$, keys $\mathbf{K}$, and values $\mathbf{V}$ are computed as linear projections of $\mathbf{X}$, such that $\mathbf{Q} = \mathbf{XW}^Q$, $\mathbf{K} = \mathbf{XW}^K$, $\mathbf{V} = \mathbf{XW}^V$, and the scaled dot-product attention is defined as

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{QK}^\top}{\sqrt{d_h}}\right) \mathbf{V}, \quad (1)$$

where d_h is the embedding dimension of a single attention head. Multi-head attention concatenates H such projections to enhance expressivity [39]. The bottleneck of this operation is the full attention matrix $\mathbf{A} = \mathbf{QK}^\top$, which grows quadratically with the number of input tokens. For a moderate input length of 10 frames at resolution 294×518 , the global attention matrix would already contain $(10 \cdot \frac{294 \cdot 518}{14^2})^2 \approx 1.2 \cdot 10^8$ elements, corresponding to more than 100MB at half precision. At 1000 frames, the required space would increase to more than 1 TB.

Block-Sparse Attention. Sparse attention reduces the quadratic cost of full attention by constraining the pattern of non-zero entries in $\mathbf{QK}^\top$. Instead of attending to all po-

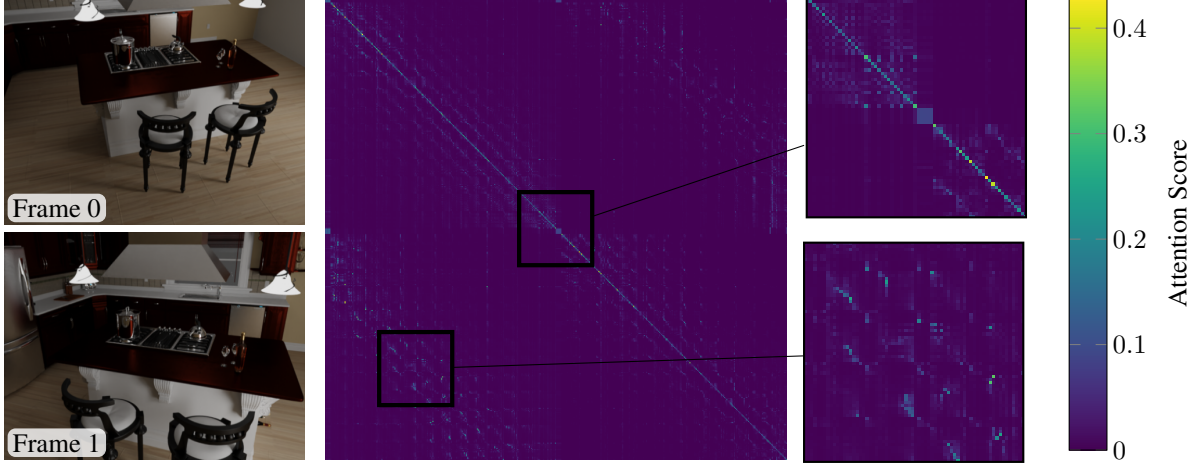


Figure 3. **Visualization of VGGT’s global attention matrix.** A very small number of entries are highly activated, while the vast majority of entries are near zero. This visualization shows the average attention map over all heads of layer 15 in the VGGT aggregator, at an input resolution of 224×182 . Upper highlight: The special tokens attend to each other and form a distinctive pattern. Lower highlight: Patch-level attention is localized on a small subset of highly activated entries. See the supplementary material for an enlarged visualization.

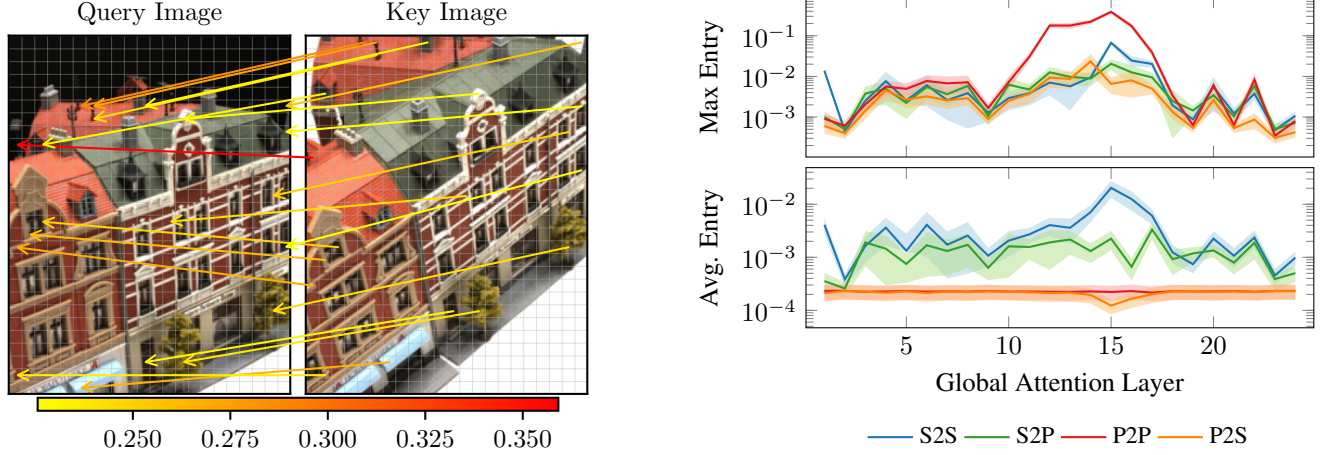


Figure 4. **VGGT’s global attention matrix is extremely sparse.** Left: We visualize the tokens corresponding to the top- k activated entries of the attention map of layer 15. Right: Average & maximum attention scores in the global attention maps; the shorthand $\{S,P\}2\{P,S\}$ denotes attention between special (S) and patch (P) tokens. Solid lines denote the mean over all tokens, while shaded regions denote the standard deviation. Increasing indices indicate layers closer to the output. Layers in the middle of the aggregator exhibit higher activations and increased sparsity. Note the different scalings of the mean and max activations.

sitions, only predefined entries are computed:

$$\text{SparseAttn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{(\mathbf{Q}\mathbf{K}^\top) \odot \mathbf{M}}{\sqrt{d_h}}\right) \mathbf{V}, \quad (2)$$

where $\mathbf{M}$ is a binary sparsity mask and the $\odot$ operator denotes the element-wise (Hadamard) product. In practice, block-sparse masks are preferred, as they align with modern hardware accelerators, enabling efficient memory access and parallelization [2, 6, 9].

Predicting Block Masks. To predict the importance of blocks from queries and keys, we first apply average pooling with block size b to the queries $\mathbf{Q}$ and keys $\mathbf{K}$, yielding pooled representations $P^b(\mathbf{Q})$ and $P^b(\mathbf{K})$. We then compute their similarity matrix $S = P^b(\mathbf{Q})P^b(\mathbf{K})^\top$ and apply

a softmax to obtain probability distributions over blocks. The resulting distributions provide a natural ranking of candidate blocks, from which we derive a binary block-sparsity mask using combined top- k and cumulative density function (CDF) thresholding. The mask directly determines the query-key blocks to evaluate, and can be used directly with standard block-sparse attention kernels. A graphical overview is provided in Fig. 6. We experimented with an additional linear projection layer after pooling queries and keys, but found no improvement over the baseline.

Selecting Blocks. We select blocks using two complementary criteria: a CDF threshold τ and a sparsity ratio ρ . The CDF threshold ensures the selected set covers at least a τ fraction of predicted cumulative probability. The spar-

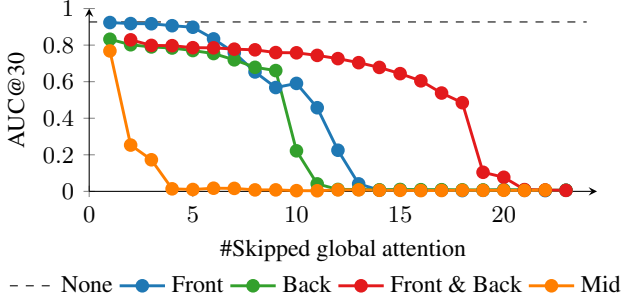


Figure 5. **Influence of dropping global attention layers.** We skip the computation of different global attention layers in the aggregator starting with the earliest (Front), last (Back), alternating (Front & Back), or from the middle layers (Middle), and evaluate pose estimation on CO3Dv2 [30]. The x-axis denotes the total number of skipped layers. The experiment shows that the model is especially sensitive to pruning of the center layers, and robust against pruning the early and late layers.

sity ratio acts as a lower bound, enforcing a minimum of $k = \lfloor B \cdot (1 - \rho) \rfloor$ top-ranked blocks, where B denotes the total number of blocks. The two criteria complement each other: In uniform layers, a fixed sparsity ratio alone may admit too few blocks, while the CDF threshold guarantees sufficient coverage; in sparse layers, the threshold may be met with very few blocks, while the sparsity ratio ensures that at least a reasonable minimum is preserved. Together, they adapt block selection to diverse sparsity patterns while maintaining stability and efficiency.

Comparison to SpargeAttention. While our method is related to SpargeAttention [49], the approaches differ in design and implementation trade-offs. SpargeAttention introduces additional mechanisms such as self-similarity filtering and warp-level PV pruning, which increases implementation complexity and maintenance overhead across GPU generations. In contrast, our method is kernel-agnostic. It produces a binary block mask from pooled query-key similarities, and blocks are chosen based on a fixed sparsity ratio and a CDF threshold, hence compatible with any standard block-sparse attention kernel. This decoupling from hardware-specific optimizations makes our approach simpler to implement, easier to maintain, and more robust to future accelerator iterations.

4.2. Handling Special Tokens

VGGT employs register tokens [10] in addition to learned camera tokens which differentiate between frames. We find that these tokens behave qualitatively differently from the regular patch tokens, and pooling them together results in worse performance for similar sparsity ratios. We therefore split the tokens into two sets: $\mathbf{X}_p$ containing the patch tokens, and $\mathbf{X}_s$, containing the special tokens, *i.e.* register and camera tokens. We only predict and use the block sparse

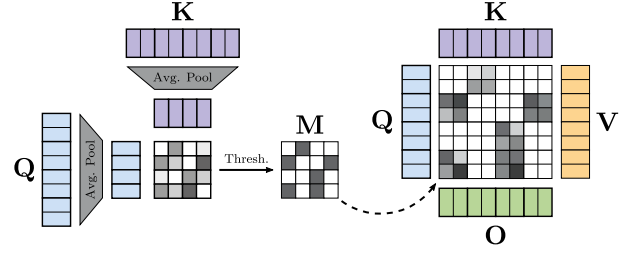


Figure 6. **Overview of the training-free adaptive sparse attention.** Keys and queries are average pooled to get a low-resolution approximation of the attention map. This low-resolution attention map is used to create the binary mask for block-sparse attention by Top-K selection and adaptive thresholding.

map on the patch tokens $\mathbf{X}_p$, and compute the full attention on $\mathbf{X}_s$, as well as cross-attention between $\mathbf{X}_s$ and $\mathbf{X}_p$ and vice-versa. We find that this strategy is essential to avoid large drops in task performance.

5. Experiments

We extend three large reconstruction models, VGGT [41], π^3 [44], and MapAnything [21] with the described sparse global attention mechanism and evaluate the performance impact on common multi-view benchmarks. In particular, we evaluate the robustness of the models against varying *effective sparsity ratios*, which we define as the ratio of computed entries of the attention map to the total number of entries. As the number of input frames increases, the theoretical speed-up from a sparsity ratio of x approaches $\frac{1}{1-x}$, since the overall computational costs are dominated by the global attention. Processing a 200 frame sequence with sparsity ratio of 0.75, for example, will accelerate the global attention by a factor of four. Depending on the ratio of global attention to the overall runtime, the end-to-end speed-up will be lower. A comprehensive analysis of the achievable speed-up is provided in the supplementary.

In the main text, we show results for common benchmarks on relative pose estimation (Real Estate 10K [52], Common Objects in 3D [30] TUM [36], and ScanNet [7]), and pointmap estimation (7Scenes [35], NRGBD [1], DTU [19], and ETH3D [33]). Additionally, we show results for scene-level pose estimation on the Tanks & Templates dataset [22]. In the supplementary materials, we also show results on longer sequences of the ScanNet [7] dataset.

5.1. Regression Tests

We evaluate our sparsified model on relative pose and pointmap estimation, using a range of common benchmark datasets. Unless indicated otherwise, we closely follow the setting of the original VGGT paper [41]. For a comprehensive high-level overview of the results, we visualize the sparsity-performance trade-off in Fig. 7. The results show

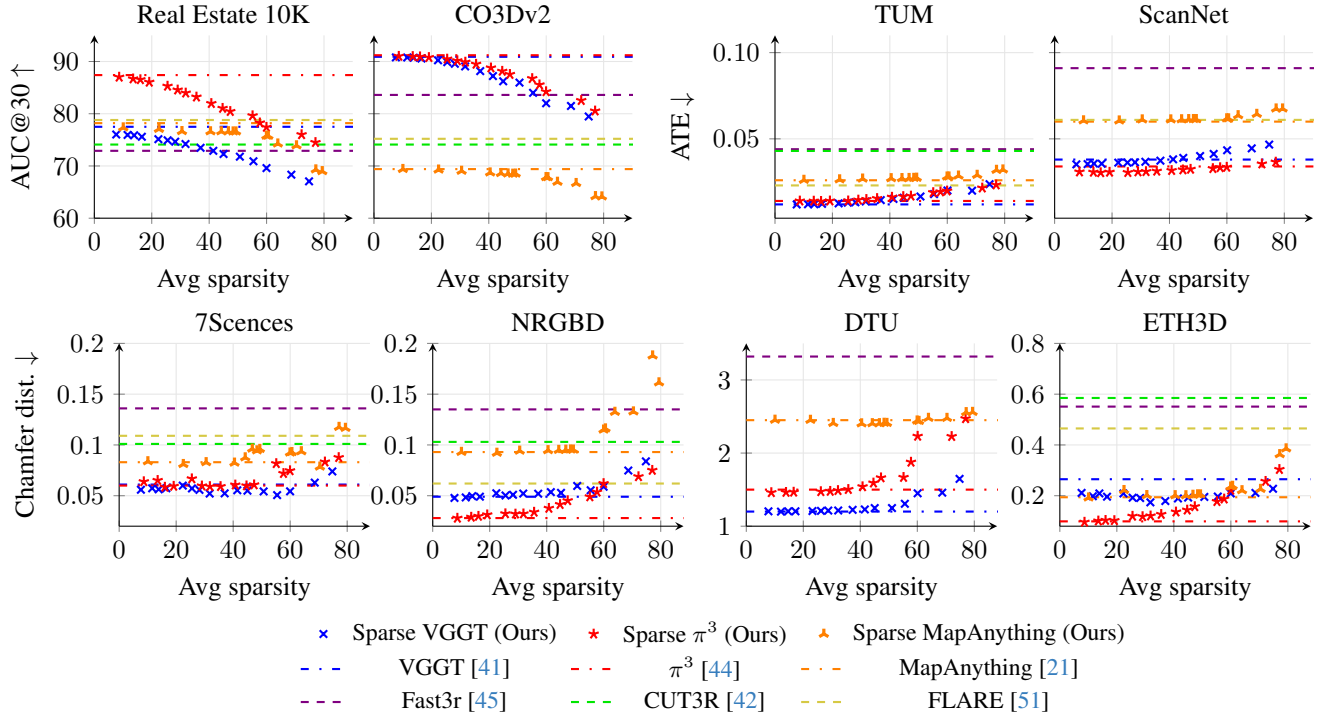


Figure 7. **Results for Relative Pose Estimation (top) Multi-View Reconstruction (bottom).** Multi-view reconstruction performance seems to be robust against sparsification of global attention; even in the highest sparsity settings, the results are on par or better than other state-of-the-art methods. We provide comprehensive tables for these results in the supplementary material.

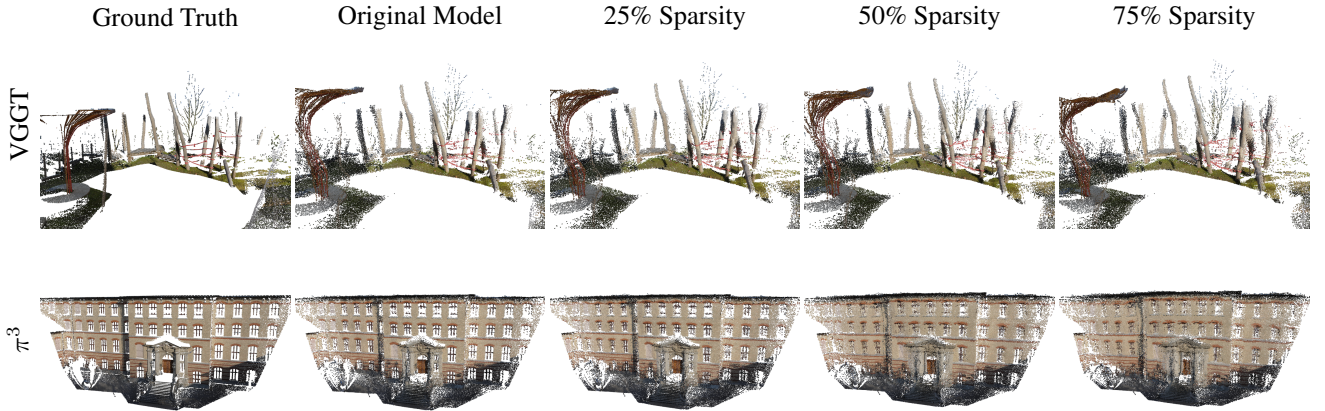


Figure 8. **Qualitative examples.** We show examples from the ETH3D dataset [33]. Increasing sparsity leads to small perturbations in the reconstruction, but the overall quality stays remarkably high. Additional qualitative examples can be found in the supplementary material.

that task performance degrades comparatively little with increasing sparsity. Both VGGT and π^3 achieve results comparable to other state-of-the-art models even at high sparsity ratios, which is further corroborated by the qualitative examples provided in Fig. 8.

Pose Estimation involves determining the 6DoF pose for each input view. We evaluate AUC@30 on Common Objects in 3D [30] and Real Estate 10K [52], as well as ATE on TUM Dynamic [36] and ScanNet [7]. For these experiments on TUM and ScanNet, we follow the setup of previ-

ous works [44, 48] and sample 90 frames with a temporal stride of 3 from the beginning of each sequence. We show aggregate results in Fig. 7 (top). While we observe a continuous decrease in pose accuracy for increasing sparsity levels, the modified model still performs comparable to other state-of-the-art methods.

Multi-view Pointmap Estimation is the task of predicting the surface 3D point for each pixel in all input images. We evaluate on 7Scenes [35], NRGD [1], ETH3D [33], and DTU [19] following π^3 [44] and VGGT [41]. We show a

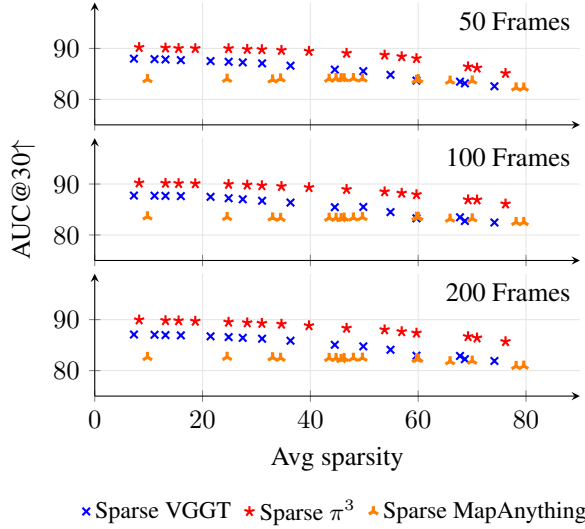


Figure 9. **Pose estimation on Tanks & Temples for different input sizes and sparsity ratios.** We show more comprehensive results in the supplementary materials.

compact representation of the results in Fig. 7 and full result tables in the supplementary. We observe similar trends as before: task performance stays at an acceptable level even at high sparsity ratios. The apparent improvements on ETH3D are likely attributable to randomness. Overall, we do not expect the sparse models to outperform the baseline.

5.2. Long-Sequence Relative Pose Estimation

To test the scalability of our block-sparse attention scheme, we evaluate pose estimation on the Tanks & Temples benchmark [22]. We report AUC@30 in Fig. 9, for which we observe very little degradation with increasing effective sparsity. On sequences with 200 frames, our sparse π^3 model with an effective sparsity ratio of 75% runs around twice as fast as the baseline model on an H100 GPU; on longer sequences, the speed-up increases further. An analysis of the achievable speed-up, time measurements, as well as similar experiments on long sequences from the ScanNet [7] dataset can be found in the supplementary materials.

5.3. Ablations

To validate our design decisions, we compare our method to SpargeAttention [49] and a random masking baseline in Fig. 10; we provide additional results in the supplementary material, including a comparison to SeerAttention [15]. As expected, performance with random masking drops rapidly as the sparsity ratio increases. SpargeAttention utilizes CDF-thresholding as the main sparsity mechanism and keeps additional tokens based on intra-block similarity thresholding; for low sparsity ratios, their performance is similar to ours, but with increasing sparsity performance drops. Since our method combines Top-K and CDF thresholding, it is more robust against high sparsity ratios.

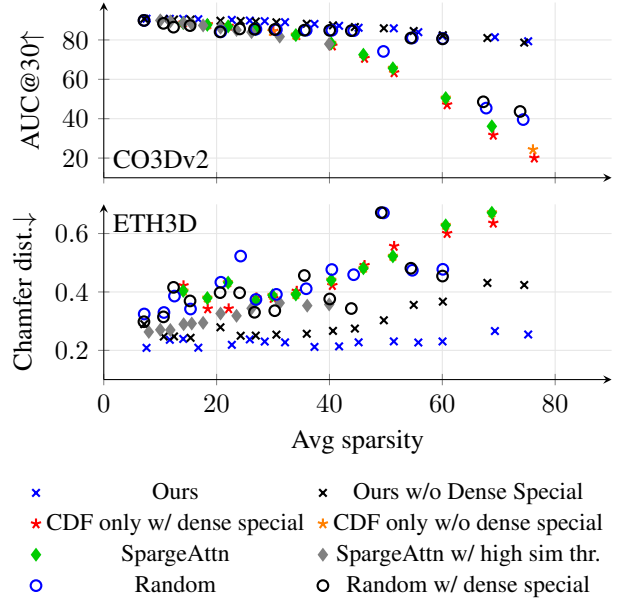


Figure 10. **Baseline sparse attention comparison.** All shown experiments do not employ SpargeAttention’s hilbert permutation. See the appendix for a more comprehensive comparison.

6. Discussion

We analyzed global attention in transformer-based geometry estimators, VGGT and π^3 , and found that it exhibits unstructured sparsity patterns, which can be interpreted as exhaustive correspondence search and is most pronounced in the middle aggregator layers. Building on these observations, we adapted a block-sparse global attention mechanism for patch-patch interactions, improving scalability to large image sets. Our approach achieves task performance comparable to the original model while being more than three times faster during inference on large-scale scenes.

While we focused on increasing the number of input images in this study, our approach is also applicable to speeding up inference on higher-resolution inputs. Doubling the input resolution results in four times as many patch tokens and in turn to $16\times$ more compute spent on global attention, drastically exacerbating the existing bottleneck.

Exploiting block-wise sparsity is orthogonal to other acceleration approaches like FlashAttention [8, 9, 34], and can potentially be integrated into the training procedure to reduce the impact on task performance.

Acknowledgments. This research was supported, in parts, by BMFTR project bridgingAI (16DHBKI023), by the Bosch-RWTH Lighthouse Collaboration “Context Understanding for Autonomous Systems”, and by the Robotics Institute Germany (RIG). Computations were performed with computing resources granted by RWTH Aachen University under projects `thes2009` and `rwth1849`. The authors thank Karim Knaebel and Kadir Yilmaz for helpful feedback and Laura Schneider for help with visualizations.

References

- [1] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *CVPR*, 2022. 6, 7
- [2] Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020. 5
- [3] Johann Cabon, Lucas Stöfl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *CVPR*, 2025. 2
- [4] Beidi Chen, Tri Dao, Eric Winsor, Zhao Song, Atri Rudra, and Christopher Ré. Scatterbrain: Unifying sparse and low-rank attention approximation. *NeurIPS*, 34:17413–17426, 2021. 3
- [5] Beidi Chen, Tri Dao, Kaizhao Liang, Jiaming Yang, Zhao Song, Atri Rudra, and Christopher Re. Pixelated butterfly: Simple and efficient sparse training for neural network models. In *ICLR*, 2022. 3
- [6] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019. 5
- [7] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 6, 7, 8
- [8] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. 1, 4, 8
- [9] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *NeurIPS*, 2022. 3, 4, 5, 8
- [10] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers, 2023. 3, 6
- [11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 3
- [12] Bardienus Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Johann Cabon, and Jerome Revaud. Mast3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. *arXiv preprint arXiv:2409.19152*, 2024. 2
- [13] Sven Elflein, Qunjie Zhou, and Laura Leal-Taixé. Light3r-sfm: Towards feed-forward structure-from-motion. In *CVPR*, 2025. 2
- [14] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *ICML*, pages 10323–10337. PMLR, 2023. 3
- [15] Yizhao Gao, Zhichen Zeng, Dayou Du, Shijie Cao, Peiyuan Zhou, Jiaxing Qi, Junjie Lai, Hayden Kwok-Hay So, Ting Cao, Fan Yang, et al. Seerattention: Learning intrinsic sparse attention in your llms. *arXiv preprint arXiv:2410.13276*, 2024. 2, 3, 8
- [16] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 2
- [17] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *CVPR*, pages 6185–6194, 2023. 3
- [18] Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jerome Revaud. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. In *CVPR*, 2025. 2
- [19] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014. 6, 7
- [20] Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir H Abdi, Dongsheng Li, Chin-Yew Lin, et al. Minference 1.0: Accelerating pre-filling for long-context llms via dynamic sparse attention. *NeurIPS*, 37:52481–52515, 2024. 3
- [21] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. *arXiv preprint arXiv:2509.13414*, 2025. 2, 3, 6, 7
- [22] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM TOG*, 2017. 6, 8
- [23] Xunhao Lai, Jianqiao Lu, Yao Luo, Yiyuan Ma, and Xun Zhou. Flexprefill: A context-aware sparse attention mechanism for efficient long-sequence inference. *arXiv preprint arXiv:2502.20766*, 2025. 3
- [24] Vincent Leroy, Johann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *ECCV*, 2024. 1, 2
- [25] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 3
- [26] Enzhe Lu, Zhejun Jiang, Jingyuan Liu, Yulun Du, Tao Jiang, Chao Hong, Shaowei Liu, Weiran He, Enming Yuan, Yuzhi Wang, et al. Moba: Mixture of block attention for long-context llms. *arXiv preprint arXiv:2502.13189*, 2025. 3
- [27] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3
- [28] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *ECCV*, 2024. 1, 2
- [29] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *CVPR*, 2021. 2, 3
- [30] Jeremy Reizenstein, Roman Shapovalov, Philipp Hertzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, 2021. 6, 7
- [31] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 2, 3

- [32] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 2, 3
- [33] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. 6, 7
- [34] Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *NeurIPS*, 2024. 8
- [35] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, 2013. 6, 7
- [36] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 6, 7
- [37] Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023. 3
- [38] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *CVPR*, 2025. 1
- [39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 4
- [40] Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. In *3DV*, 2024. 2
- [41] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vgg: Visual geometry grounded transformer. In *CVPR*, 2025. 1, 2, 3, 6, 7
- [42] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025. 2, 7
- [43] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 1, 2
- [44] 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*, 2025. 2, 3, 4, 6, 7
- [45] 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 *CVPR*, 2025. 2, 7
- [46] Yifan Yang, Kai Zhen, Bhavana Ganesh, Aram Galstyan, Goeric Huybrechts, Markus Müller, Jonas M Kübler, Rupa Vignesh Swaminathan, Athanasios Mouchtaris, Sravan Babu Bodapati, et al. Wanda++: Pruning large language models via regional gradients. *arXiv preprint arXiv:2503.04992*, 2025. 3
- [47] Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Yuxing Wei, Lean Wang, Zhiping Xiao, et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23078–23097, 2025. 3
- [48] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. In *ICLR*, 2025. 7
- [49] Jintao Zhang, Chendong Xiang, Haofeng Huang, Haocheng Xi, Jun Zhu, Jianfei Chen, et al. Spargeattention: Accurate and training-free sparse attention accelerating any model inference. In *ICML*, 2025. 3, 4, 6, 8
- [50] Peiyuan Zhang, Yongqi Chen, Haofeng Huang, Will Lin, Zhengzhong Liu, Ion Stoica, Eric P Xing, and Hao Zhang. Faster video diffusion with trainable sparse attention. In *NeurIPS*, 2025. 3
- [51] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *CVPR*, 2025. 4, 7
- [52] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM TOG*, 2018. 6, 7