VGGT4D: Mining Motion Cues in Visual Geometry Transformers for 4D Scene Reconstruction

Supplementary Material

6. Empirical Analysis of Attention Layers

We analyze how the decoder layers of VGGT encode dynamic content. Our findings motivate the specific design choices in our dynamic cue extraction module.

6.1. Gram Similarity vs. Standard Attention

We first compare standard attention maps with our proposed Gram similarity statistics. As shown in Figs. 7 and 8, we visualize the statistics (mean and variance) for each decoder layer. Here, *ref* denotes the reference frame and *src* denotes a source frame sampled from the temporal window.

Standard Attention ($QK^\top$). Inspecting the $Q_{\text{ref}}K_{\text{src}}$ column reveals that standard attention is dominated by semantic activations. Motion cues are barely visible. This also explains why applying Easi3R to VGGT fails: the strong semantic bias in $QK^\top$ washes out dynamic motion signals.

Gram Similarity ($QQ^\top$ and $KK^\top$). In contrast, the Gram similarities (columns $Q_{\text{ref}}Q_{\text{src}}$ and $K_{\text{ref}}K_{\text{src}}$) make physically dynamic regions salient. This confirms that while motion cues are suppressed in standard attention, they are preserved in the self-similarity of queries and keys.

6.2. Layer-wise Dynamic Cues

Based on the Gram similarity patterns, we identify three distinct regimes across the decoder layers:

- **Shallow layers (e.g., layer 1).** In the $K_{\text{ref}}K_{\text{src}}$ column, the model shows a strong semantic bias. Foreground objects stand out clearly from the background, regardless of their motion state.
- **Middle layers (e.g., layers 4–8).** As seen in the $Q_{\text{ref}}Q_{\text{src}}$ column, VGGT begins to encode motion variability. The Gram similarities computed from Q vectors over the temporal window sharpen the contrast between truly dynamic regions and the static background.
- **Deep layers (e.g., layers 18–22).** In the $Q_{\text{ref}}Q_{\text{src}}$ column, spatial priors dominate. This suppresses noisy responses from earlier layers, resulting in sharper and more stable boundaries.

6.3. Ineffectiveness of Camera Token Attention

A potential alternative to our method is relying on the camera token to identify dynamic regions. We test this hypothesis in Fig. 9. While the camera token focuses on dynamic regions in shallow layers (Layer 1), it fails to suppress them in deep layers (Layer 18). The actor's body receives attention comparable to the static background. This indicates that deep camera-token attention is unreliable for dynamic disentanglement, justifying our explicit mining of Gram similarity cues.

6.4. Explanation of the Design

Hypothesis on gram similarity. We hypothesize that the Gram similarity ($QQ^\top$ or $KK^\top$) serves as a superior amplifier for dynamic cues compared to the standard cross-attention ($QK^\top$). The standard attention mechanism computes interactions between Query and Key vectors, which originate from distinct projection heads to perform semantic alignment. The inherent distributional gap between these heterogeneous vectors tends to overshadow the subtle feature variations induced by object motion, rendering dynamic signals nearly invisible. In contrast, the Gram similarity operates on vectors within the same latent distribution. Without the interference of cross-projection discrepancies, the feature bias introduced by dynamics becomes the dominant source of variance. Consequently, the Gram operation effectively magnifies these intra-distributional differences, making dynamic regions significantly more salient than the static background.

Why camera token is unreliable? One might expect the global camera token in deep layers to strictly attend only to static regions for robust pose estimation. However, we observe that Transformer-based foundation models achieve robustness through soft aggregation rather than hard exclusion. The model learns to tolerate a certain degree of dynamic noise (outliers) within its massive context window to solve the global pose optimization. As a result, the camera token does not explicitly 'zero out' dynamic regions but rather down-weights them subtly or mixes them into the context. This leads to ambiguous attention maps that are insufficient for generating precise, binary dynamic masks.

7. Implementation Details

7.1. Hyperparameters

We provide the specific hyperparameters used in our pipeline in Tab. 7.

7.2. Dynamic Cue Extraction & Memory Optimization

As detailed in Sec. 6, we combine cues from three layer groups. We extract w_{shallow} from Layer 1 for strong semantics. We extract w_{middle} from Layers 4–8 to capture motion variance. Finally, we extract w_{deep} from Layers 18–22 to utilize spatial priors for outlier suppression.

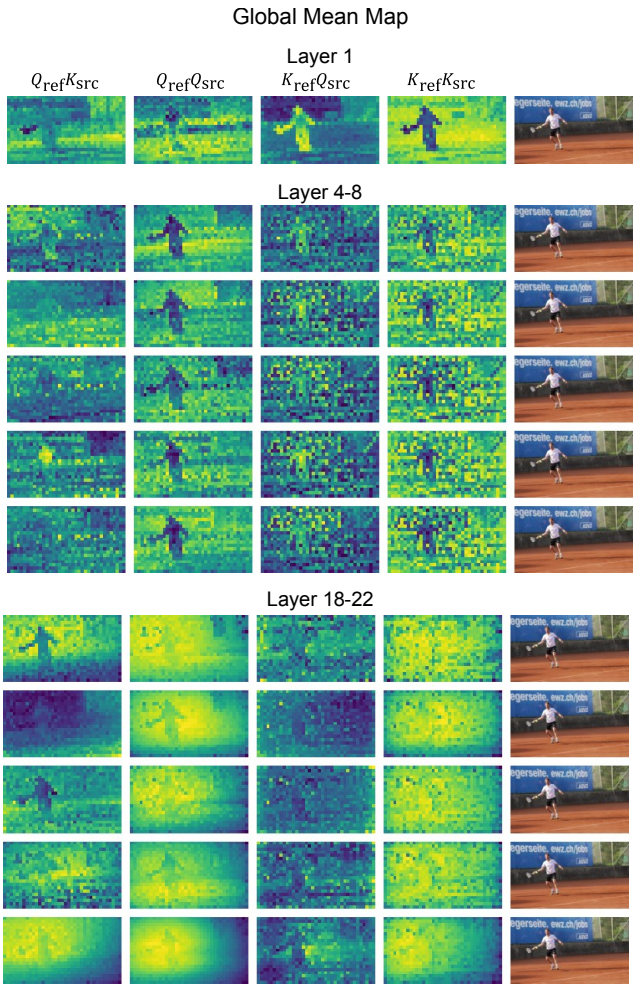

Figure 7. **Mean Gram similarity across decoder layers.** We visualize Gram similarity maps averaged over a temporal window. The $Q_{\text{ref}}K_{\text{src}}$ column shows the standard attention map ($QK^\top$). Note that motion cues are faint in standard attention but distinct in Gram similarities ($QQ^\top, KK^\top$).

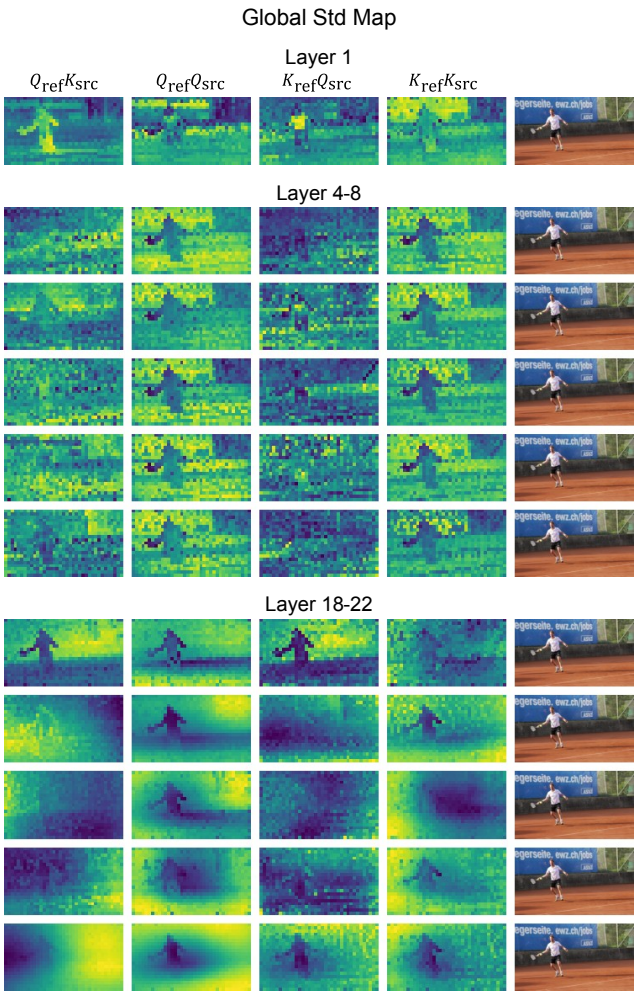

Figure 8. **Variance of Gram similarity across decoder layers.** We visualize the variance of Gram similarity. Similar to the mean statistics, Gram similarities capture motion variance more effectively than standard attention ($Q_{\text{ref}}K_{\text{src}}$).

Parameter	Value
Temporal Window	6 source frames (stride 2)
Layers for w_{shallow}	Layer 1
Layers for w_{middle}	Layers 4–8
Layers for w_{deep}	Layers 19–20 (Var), 18–22 (Mean)
Early-stage Masking Layers	Layers 1–5
SOR Neighbors (k)	20
SOR Std. Dev. Mult. (σ)	2.5

Table 7. **Hyperparameter settings.** Key parameters for cue extraction and refinement.

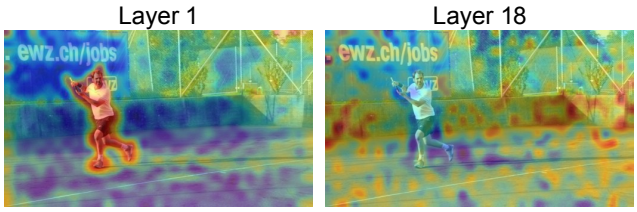

Figure 9. **Camera-to-image attention limitations.** We visualize attention from the camera token to image tokens. In Layer 18, the camera token fails to suppress the moving actor, treating it similarly to the static background. This confirms that camera tokens are insufficient for robust dynamic masking.

To define the threshold α, which is used in Sec. 3.3 to obtain dynamic mask, we use VGGT's ViT backbone to extract image features and apply k-means clustering to group

tokens across frames. We verify the dynamic score of each cluster and use Otsu's algorithm to determine the optimal separation threshold.

Memory Optimization. VGGT uses PyTorch's Scaled Dot Product Attention (SDPA) for efficiency. Explicitly computing the full Gram matrix $QK^\top$ scales quadratically with token count ($O(N^2)$), which would cause Out-Of-Memory (OOM) errors given the thousands of tokens in multi-view inputs. To resolve this, we compute per-frame Gram similarities in-place within the attention layer. This avoids materializing the massive full matrix and keeps memory usage linear with respect to the number of frames.

7.3. Mask Refinement

Raw masks from 3D foundation models often contain outliers ("floaters"). These artifacts degrade the projection-based refinement. We first apply Statistical Outlier Removal (SOR) to the point cloud to filter noise. Next, we cluster the remaining points and average the projection gradients within each cluster. This aggregation stabilizes the gradient signal, preventing isolated outliers from triggering false positives in the dynamic classification.

7.4. Early-Stage Masking

During inference, we apply the dynamic masks only to layers 1–5. We suppress the key (K) vectors of dynamic tokens in these layers. This prevents query (Q) vectors from attending to dynamic regions early in the network, preserving the geometric consistency of deeper layers.

8. Additional Experiments

8.1. Ablation Study on Dynamic Map Components

We validate the contribution of each term in Eq. (5) (Main Paper). As shown in Tab. 8, w_{shallow} and w_{middle} provide the primary motion signals. However, performance drops significantly without w_{deep}, confirming its role in suppressing residual outliers.

Method	DAVIS-2016			
	JM↑	JR↑	FM↑	FR↑
w/o w_{shallow}	54.15	62.44	46.43	44.27
w/o w_{middle}	56.13	57.12	44.07	41.90
w/o w_{deep}	46.85	48.89	41.52	45.30
w/o refinement	<u>59.74</u>	<u>73.10</u>	<u>50.64</u>	<u>58.30</u>
Ours	**62.12**	**76.80**	**56.04**	**67.49**

Table 8. **Ablation on dynamic mask estimation.** We evaluate the contribution of each component. Note that the "w/o" variants are evaluated before the refinement stage to isolate the impact of the cue extraction.

8.2. Zero-shot vs. Trained 2D Segmentation

We compare our method against FlowSAM, a strong 2D video segmentation baseline. FlowSAM lacks 3D spatial reasoning and typically requires ground-truth supervision (Hungarian matching) for evaluation. For a fair comparison, we adapt FlowSAM to a zero-shot setting.

As shown in Tab. 9, our training-free approach outperforms the trained FlowSAM baseline. This demonstrates that leveraging the implicit 3D/4D priors in VGGT yields better temporal consistency than pure 2D video analysis.

Method	DAVIS-2016			
	JM↑	JR↑	FM↑	FR↑
FlowSAM (zero-shot)	54.53	56.86	<u>52.48</u>	<u>52.97</u>
DAS3R	41.13	38.67	44.50	36.94
Easi3R$_{\text{dust3r}}$	50.10	55.77	43.40	37.25
Easi3R$_{\text{monst3r}}$	<u>54.93</u>	<u>68.00</u>	45.29	47.30
Ours	**62.12**	**76.80**	**56.04**	**67.49**

Table 9. **Dynamic object segmentation comparison.** Our method outperforms the trained 2D baseline FlowSAM in a zero-shot setting.

9. Limitations

While our method achieves robust 4D reconstruction without training, it has limitations. First, computing Gram similarities adds computational overhead compared to single-pass inference. Second, our mask refinement depends on the quality of the initial depth estimates from VGGT. If the backbone misestimates depth (e.g., blending foreground and background), the projection gradients become unreliable. Finally, our refinement assumes rigid motion for projection checks, which may struggle with highly non-rigid or fluid deformations. Future work will focus on optimizing efficiency and handling non-rigid dynamics.