TTT4D: MINING MOTION CUES IN TEST-TIME TRAINING FOR SCALABLE 4D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

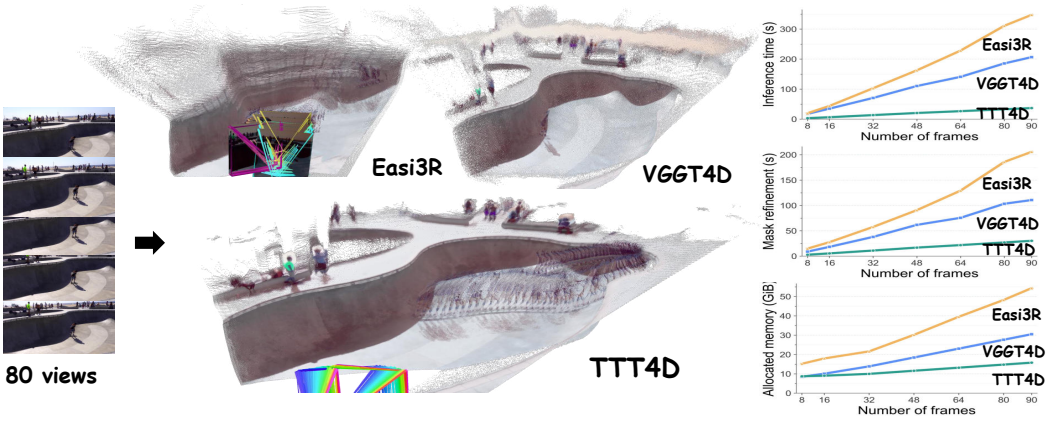


Figure 1: **TTT4D** is a training-free framework that adapts the pretrained 3D foundation model ZipMap to dynamic 4D reconstruction through test-time training. We demonstrate the potential of stateful feed-forward 3D models with linear-time complexity $\mathcal{O}(N)$ for dynamic 4D reconstruction. On the right, we compare different models in terms of inference time, mask update time, and allocated GPU memory with increasing numbers of input frames on the TUM-Dynamics sequence.

ABSTRACT

Reconstructing dynamic 4D scenes from monocular videos remains challenging because moving objects violate the static scene assumptions of most 3D foundation models and interfere with camera and geometry estimation. Existing 4D methods often rely on fine tuning with limited dynamic data or external priors, while recent training-free approaches (e.g., Easi3R and VGGT4D) remain constrained by limited temporal scalability. In this work, we introduce TTT4D, a training-free framework that adapts the stateful 3D foundation model ZipMap to dynamic 4D reconstruction through test-time training. We observe that independent motion manifests as complementary inconsistencies in both ZipMap's internal representations and its online adaptation dynamics. Individual signals, however, are not motion specific and may also respond to static ambiguities such as newly observed regions, rich appearance, or uncertain geometry. Crucially, these ambiguities affect different internal signals differently, whereas independently moving regions exhibit consistent cross cue responses. Based on this observation, we extract complementary adaptation- and representation-level cues and fuse them to suppress ambiguous static responses and highlight coherent dynamic regions. The resulting masks guide a shallow attention routing strategy that selectively isolates dynamic features to reduce dynamic interference without retraining the pretrained model. TTT4D achieves strong performance in dynamic segmentation, camera pose estimation, and dense 4D reconstruction while retaining the efficiency and scalability of stateful feed-forward architectures, with $5.85\times$ /$6.87\times$ speedups over VGGT-4D and $10.29\times$ /$8.27\times$ over Easi3R on TUM-Dynamics / ADT benchmarks, respectively, using a single NVIDIA RTX PRO 6000 (96GB) GPU. Code will be released to contribute to the community.

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.

1 INTRODUCTION

Recovering camera motion, scene geometry, and object dynamics from dynamic videos is a fundamental challenge in 3D vision and 4D scene understanding. Such capability is essential for applications including novel view synthesis, autonomous navigation, and embodied intelligence. Classical Structure-from-Motion (SfM) (Schonberger & Frahm, 2016; Pan et al., 2024) and Multi-View Stereo (MVS) (Furukawa & Ponce, 2009; Schönberger et al., 2016) methods rely on static-scene assumptions and therefore degrade in dynamic environments, where independently moving objects interfere with camera motion estimation and scene geometry reconstruction.

Recent advances in 3D foundation models have significantly improved feed-forward reconstruction. Attention-based models such as DUSt3R (Wang et al., 2024) and VGGT (Wang et al., 2025a) learn strong geometric priors from large-scale data, while the stateful feed-forward model ZipMap (Jin et al., 2026) further enables efficient long-sequence reconstruction through test-time training. Nevertheless, these models are primarily trained under static-world assumptions and lack explicit mechanisms to separate dynamic objects from underlying scene.

Existing approaches to dynamic 4D reconstruction mainly follow two directions. The first direction adapts static 3D models to dynamic environments through additional training (Xu et al., 2024; Zhang et al., 2025a; Wang et al., 2025b) or optimization (Li et al., 2025; Kopf et al., 2021). However, these approaches require costly dynamic supervision and may suffer from domain gaps introduced by external components. The second direction explores training-free adaptation by exploiting internal representations of pretrained models. Easi3R (Chen et al., 2025a) discovers motion-related signals from pairwise cross-attention statistics in DUSt3R and performs an additional inference stage to suppress dynamic interference. VGGT4D (Hu et al., 2026) further extends this idea by analyzing multi-frame global attention patterns in VGGT to obtain temporally consistent dynamic masks. Despite these advances, existing training-free approaches remain limited by their reliance on explicit attention analysis. Pairwise attention mechanisms provide only local temporal interactions and struggle with complex long-term motion, while global attention architectures ($\mathcal{O}(N^2)$) introduce rapidly increasing memory and computational costs as the sequence length grows. More importantly, current approaches focus on static attention responses, leaving an important question unexplored: *Can the online adaptation dynamics of a stateful 3D foundation model itself reveal motion information without additional training or external motion priors?*

We investigate whether these dynamics can reveal motion without additional training or external motion priors. We study TTT dynamics through ZipMap, a stateful feed-forward reconstruction model based on Test-Time Training (TTT) (Zhang et al., 2025b; Sun et al., 2024). Instead of explicit key-value caches, ZipMap compresses an image sequence into compact TTT fast weights that each token updates and queries during inference. Our key observation is that independent motion manifests as complementary inconsistencies in this stateful process. At the adaptation level, they are less compatible with the accumulated scene state and tend to induce larger fast-weight readout mismatches and stronger adaptation responses. At the representation level, independently moving patches tend to deviate from the dominant frame-level geometric context in query space. Importantly, neither response is motion-specific: state incompatibility may also arise from novel or difficult static observations, while representation outliers may correspond to salient but static regions. Their distinct failure modes, however, make jointly supported regions more reliable indicators of independent motion. Based on this observation, we propose TTT4D, a training-free framework that extracts two principal cues from stateful inference: a TTT write-aware proxy for adaptation demand and a frame-centered query residual for representation deviation. We further complement them with gate-update magnitude and value-channel entropy to reinforce jointly supported regions and suppress cue-specific ambiguities. The resulting scores are refined into spatially coherent masks that guide a shallow hard-to-soft attention routing strategy. Early layers suppress dynamic keys and values, while later layers restore full attention and substitute static feature statistics for dynamic values. Overall, our main contributions are summarized as follows:

- We introduce TTT4D, a training-free framework that extends the stateful 3D foundation model ZipMap to dynamic 4D reconstruction through test-time training.
- We reveal complementary adaptation- and representation-level signals in stateful inference, whose joint patterns help separate independent motion from ambiguous static responses.

- We design a dynamic-aware inference strategy that performs shallow hard-to-soft attention routing with selective dynamic feature isolation, effectively reducing dynamic interference for more accurate camera pose estimation and 4D scene reconstruction.

2 Related Work

2.1 3D Foundation Models

DUSt3R (Wang et al., 2024) pioneered feed-forward 3D foundation models by predicting dense point maps from image pairs. Building on this paradigm, MASt3R (Leroy et al., 2024) improves correspondence estimation through explicit 3D-aware matching. Recent multi-view extensions further enhance cross-view reasoning by incorporating memory mechanisms, as in CUT3R (Wang et al., 2025b) and MUSt3R (Cabon et al., 2025), or global attention, as exemplified by Fast3R (Yang et al., 2025), VGGT (Wang et al., 2025a), and π^3 (Wang et al., 2025c). While global Transformers enable effective cross-view reasoning, their growing token representations and attention costs limit scalability. TTT (Sun et al., 2024) addresses this bottleneck by compressing visual context into compact adaptive states with linear sequence scaling. TTT3R (Chen et al., 2025b) formulates recurrent reconstruction as an online adaptation process, while VGG-T^3 (Elflein et al., 2026) compresses VGGT's growing key–value representation into fixed-size fast weights. Similarly, ZipMap (Jin et al., 2026) encodes image sequences into compact test-time-trained states that are updated and queried online. These stateful architectures provide the foundation for our method.

2.2 4D Scene Reconstruction

Classical approaches for dynamic scenes often depend on RGB-D inputs (Innmann et al., 2016; Newcombe et al., 2015) or simultaneously recover depth, pose, and object motion (Gordon et al., 2019; Zhang et al., 2022; Li et al., 2025; Kopf et al., 2021). Despite recent progress, these methods remain limited in scalability due to their reliance on external priors. A growing body of work therefore shifts toward learned representations and lighter adaptation strategies. MonST3R (Zhang et al., 2025a) adapts pretrained models on dynamic data with optical-flow guidance; DAS3R (Xu et al., 2024) attaches a lightweight head for direct motion-mask prediction; CUT3R (Wang et al., 2025b) retrains on mixed static-dynamic collections to support fast feed-forward reconstruction; Easi3R (Chen et al., 2025a) performs inference-time attention modulation for training-free 4D recovery from DUSt3R (Wang et al., 2024); and VGGT4D (Hu et al., 2026) extracts temporally coherent dynamic masks from multi-frame attention statistics. Nevertheless, the majority of these solutions still involve either fine-tuning or auxiliary post-processing, and existing training-free alternatives are constrained by the restricted temporal scope of pairwise attention or the rapidly growing cost of global attention. In this work, we investigate whether a stateful 3D foundation model's adaptation dynamics can reveal motion cues for training-free dynamic scene separation without priors.

3 Method

3.1 Preliminary

Large-Chunk Test-Time Training. Large-chunk Test-Time Training (LaCT) (Zhang et al., 2025b) provides a stateful mechanism for compressing token interactions into adaptive fast weights. We follow the formulation of LaCT and consider a single TTT block (Sun et al., 2024), omitting the layer index for clarity. Given M input tokens, let $\mathbf{q}_i$, $\mathbf{k}_i$, and $\mathbf{v}_i$ denote their query, key, and value representations. LaCT writes the key–value associations of the entire token chunk into a fixed-size fast-weight memory and reads it with the queries:

$$\begin{aligned} \mathcal{L}_i &= -f_{\mathbf{W}}(\mathbf{k}_i)^\top \mathbf{v}_i, \\ \widehat{\mathbf{W}} &= \texttt{Write}\left(\mathbf{W}; \nabla_{\mathbf{W}} \sum_{i=1}^{M} \eta_i \mathcal{L}_i\right), \\ \mathbf{o}_i &= \texttt{Read}\left(\widehat{\mathbf{W}}, \mathbf{q}_i\right) = f_{\widehat{\mathbf{W}}}(\mathbf{q}_i). \end{aligned} \tag{1}$$

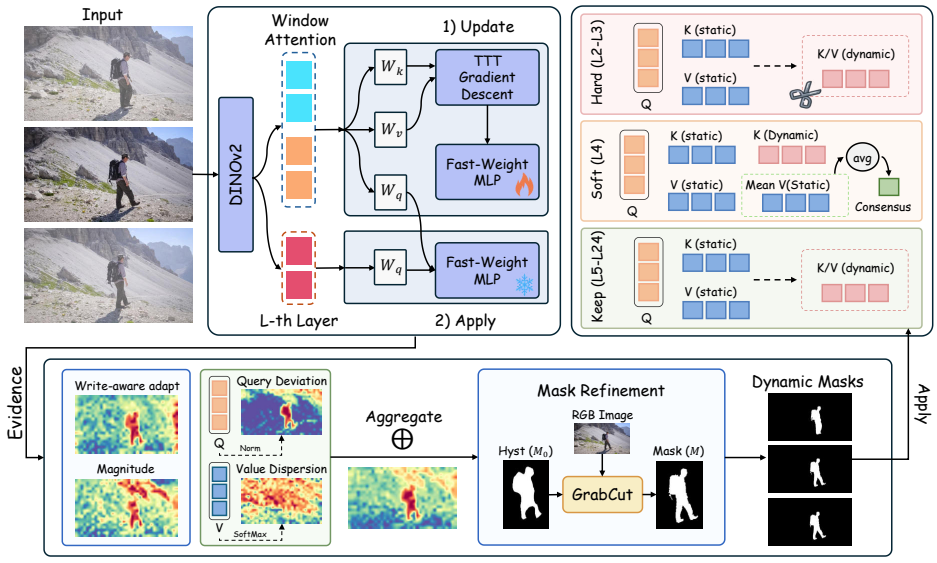


Figure 2: **Overview of TTT4D.** An input image sequence is processed by the stateful 3D foundation model ZipMap. We extract and fuse complementary adaptation- and representation-level cues from TTT inference to suppress ambiguous static responses and highlight independently moving regions. The resulting masks guide hard-to-soft attention routing strategy, enabling robust pose estimation and dynamic 4D reconstruction without additional training or motion priors.

Here, η_i is the input-dependent learning rate, and $f_{\mathbf{W}}$ is a SwiGLU-MLP (Shazeer, 2020) based fast-weight function defined as $f_{\mathbf{W}}(\mathbf{k}_i) = \mathbf{W}_1(\mathrm{SiLU}(\mathbf{k}_i\mathbf{W}_0) \odot (\mathbf{k}_i\mathbf{W}_2))$, where $\mathbf{W} = \{\mathbf{W}_0, \mathbf{W}_1, \mathbf{W}_2\}$. By compressing M key–value associations into fixed-size fast weights rather than explicitly modeling pairwise token interactions, TTT mechanism maintains linear computational complexity with respect to the input length, enabling processing of long token sequences.

ZipMap. ZipMap (Jin et al., 2026) is a 3D foundation model that instantiates TTT for feed-forward multi-view reconstruction. Given N input images $\mathbf{I} = \{\mathbf{I}_t\}_{t=1}^N$, it interleaves local window attention with L LaCT-style TTT blocks for linear-time multi-view aggregation. Each TTT block maintains an independent fast-weight memory. For block l, we denote its pre- and post-write states by $\mathbf{W}^{(l)}$ and $\widehat{\mathbf{W}}^{(l)}$, respectively; hence, $\mathbf{W}$ and $\widehat{\mathbf{W}}$ in Eq. equation 1 always refer to the within-block states before and after the current write. Through these interleaved local and global aggregation operations, ZipMap builds an implicit multi-view scene representation, which is decoded into $\mathbf{C} = \{\mathbf{c}_t\}_{t=1}^N$, $\mathbf{D} = \{\mathbf{D}_t\}_{t=1}^N$, and $\mathbf{P} = \{\mathbf{P}_t\}_{t=1}^N$, where $(\mathbf{c}_t, \mathbf{D}_t, \mathbf{P}_t)$ denote the camera parameters, depth map, and 3D point map associated with $\mathbf{I}_t$.

3.2 Motion Evidence from Stateful Inference

Although ZipMap is pretrained primarily on static scenes, independent motion can perturb several aspects of its stateful inference. Moving regions tend to deviate from the dominant frame-level representation and be less compatible with the current fast-weight state, yielding complementary adaptation- and representation-level signals. As shown in Figure 3, individual cues may also respond to static ambiguities such as rich texture, novel observations, or uncertain geometry. However, these ambiguities manifest differently across cues, allowing their fusion to suppress isolated responses and highlight independent motion.

Write-aware adaptation demand. Motivated by the write operation in Eq. equation 1, we measure how strongly each observation pressures the incoming fast-weight state. Intuitively, a static observation that is compatible with the shared scene representation should be well explained by the current key–value association encoded in the fast weights. In contrast, an independently moving ob-

servation is more likely to violate this association and thus demand stronger adaptation when written into the state. For token (t, j), we define its write-aware adaptation demand as

$$u_{t,j} = \eta_{t,j} \left\| \mathbf{v}_{t,j} - f_{\mathbf{W}}(\mathbf{k}_{t,j}) \right\|_2 \left\| \mathbf{k}_{t,j} \right\|_2 . \tag{2}$$

where $\eta_{t,j}$ is the input-dependent learning rate defined in Eq. equation 1. The discrepancy $\|\mathbf{v}_{t,j} - f_{\mathbf{W}}(\mathbf{k}_{t,j})\|_2$ measures how well the incoming fast-weight state explains the observed key–value association, while $\eta_{t,j}\|\mathbf{k}_{t,j}\|_2$ characterizes its effective write sensitivity. Their product therefore provides a token-wise proxy for the adaptation pressure exerted on the scene state. As visualized in Figure 3, $u_{t,j}$ provides broad sensitivity to dynamic regions, but also responds to newly observed or disoccluded areas and difficult static backgrounds.

Frame-centered query deviation. By analyzing the query representations within the TTT blocks, we observe that background regions often form a dominant frame-level context, whereas salient foreground regions are more likely to produce outlying query directions. We first normalize each patch query and compute the frame-level query centre.

$$\bar{q}_t = \frac{\frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} q_{t,j}}{\left\| \frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} q_{t,j} \right\|_2}. \tag{3}$$

We quantify this deviation as the cosine distance to the frame-level query centre:

$$r_{t,j} = 1 - q_{t,j}^T \bar{q}_t. \tag{4}$$

As shown in Figure 3, $r_{t,j}$ suppresses many background responses present in $u_{t,j}$ and produces more coherent foreground localization. However, it measures frame-relative representation deviation and cannot by itself determine whether an outlier is dynamic or static.

Gate-update magnitude. While Eq. (2) estimates this demand from the mismatch objective, we additionally characterize the actual update response of the TTT gating pathway. Specifically, we define the magnitude $m_{t,j}$ to quantify each token's update contribution to the matrix W_0:

$$m_{t,j} = \left\| \Delta W_{0,t,j} \right\|_F . \tag{5}$$

Larger $m_{t,j}$ indicates a stronger actual adaptation response. As shown in Figure 3, this cue is typically sparser than $u_{t,j}$ and $r_{t,j}$, but provides localized support within candidate moving regions.

Value-channel dispersion. Besides geometric inconsistency, some independently moving regions exhibit less stable local representations, resulting in more diffuse value-channel responses. For the value vector $v \in \mathbb{R}^d$, we convert absolute channel activations into a distribution:

$$p_{t,j} = \mathrm{softmax}\left(v_{t,j}\right). \tag{6}$$

We then compute the normalized channel entropy:

$$e_{t,j} = -\frac{1}{\log d} \sum_{c=1}^{d} p_{t,j,c} \log p_{t,j,c}. \tag{7}$$

The resulting entropy $e_{t,j}$ measures value-channel dispersion. As shown in Figure 3, it is relatively diffuse and also responds to ambiguous static regions; we therefore use it only as auxiliary representation-ambiguity evidence rather than a standalone localization cue. Together, the cues play asymmetric roles: $u_{t,j}$ captures observations poorly accommodated by the current fast-weight memory, $r_{t,j}$ provides more coherent frame-relative localization, $m_{t,j}$ supplies adaptation-side support, and $e_{t,j}$ contributes auxiliary ambiguity evidence. Their distinct failure modes make jointly supported regions more reliable than isolated cue responses.

3.3 Motion-Guided Geometric Inference

$$S = \widetilde{u}_{t,j} + \widetilde{r}_{t,j} + \widetilde{m}_{t,j} + \widetilde{e}_{t,j}. \tag{8}$$

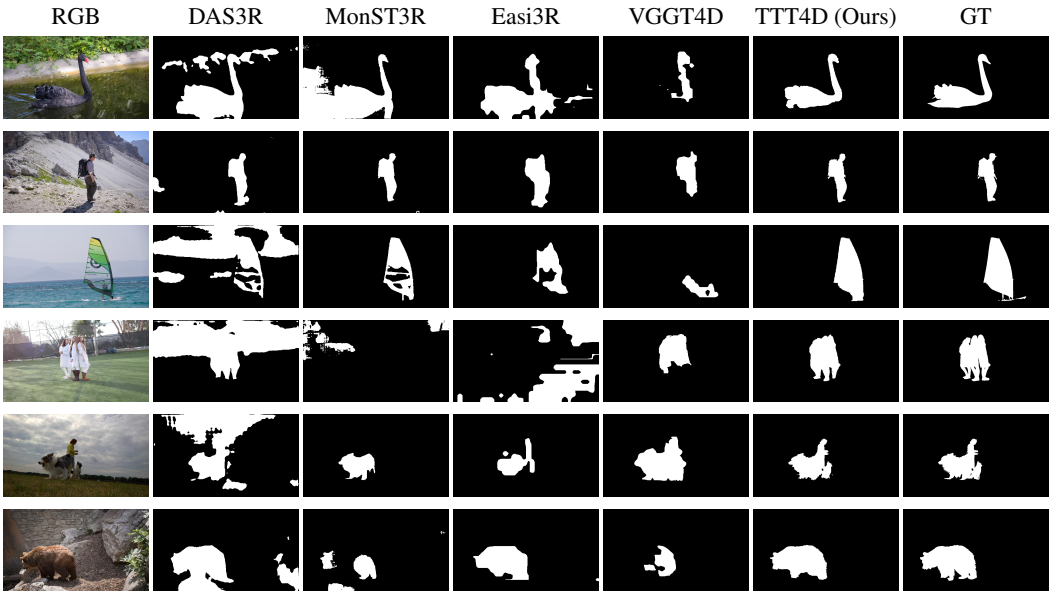


Figure 4: **Qualitative comparison of dynamic object segmentation.** TTT4D produces more accurate and coherent dynamic masks, with finer boundary preservation and better separation between dynamic objects and static backgrounds.

All cues are normalized and summed into the final dynamic score. Their distinct failure modes make jointly supported regions more reliable than isolated responses.

Given the fused score map S_t, we apply connectivity-aware hysteresis thresholding to obtain spatially coherent dynamic regions:

$$M_t^0 = \mathrm{Hyst}\left(S_t\right). \tag{9}$$

where $\mathrm{Hyst}(\cdot)$ retains low-confidence responses only when they are connected to high-confidence dynamic regions (Canny, 1986). Since the score map is computed from patch-level representations, its boundaries may not precisely follow object contours. We therefore use the coarse mask to initialize a image-guided GrabCut (Rother et al., 2004) refinement:

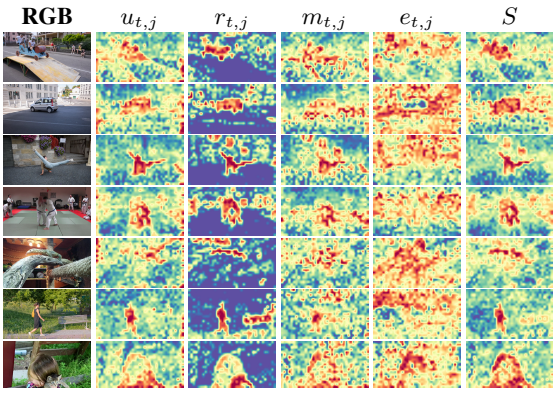


Figure 3: **Visualization of the motion cues.** From left to right: RGB, $u_{t,j}$, $r_{t,j}$, $m_{t,j}$, $e_{t,j}$ and S. The cues exhibit distinct spatial responses, whose fusion reinforces jointly supported dynamic regions.

$$M_t = \mathrm{GrabCut}\left(I_t, M_t^0\right). \tag{10}$$

where I_t is the corresponding RGB frame. This step adjusts the coarse prediction to local appearance boundaries without introducing additional motion cues. Compared with the post-processing strategies of Easi3R (Chen et al., 2025a) and VGGT4D (Hu et al., 2026), our approach is simpler and more efficient. Using the predicted dynamic mask, we modify shallow attention to reduce dynamic interference. In the first stage, dynamic keys and values are removed from patch attention to prevent dynamic interference during geometric aggregation. In the second stage, the full key topology is restored, while dynamic values are replaced by the mean of static values to preserve contextual reasoning. The modification is performed only at inference time without retraining the QKV projections. Although mask-guided routing substantially reduces dynamic interference, the estimated camera trajectory may still exhibit mild frame-wise jitter. We therefore apply temporal stabilization over a short temporal neighborhood, smoothing translations in displacement space and performing gated rotation averaging on SO(3) to suppress local pose fluctuations.

Table 1: **Dynamic object segmentation on DAVIS.** Top-1, Top-2, and Top-3 results are highlighted in red, yellow, and orange, respectively. Easi3R$_{\mathrm{DUSt3R}}$ and Easi3R$_{\mathrm{MonST3R}}$ correspond to different variants used by Easi3R. These results demonstrate that TTT4D effectively reveals the inherent dynamic understanding capability of stateful feed-forward models.

Method	DAVIS-2016				DAVIS-2017				DAVIS-all			
	J_M	J_R	F_M	F_R	J_M	J_R	F_M	F_R	J_M	J_R	F_M	F_R
$\mathcal{O}(N^2)$ *Inference Speed*												
MonST3R	40.42	40.39	49.54	52.12	38.07	36.05	48.24	49.01	36.98	34.52	47.03	46.72
DAS3R	41.13	38.67	44.50	36.94	44.51	43.95	46.71	44.96	43.33	38.93	45.24	38.78
Easi3R$_{\mathrm{DUSt3R}}$	50.10	55.77	43.40	37.25	46.86	50.54	39.06	30.05	44.10	50.85	35.16	27.24
Easi3R$_{\mathrm{MonST3R}}$	54.93	68.00	45.29	47.30	54.75	66.16	44.09	42.36	51.64	63.06	40.98	38.49
VGGT4D	60.26	73.11	49.10	56.06	53.65	57.79	44.27	43.16	47.93	49.00	39.24	31.17
$\mathcal{O}(N)$ *Inference Speed*												
TTT4D (Ours)	60.58	72.86	56.51	63.76	58.32	66.91	54.20	60.06	55.98	65.16	54.04	60.11

Table 2: **Camera pose estimation on TUM-Dynamics, DyCheck, and ADT.** With linear complexity $\mathcal{O}(N)$, TTT4D surpasses quadratic-complexity $\mathcal{O}(N^2)$ baselines on TUM-Dynamics and ADT to achieve state-of-the-art performance, while further outperforming ZipMap on DyCheck.

Method	TUM-Dynamics			DyCheck			ADT		
	ATE↓	RTE↓	RRE↓	ATE↓	RTE↓	RRE↓	ATE↓	RTE↓	RRE↓
$\mathcal{O}(N^2)$ *Inference Speed*									
Easi3R$_{\mathrm{DUSt3R}}$	0.063	0.046	2.523	0.021	0.014	1.092	0.042	0.015	0.655
Easi3R$_{\mathrm{MonST3R}}$	0.133	0.120	4.366	0.030	0.021	1.390	0.049	0.028	0.684
MonST3R	0.156	0.103	12.041	0.033	0.024	1.501	0.055	0.025	0.776
DAS3R	0.155	0.102	12.038	0.033	0.022	1.467	0.040	0.017	0.685
VGGT	0.012	0.010	0.309	0.012	0.011	0.483	0.064	0.025	0.425
VGGT4D	0.010	0.009	0.300	0.011	0.010	0.420	0.065	0.025	0.424
$\mathcal{O}(N)$ *Inference Speed*									
CUT3R	0.054	0.041	5.346	0.029	0.020	1.383	0.084	0.025	0.490
ZipMap	0.014	0.010	0.312	0.019	0.018	1.009	0.033	0.017	0.337
TTT4D (Ours)	0.009	0.005	0.298	0.016	0.013	0.973	0.030	0.012	0.330

4 Experiments

We evaluate TTT4D on a range of dynamic scene understanding tasks, including dynamic object segmentation, camera pose estimation, and 4D scene reconstruction. We compare TTT4D with state-of-the-art 4D reconstruction methods, including DUSt3R (Wang et al., 2024), MonST3R (Zhang et al., 2025a), DAS3R (Xu et al., 2024), CUT3R (Wang et al., 2025b), Easi3R (Chen et al., 2025a), VGGT (Wang et al., 2025a), VGGT4D (Hu et al., 2026). Among them, MonST3R, DAS3R, and CUT3R rely on fine-tuning on dynamic-scene datasets or additional optimization with motion-related supervision. TTT4D is built upon the pretrained 3D foundation model ZipMap (Jin et al., 2026) model. Similar to Easi3R and VGGT4D, it requires no additional fine-tuning or optimization, while improving reconstruction performance by uncovering motion cues from the model's own adaptation dynamics. All experiments are conducted on a single NVIDIA RTX PRO 6000 (96GB) GPU. For more implementation details, please refer to the supplementary material.

4.1 Dynamic Object Segmentation

We evaluate TTT4D on DAVIS (Perazzi et al., 2016; Pont-Tuset et al., 2017), which provides densely annotated videos with diverse object motions and appearance variations. Following past methods (Perazzi et al., 2016; Hu et al., 2026; Chen et al., 2025a), we adopt Intersection over Union (J_M), Boundary F-measure (F_M), and their corresponding recalls (J_R and F_R) to evaluate dynamic mask estimation. The quantitative results are summarized in Table 1. As illustrated in Figure 4, DAS3R tends to over-segment dynamic regions (e.g., the sailboat, the person on the grass, and the dog), MonST3R often fails to separate dynamic objects or over-segments them due to its reliance on optical flow (e.g., the swan, the person on the grass, and the bear), Easi3R struggles to preserve fine

Table 3: **4D scene reconstruction on the DyCheck benchmark.** Although VGGT4D ($\mathcal{O}(N^2)$) performs best on DyCheck, TTT4D ($\mathcal{O}(N)$) significantly narrows the gap with an efficient stateful architecture, while consistently improving the reconstruction quality of its backbone, ZipMap.

Method	Accuracy↓		Completeness↓		Distance↓	
	Mean	Median	Mean	Median	Mean	Median
$\mathcal{O}(N^2)$ Inference Speed						
DUSt3R	0.802	0.595	1.950	0.815	0.353	0.233
MonST3R	0.851	0.689	1.734	0.958	0.353	0.254
DAS3R	1.772	1.438	2.503	1.548	0.475	0.352
Easi3R$_{\text{MonST3R}}$	0.834	0.643	1.661	0.916	0.350	0.255
Easi3R$_{\text{DUSt3R}}$	0.703	0.589	1.474	0.586	0.301	0.186
VGGT	0.015	0.008	0.100	0.025	0.260	0.136
VGGT4D	0.014	0.007	0.094	0.022	0.250	0.128
$\mathcal{O}(N)$ Inference Speed						
CUT3R	0.458	0.342	1.633	0.792	0.326	0.229
ZipMap	0.023	0.015	0.109	0.043	0.272	0.156
TTT4D (Ours)	0.020	0.013	0.101	0.037	0.264	0.146

object boundaries (e.g., the swan, the hiker, and the sailboat), and VGGT4D produces fragmented masks for dynamic objects (e.g., the swan, the sailboat, and the bear). In contrast, our method generates more accurate masks with superior boundary preservation, effectively separating dynamic objects from the static background. These results demonstrate that the intrinsic update mechanism of stateful 3D feed-forward models captures rich scene representations that naturally encode dynamic cues, enabling robust motion perception without external motion supervision.

4.2 Camera Pose Estimation

We evaluate camera pose estimation on three dynamic benchmarks: DyCheck (Gao et al., 2022), TUM-Dynamics (Sturm et al., 2012), and ADT (Koppula et al., 2024; Pan et al., 2023). Following prior works (Hu et al., 2026; Chen et al., 2025a), we report the standard metrics of Absolute Trajectory Error (ATE), Relative Translation Error (RTE), and Relative Rotation Error (RRE). As shown in Table 2, TTT4D consistently improves the pose estimation accuracy of the underlying stateful model ZipMap across all three benchmarks, demonstrating the effectiveness of the proposed motion-aware inference strategy. Notably, on both TUM-Dynamics and ADT benchmarks, TTT4D achieves the best overall performance, outperforming the $\mathcal{O}(N^2)$ baselines Easi3R and VGGT4D models in pose accuracy. By identifying moving regions through the intrinsic representation and adaptation dynamics of the pretrained model, TTT4D produces more reliable geometric correspondences for camera estimation while preserving the efficiency advantages of stateful feed-forward inference as shown in Fig. 5. TTT4D is $5.85\times$ and $6.87\times$ faster than VGGT-4D, and $10.29\times$ and $8.27\times$ faster than Easi3R on TUM-Dynamics and ADT, respectively. The supplementary material provides a detailed efficiency comparison and further evaluates different models on long-sequence inputs.

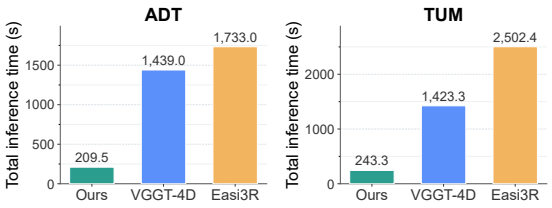


Figure 5: **Runtime comparison on ADT and TUM-Dynamics.** Total time includes mask extraction, mask update, and masked inference.

4.3 4D Scene Reconstruction

We evaluate dense 4D point cloud reconstruction on the DyCheck benchmark using the standard metrics of Accuracy, Completeness, and Distance (Wang & Agapito, 2025; Aanæs et al., 2016; Chen et al., 2025a). Here, we use the original poses for scene reconstruction. As reported in Table 3, TTT4D consistently improves the reconstruction quality of the underlying ZipMap model across all evaluation metrics. Specifically, the median Accuracy is reduced from 0.015 to 0.013, the median

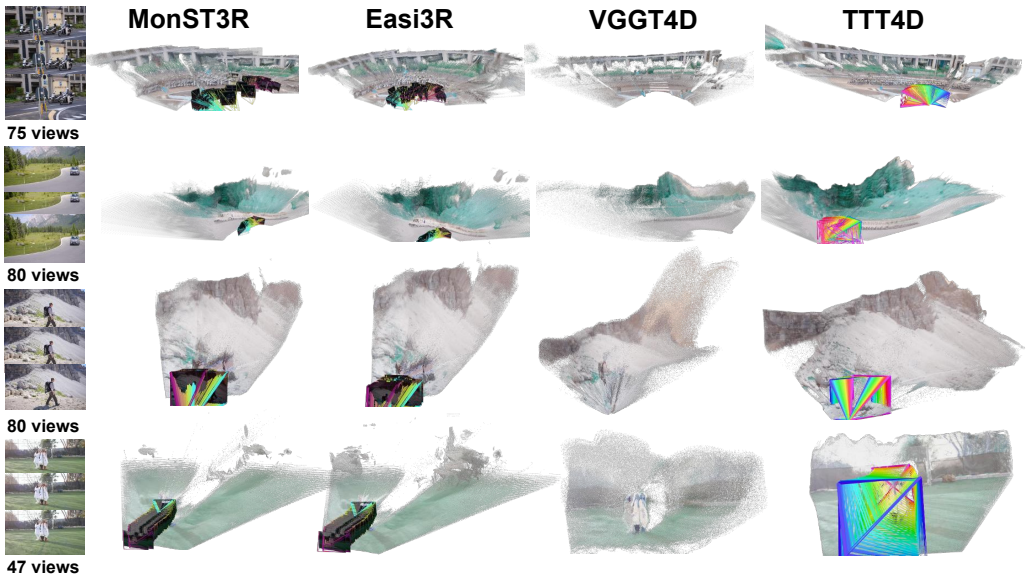


Figure 6: **Qualitative Comparison.** TTT4D ($\mathcal{O}(N)$) produces more coherent and consistent reconstruction of static regions. In contrast, the baseline methods exhibit misaligned background structures, reconstruction artifacts, and unstable geometry estimation.

Completeness from 0.043 to 0.037, and the median Distance from 0.156 to 0.146, demonstrating that suppressing dynamic interference substantially benefits both geometric accuracy and reconstruction completeness. Although VGGT4D achieves the best overall performance on the DyCheck benchmark, TTT4D substantially narrows the gap while leveraging a more efficient stateful architecture, demonstrating the effectiveness of exploiting the intrinsic dynamics of Test-Time Training for dynamic scene reconstruction. Motion evidence identifies independently moving regions from representation-space signatures and fast-weight adaptation dynamics, while refined masks and attention routing suppress dynamic interference during early geometric aggregation.

4.4 Ablation Studies

We provide comprehensive ablation studies in the supplementary material, systematically examining the contribution and complementarity of individual motion cues through both leave-one-out and single-cue evaluations. We further investigate different mask-refinement strategies to assess the effects of post-processing on dynamic-region estimation, and analyze dynamic QKV isolation across shallow, middle, and deep layer ranges to understand how layer-wise selective routing influences dynamic interference suppression and robust geometric aggregation.

5 Conclusion

We presented TTT4D, a training-free framework that extends the stateful 3D foundation model ZipMap to dynamic 4D scene reconstruction. Our key finding is that independent motion manifests through complementary adaptation- and representation-level signals during stateful inference. While individual signals may also respond to ambiguous static regions, their joint patterns enable robust motion disambiguation. By integrating these cues with spatial refinement, TTT4D produces reliable dynamic masks without external motion priors or additional training. These masks further guide shallow hard-to-soft attention routing with selective dynamic feature isolation, reducing dynamic interference in camera and geometry estimation. Experiments on five benchmarks show consistent gains in dynamic object segmentation, camera pose estimation, and 4D scene reconstruction while preserving the efficiency of stateful feed-forward models.

Ethics Statement

This study is conducted in accordance with the ICLR Code of Ethics. Our proposed approach does not require the involvement of human participants or animal subjects and does not introduce concerns related to privacy, security, or regulatory compliance. We comply with applicable licenses and responsible usage policies for all resources utilized in this work. To the best of our knowledge, this research does not result in adverse societal impacts, conflicts of interest, or fairness-related concerns. All experiments are performed following established principles of research integrity and reproducible scientific practice.

Reproducibility Statement

We aim to promote transparent and reproducible research. Upon publication, we will provide the source code, pretrained models, and comprehensive instructions required to reproduce our experimental results. Detailed descriptions of data preparation procedures and evaluation protocols are provided in the main manuscript and appendix, while further implementation specifications are included in the supplementary materials.

References

Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120(2): 153–168, 2016.

Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. Must3r: Multi-view network for stereo 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1050–1060, 2025.

John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, 8(6):679–698, 1986.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Easi3r: Estimating disentangled motion from dust3r without training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9158–9168, 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.

Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of the IEEE international conference on computer vision*, pp. 686–695, 2017.

Sven Elflein, Ruilong Li, Sérgio Agostinho, Zan Gojcic, Laura Leal-Taixé, Qunjie Zhou, and Aljosa Osep. Vgg-t^3: Offline feed-forward 3d reconstruction at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36464–36474, 2026.

Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009.

Hang Gao, Ruilong Li, Shubham Tulsiani, Bryan Russell, and Angjoo Kanazawa. Monocular dynamic view synthesis: A reality check. *Advances in Neural Information Processing Systems*, 35: 33768–33780, 2022.

Ariel Gordon, Hanhan Li, Rico Jonschkowski, and Anelia Angelova. Depth from videos in the wild: Unsupervised monocular depth learning from unknown cameras. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8977–8986, 2019.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 414–424, 2026.

Nan Huang, Wenzhao Zheng, Chenfeng Xu, Kurt Keutzer, Shanghang Zhang, Angjoo Kanazawa, and Qianqian Wang. Segment any motion in videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3406–3416, 2025.

Matthias Innmann, Michael Zollhöfer, Matthias Nießner, Christian Theobalt, and Marc Stamminger. Volumedeform: Real-time volumetric non-rigid reconstruction. In *European conference on computer vision*, pp. 362–379. Springer, 2016.

Haian Jin, Rundi Wu, Tianyuan Zhang, Ruiqi Gao, Jonathan T Barron, Noah Snavely, and Aleksander Hołyński. Zipmap: Linear-time stateful 3d reconstruction via test-time training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21748–21759, 2026.

Laurynas Karazija, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Learning segmentation from point trajectories. *arXiv preprint arXiv:2501.12392*, 2025.

Margret Keuper, Bjoern Andres, and Thomas Brox. Motion trajectory segmentation via minimum cost multicuts. In *Proceedings of the IEEE international conference on computer vision*, pp. 3271–3279, 2015.

Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1611–1621, 2021.

Skanda Koppula, Ignacio Rocco, Yi Yang, Joe Heyward, Joao Carreira, Andrew Zisserman, Gabriel Brostow, and Carl Doersch. Tapvid-3d: A benchmark for tracking any point in 3d. *Advances in Neural Information Processing Systems*, 37:82149–82165, 2024.

Hala Lamdouar, Weidi Xie, and Andrew Zisserman. Segmenting invisible moving objects. 2021.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European conference on computer vision*, pp. 71–91. Springer, 2024.

Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10486–10496, 2025.

Long Lian, Zhirong Wu, and Stella X Yu. Bootstrapping objectness from videos by relaxed common fate and visual grouping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14582–14591, 2023.

Etienne Meunier, Anaïs Badoual, and Patrick Bouthemy. Em-driven unsupervised learning for efficient motion segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (4):4462–4473, 2022.

Richard A Newcombe, Dieter Fox, and Steven M Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 343–352, 2015.

Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision*, pp. 58–77. Springer, 2024.

Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20133–20143, 2023.

Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *Proceedings of the IEEE international conference on computer vision*, pp. 1777–1784, 2013.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 724–732, 2016.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. " grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3):309–314, 2004.

Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pp. 501–518. Springer, 2016.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

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*, pp. 573–580. IEEE, 2012.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, et al. Learning to (learn at test time): Rnns with expressive hidden states. *arXiv preprint arXiv:2407.04620*, 2024.

Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. In *2025 International Conference on 3D Vision (3DV)*, pp. 78–89. IEEE, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10510–10522, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20697–20709, 2024.

Wenguan Wang, Jianbing Shen, and R Yang. F. porikli, saliency-aware video object segmentation. *IEEE Trans. Pattern Anal. Mach. Intell*, 40(1):20–33, 2018.

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

Junyu Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. *Advances in neural information processing systems*, 35:28023–28036, 2022.

Junyu Xie, Charig Yang, Weidi Xie, and Andrew Zisserman. Moving object segmentation: All you need is sam (and flow). In *Proceedings of the Asian conference on computer vision*, pp. 162–178, 2024.

Kai Xu, Tze Ho Elden Tse, Jizong Peng, and Angela Yao. Das3r: Dynamics-aware gaussian splatting for static scene reconstruction. *arXiv preprint arXiv:2412.19584*, 2024.

Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7177–7188, 2021.

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.

Yanchao Yang, Antonio Loquercio, Davide Scaramuzza, and Stefano Soatto. Unsupervised moving object detection via contextual information separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 879–888, 2019.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Forrester Cole, Deqing Sun, Ming-Hsuan Yang, et al. Monst3r: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, volume 2025, pp. 82863–82886, 2025a.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T Freeman, and Hao Tan. Test-time training done right. *arXiv preprint arXiv:2505.23884*, 2025b.

Zhoutong Zhang, Forrester Cole, Zhengqi Li, Michael Rubinstein, Noah Snavely, and William T Freeman. Structure and motion from casual videos. In *European Conference on Computer Vision*, pp. 20–37. Springer, 2022.

Appendix

A LLMs USAGE

Large language models (LLMs) were used solely as general-purpose writing assistants to improve the presentation quality of this manuscript. Specifically, LLMs were utilized for language proofreading, grammar correction, sentence refinement, and enhancing the clarity and readability of the text. All scientific ideas, methodological designs, experimental setups, analyses, and conclusions presented in this work were independently developed and verified by the authors. All AI-assisted modifications were carefully reviewed by the authors to ensure accuracy, consistency, and alignment with the intended meaning. Furthermore, all factual statements, citations, and technical descriptions were independently checked by the authors. We take full responsibility for the final content of this paper, including all text, claims, experimental results, and conclusions.

B IMPLEMENTATION DETAILS

Level selection. The dynamic mask is constructed from four complementary internal cues extracted at different depths of the ZipMap backbone. Specifically, the TTT readout mismatch proxy is obtained from L8, the Fast-Weight adaptation demand from L7–L8, the Frame-centered query deviation from L6, and the Value-channel dispersion from L2–L5 (Table 4).

Hard-to-Soft information routing. Based on the patch-level dynamic mask, we divide the patch tokens into dynamic and static groups. To reduce the influence of dynamic regions during feature aggregation, we introduce a hard-to-soft routing strategy across the early transformer layers. Specifically, at layers L2–L3, patch tokens are restricted to attend only to static patch tokens and special tokens, preventing potentially corrupted dynamic regions from contributing to patch-level feature aggregation. In contrast, camera tokens and register tokens maintain their original full attention over all tokens, preserving global scene-level reasoning. At layer L4, the routing gradually transitions from hard to soft. The original global key connectivity is restored, while the value features of dynamic patch tokens are replaced with a static consensus representation computed from static patches. This design allows the model to recover global contextual interaction while reducing the negative influence of dynamic content in value aggregation. After L4, all subsequent layers and the original TTT update mechanism remain unchanged. Therefore, the proposed routing strategy only modifies early information flow while preserving the original feed-forward inference behavior.

Cues	Source layers
TTT readout mismatch proxy	L8
Fast-Weight adaptation demand	L7–L8
Frame-centered query deviation	L6
Value-channel dispersion	L2–L5

Table 4: Source layers of the cues used to construct dynamic mask. For cues extracted from multiple layers, the corresponding layer-wise responses are aggregated using equal-weight averaging. Here, L0 refers to the first block, and the ZipMap TTT architecture comprises a total of 24 blocks.

C ABLATION STUDIES

Dynamic secrets. We first evaluate the contribution of each internal cue by removing it from the complete model. As shown in Table 5, removing the residual cue (w/o r) causes the largest performance degradation across all three datasets, demonstrating its dominant role in geometry-aware motion discrimination. Removing the update cue (w/o u) also leads to a substantial drop, highlighting its importance in recovering complete dynamic regions. The magnitude cue mainly improves J_M and J_R, yielding stronger overall performance despite a slight trade-off in boundary accuracy. Removing the entropy cue (w/o e) improves several metrics on DAVIS-2016 but degrades performance on DAVIS-all, suggesting that it mainly benefits cross-scene generalization. Overall, the four

Table 5: **Ablation of the motion cues on DAVIS.** Starting from the complete model, we remove one cue at a time while keeping all remaining components unchanged. Here, u, r, m, and e denote the write-aware adaptation, query deviation, update magnitude, and value dispersion cues, respectively.

Setting	DAVIS-2016				DAVIS-2017				DAVIS-all			
	JM	JR	FM	FR	JM	JR	FM	FR	JM	JR	FM	FR
w/o u	53.49	56.77	51.09	50.82	52.45	55.55	49.65	49.91	52.36	57.84	50.86	54.46
w/o r	46.25	49.64	42.85	33.19	47.95	49.88	44.22	35.62	47.95	53.55	45.93	44.52
w/o m	59.40	67.36	<u>56.84</u>	<u>64.79</u>	57.18	63.60	**54.46**	**61.74**	<u>55.52</u>	<u>63.33</u>	**54.81**	**61.69**
w/o e	**62.53**	**76.67**	**58.46**	**66.04**	<u>57.81</u>	**67.80**	53.84	58.64	52.94	60.38	51.82	54.47
Ours	<u>60.58</u>	<u>72.86</u>	56.51	63.76	**58.32**	<u>66.91</u>	<u>54.20</u>	<u>60.06</u>	**55.98**	**65.16**	<u>54.04</u>	<u>60.11</u>

Table 6: **Single-cue ablation on DAVIS.** Each row uses only one motion cue for dynamic prediction. The best and second-best results are **bold** and <u>underlined</u>, respectively.

Cue	DAVIS-2016				DAVIS-2017				DAVIS-all			
	JM	JR	FM	FR	JM	JR	FM	FR	JM	JR	FM	FR
u	45.43	51.27	45.19	49.70	40.65	41.77	41.27	41.47	38.92	40.94	41.28	42.34
r	<u>49.16</u>	<u>56.68</u>	<u>50.53</u>	<u>58.23</u>	<u>46.66</u>	<u>52.63</u>	<u>46.96</u>	<u>50.31</u>	<u>43.81</u>	<u>47.27</u>	<u>45.12</u>	<u>46.80</u>
m	38.75	41.79	36.40	23.91	39.61	37.20	37.02	24.04	37.66	35.57	37.07	28.59
e	12.01	2.45	20.00	0.50	17.38	7.76	22.53	2.35	17.09	9.19	22.96	4.15
Ours	**60.58**	**72.86**	**56.51**	**63.76**	**58.32**	**66.91**	**54.20**	**60.06**	**55.98**	**65.16**	**54.04**	**60.11**

cues provide complementary evidence for robust motion disambiguation. We further evaluate each cue individually (Table 6). Among all four cues, the residual cue consistently achieves the strongest standalone performance, followed by the update cue, whereas the magnitude cue provides limited discriminative capability and the entropy cue alone is insufficient for accurate segmentation. Nevertheless, all individual cues remain substantially inferior to the complete model, demonstrating that the proposed equal-weight fusion effectively exploits their complementary properties. As shown in Table 9, we further compare with 2D baselines, where TTT4D achieves state-of-the-art segmentation performance among zero-shot methods.

Mask refinement. We also investigate the effect of mask refinement (Table 7). These results demonstrate that hysteresis and GrabCut play complementary roles, where connectivity-aware hysteresis provides reliable object support and GrabCut mainly refines object boundaries.

Selective routing. We compare our selective routing with continuous dynamic-QKV isolation over different depth ranges (Table 8). Replacing TTT4D with uniform layer isolation consistently degrades pose estimation across all three datasets. Isolating all layers or only the middle layers causes the largest performance drop, indicating that aggressive removal of dynamic interactions severely disrupts geometric reasoning. Restricting the isolation to early or late layers alleviates this degradation but still fails to match the proposed strategy. These results demonstrate that effective dynamic suppression requires layer-selective routing rather than uniformly blocking dynamic information throughout a continuous depth range.

D Efficiency Analysis

We compare the end-to-end runtime of TTT4D with VGGT4D and Easi3R on the complete TUM-Dynamics and ADT benchmarks. As shown in Tables 10 and 11, TTT4D is substantially more efficient across all stages. On TUM-Dynamics, our method requires only 34.76 s per sequence on average, achieving $5.85\times$ and $10.29\times$ speedups over VGGT4D and Easi3R, respectively. On ADT, the average runtime is reduced to 26.19 s, corresponding to $6.87\times$ and $8.27\times$ speedups. This advantage consistently extends to cue extraction, mask post-processing, and the second forward pass. In particular, extracting motion cues directly from stateful inference avoids the expensive attention

Table 7: **Ablation of mask refinement on the DAVIS datasets.** We evaluate the contribution of each post-processing step using the same motion score map. "GrabCut" applies a single GrabCut refinement initialized from the binarized mask, "Hysteresis" performs hysteresis thresholding only, and "Ours" combines hysteresis thresholding with the same GrabCut refinement.

Method	DAVIS-2016				DAVIS-2017				DAVIS-all			
	JM	JR	FM	FR	JM	JR	FM	FR	JM	JR	FM	FR
GrabCut	45.17	40.47	37.13	22.42	45.70	43.39	35.91	20.22	44.12	42.72	36.92	26.63
Hysteresis	52.45	58.32	44.23	37.75	50.40	54.73	41.12	32.59	48.45	55.84	42.31	39.70
Ours	**60.58**	**72.86**	**56.51**	**63.76**	**58.32**	**66.91**	**54.20**	**60.06**	**55.98**	**65.16**	**54.04**	**60.11**

Table 8: **Ablation of dynamic QKV isolation across different layer ranges on TUM-Dynamics, DyCheck, and ADT.** Shallow, middle, and deep denote applying dynamic QKV isolation to layers L1–L8, L9–L16, and L17–L24, respectively.

Strategy	TUM-Dynamics			DyCheck			ADT		
	ATE↓	RTE↓	RRE↓	ATE↓	RTE↓	RRE↓	ATE↓	RTE↓	RRE↓
Shallow layers	0.011	0.006	0.300	0.022	0.017	1.429	0.031	0.013	0.346
Middle layers	0.101	0.053	1.660	0.069	0.050	5.162	0.278	0.106	6.827
Deep layers	0.019	0.012	0.943	0.018	0.016	1.738	0.041	0.019	0.863
All layers	0.088	0.049	1.689	0.073	0.060	6.158	0.282	0.096	11.695
Ours	**0.009**	**0.005**	**0.298**	**0.016**	**0.013**	**0.973**	**0.030**	**0.012**	**0.330**

processing and additional inference overhead of previous training-free approaches. Furthermore, for mask optimization and post-processing, Easi3R aggregates pairwise attention statistics, applies K-means clustering and refinement, reconstructs the masked pair graph, and performs a second global alignment followed by 300 iterations of pose optimization. VGGT4D organizes the collected Q/K features, extracts dynamic maps, performs feature clustering and adaptive thresholding, and subsequently refines the masks through depth- and pose-based geometric consistency. In contrast, TTT4D employs lightweight mask refinement and post-processing, avoiding these costly iterative optimization procedures. Consequently, TTT4D achieves substantially lower end-to-end runtime while retaining the linear-time scalability of the underlying stateful architecture.

To further examine scalability beyond the sequence lengths covered by the standard benchmarks, we conduct a long-sequence stress test on TUM-Dynamics, progressively increasing the input from 50 to 850 frames. All methods are evaluated on the same machine equipped with a single NVIDIA RTX PRO 6000 GPU with 96 GB of VRAM and 120 GB of host memory, using their complete official refinement pipelines. As shown in Table 12, the computational cost of TTT4D grows approximately linearly with sequence length: the total runtime increases from 24.40 s at 50 frames to 324.24 s at 850 frames, while peak allocated VRAM increases from 11.80 GB to 91.73 GB. In contrast, the competing methods quickly become infeasible as the sequence length increases. VGGT4D is externally terminated by the operating system under host-memory pressure at 100 frames, rather than encountering a CUDA out-of-memory error, whereas Easi3R already consumes 60.49 GB of peak allocated VRAM at 100 frames and encounters an out-of-memory error during its official optimization at 200 frames. TTT4D, by contrast, successfully processes all tested sequence lengths up to 850 frames on the same hardware. These results complement the full-benchmark runtime comparison and demonstrate that the computational efficiency of TTT4D translates into substantially better practical scalability for long-sequence 4D reconstruction.

Table 9: **Comparisons of dynamic object segmentation on DAVIS.** The best and second-best results within each supervision setting are **bold** and underlined, respectively.

Method	Input			Output		DAVIS-16	DAVIS-17	DAVIS-All
	RGB	Optical Flow	Point Tracks	Mask	4D Reconstruction.	JM↑	JM↑	JM↑
Supervised								
SFL (Cheng et al., 2017)	✓	✓		✓		67.4	–	–
SIMO (Lamdouar et al., 2021)	✓	✓		✓		67.8	–	–
OCLR-flow (Xie et al., 2022)	✓	✓		✓		72.0	–	–
OCLR-TTA (Xie et al., 2022)	✓	✓		✓		80.8	–	–
FlowSAM (Xie et al., 2024)	✓	✓		✓		87.1	–	–
SegAnyMo (Huang et al., 2025)	✓		✓	✓		**90.6**	–	–
DAS3R (Xu et al., 2024)	✓			✓	✓	41.13	44.51	43.33
Unsupervised								
SAGE (Wang et al., 2018)	✓	✓		✓		42.6	–	–
CUT (Keuper et al., 2015)	✓	✓		✓		55.2	–	–
FTS!(Papazoglou & Ferrari, 2013)	✓	✓		✓		55.8	–	–
CIS (Yang et al., 2019)	✓	✓		✓		70.3	–	–
Motion Grouping (Yang et al., 2021)	✓	✓		✓		68.3	–	–
EM (Meunier et al., 2022)	✓	✓		✓		69.3	–	–
RCF-Stage1 (Lian et al., 2023)	✓	✓		✓		80.2	–	–
RCF-All (Lian et al., 2023)	✓	✓		✓		82.1	–	–
LRTL (Karazija et al., 2025)	✓	✓	✓	✓		**82.2**	–	–
Zero-shot								
MonoST3R (Zhang et al., 2025a)	✓	✓		✓	✓	40.42	38.07	36.98
Easi3R$_{\mathrm{DUSt3R}}$ (Chen et al., 2025a)	✓			✓	✓	50.10	46.86	44.10
Easi3R$_{\mathrm{MonST3R}}$ (Chen et al., 2025a)	✓			✓	✓	54.93	<u>54.75</u>	<u>51.64</u>
VGGT4D (Hu et al., 2026)	✓			✓	✓	<u>60.26</u>	53.65	47.93
TTT4D (Ours)	✓			✓	✓	**60.58**	**58.32**	**55.98**

Table 10: Per-sequence average runtime on the TUM-Dynamics and ADT benchmarks. TUM-Dynamics contains seven sequences (630 frames), whereas ADT contains eight sequences (480 frames). Here, "Total" denotes the end-to-end runtime. "Cue fwd." denotes the first feed-forward pass used to extract motion cues, while "Second fwd." denotes the second feed-forward pass using the estimated dynamic masks for dynamic-aware inference. "Mask post." includes mask optimization and additional optimization applied before the second inference. All values are in seconds.

Dataset	Method	Total	Cue fwd.	Mask post.	Second fwd.
TUM	Ours	**34.76**	**4.31**	**28.13**	**2.31**
	VGGT-4D	203.33	46.44	109.29	47.60
	Easi3R	357.49	73.00	215.67	68.37
ADT	Ours	**26.19**	**3.80**	**20.34**	**2.04**
	VGGT-4D	179.88	41.14	97.31	41.43
	Easi3R	216.63	47.35	124.09	44.89

E Details of Mask refinement

Let $S_t : \Omega_t \to [0, 1]$ be the fused pixel-level dynamic score, where Ω_t denotes the image domain. To suppress local response noise while preserving the overall score distribution, the score map is first smoothed using a 3×3 local mean filter and subsequently normalized. Thresholds h and ℓ partition the normalized score into strong and weak supports:

$$\begin{aligned} \mathcal{A}_t &= \left\{ p \in \Omega_t \mid \widetilde{S}_t(p) \geq h \right\}, \\ \mathcal{B}_t &= \left\{ p \in \Omega_t \mid \widetilde{S}_t(p) \geq \ell \right\}. \end{aligned} \tag{11}$$

The strong support identifies high-confidence dynamic responses, whereas the weak support preserves spatially connected regions with lower confidence. Let $\mathcal{H}(\mathcal{B}_t)$ denote the set of 8-connected

Table 11: Cumulative runtime over the TUM-Dynamics and ADT benchmarks. All values are in seconds and lower is better.

Dataset	Method	Total	Cue fwd.	Mask post.	Second fwd.
TUM	Ours	**243.31**	**30.15**	**196.89**	**16.17**
	VGGT-4D	1423.33	325.10	765.04	333.20
	Easi3R	2502.45	510.97	1509.72	478.62
ADT	Ours	**209.52**	**30.44**	**162.75**	**16.29**
	VGGT-4D	1439.04	329.13	778.48	331.43
	Easi3R	1733.03	378.79	992.70	359.15

Table 12: Long-sequence runtime and memory comparison on TUM-Dynamics dataset, evaluated on a single NVIDIA RTX PRO 6000 (96 GB) GPU with 120 GB host memory. "OOM" denotes failure due to memory limitations.

Frames	Ours	VGGT-4D	Easi3R
	Total Time (s) / Mask Post. (s) / VRAM (GB)		
50	24.40/20.67/11.80	109.46/60.05/19.03	173.88/98.57/30.35
100	46.82/39.55/16.79	OOM	402.26/243.03/60.49
200	79.88/65.58/26.79	OOM	OOM
300	119.13/97.84/36.78	OOM	OOM
400	152.09/123.69/46.78	OOM	OOM
500	196.78/161.29/56.78	OOM	OOM
600	235.55/193.00/66.75	OOM	OOM
700	275.43/225.64/76.74	OOM	OOM
800	308.89/252.06/86.76	OOM	OOM
850	324.24/263.75/91.73	OOM	OOM

components in the weak support. A connectivity-aware hysteresis procedure retains only components that contain at least one strong response:

$$M_t^0 = \bigcup_{\mathcal{C} \in \mathcal{H}(\mathcal{B}_t)} \{\mathcal{C} \mid \mathcal{C} \cap \mathcal{A}_t \neq \varnothing\}. \tag{12}$$

This removes low-confidence responses while preserving coherent dynamic regions. The resulting coarse mask is further aligned with image boundaries through a lightweight GrabCut refinement. Foreground and background regions are obtained by eroding and dilating the coarse mask:

$$\begin{aligned} \mathcal{F}_t &= \mathrm{Erode}_r(M_t^0), \\ \widetilde{\mathcal{B}}_t &= \Omega_t \setminus \mathrm{Dilate}_r(M_t^0). \end{aligned} \tag{13}$$

Pixels in $\mathcal{F}_t$ are labeled as definite foreground, while those in $\widetilde{\mathcal{B}}_t$ are labeled as definite background. The remaining boundary band is regarded as uncertain and optimized according to the standard GrabCut objective:

$$\begin{aligned} E(L_t) = \sum_{p \in \Omega_t} D_p(L_t(p); I_t) \quad + \\ \lambda \sum_{(p,q) \in \mathcal{E}} [L_t(p) \neq L_t(q)] \exp\Big(-\gamma \, \|I_t(p) - I_t(q)\|_2^2\Big). \end{aligned} \tag{14}$$

subject to $L_t(p) = 1$ for $p \in \mathcal{F}_t$ and $L_t(p) = 0$ for $p \in \widetilde{\mathcal{B}}_t$. Here, D_p denotes the unary appearance cost for foreground and background assignment, and $\mathcal{E}$ contains neighboring pixel pairs. Here, the first term encourages each pixel to be assigned to the foreground or background according to its appearance likelihood under the corresponding color model. The second term enforces spatial smoothness by encouraging neighboring pixels with similar colors to share the same label, while permitting segmentation boundaries at locations with strong color discontinuities. The hyperparameters λ and γ balance the contributions of the two terms. After one GrabCut iteration, definite and

probable foreground labels are merged to obtain the final pixel-level dynamic mask. This refinement uses only RGB appearance cues and introduces no additional motion evidence. The resulting mask is then mapped to the ZipMap patch grid by nearest-neighbor assignment.

F More Visualization

To provide a detailed view of the hierarchical motion representations learned by TTT4D, we visualize the motion cues generated across its 24 layers for the tennis and hike scenes, as shown in Figs. 7–10 and Figs. 11–14, respectively. The visualizations include the input RGB frame and the motion cues $u_{t,j}$, $r_{t,j}$, $m_{t,j}$, and $e_{t,j}$. The layers are displayed as L01–L24, corresponding to layers 0–23 in the implementation. S denotes the final result produced by the TTT4D strategy.

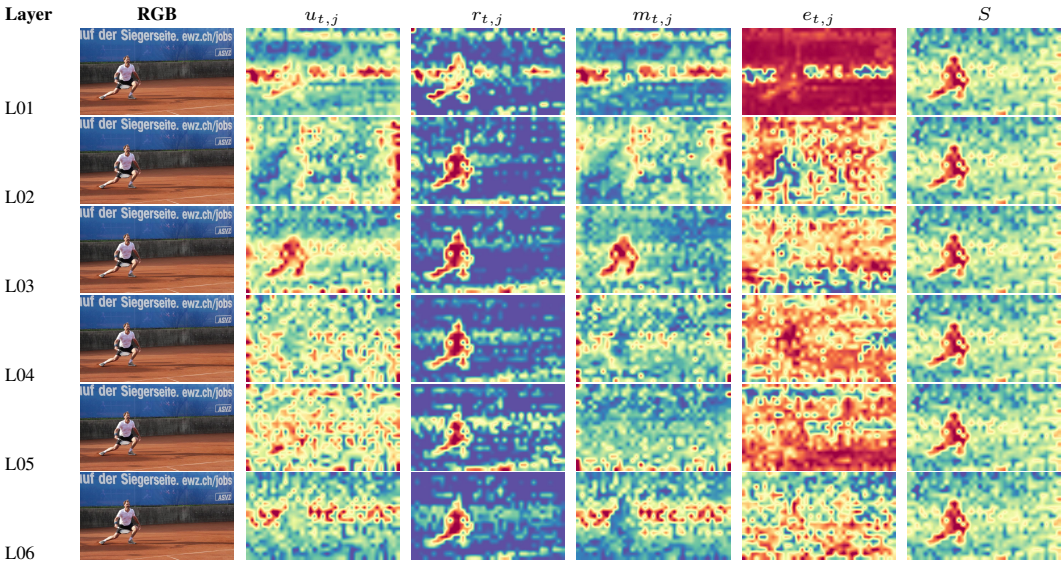


Figure 7: Layer-wise TTT4D motion cues on tennis for layers L01–L06.

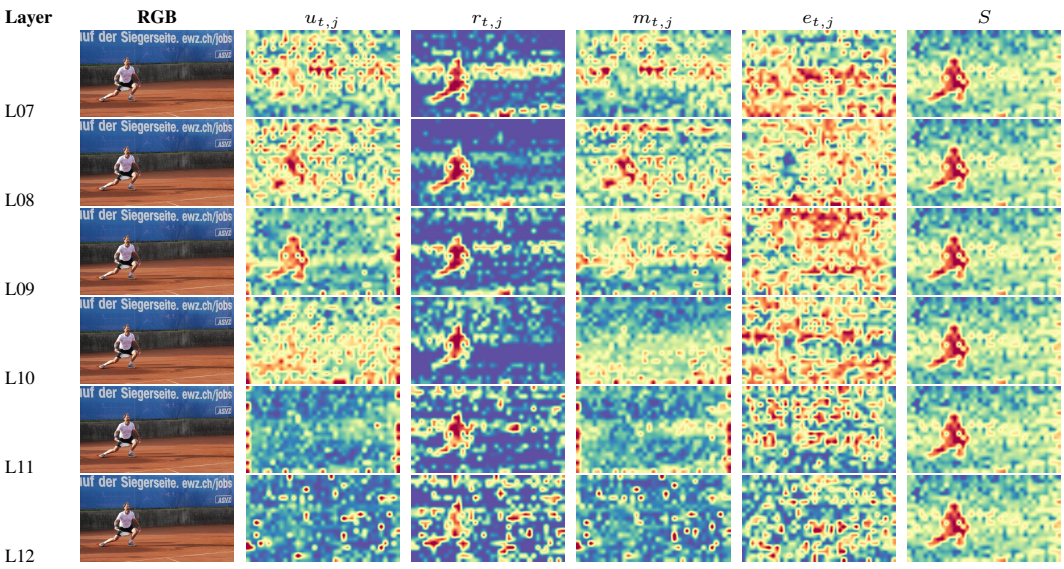


Figure 8: Layer-wise TTT4D motion cues on tennis for layers L07–L12.

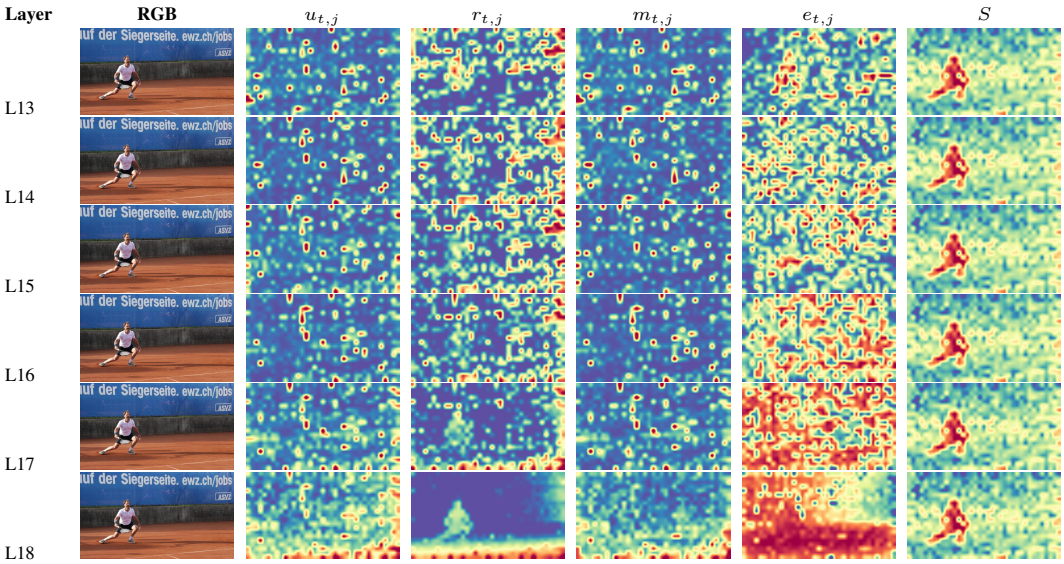


Figure 9: Layer-wise TTT4D motion cues on tennis for layers L13–L18.

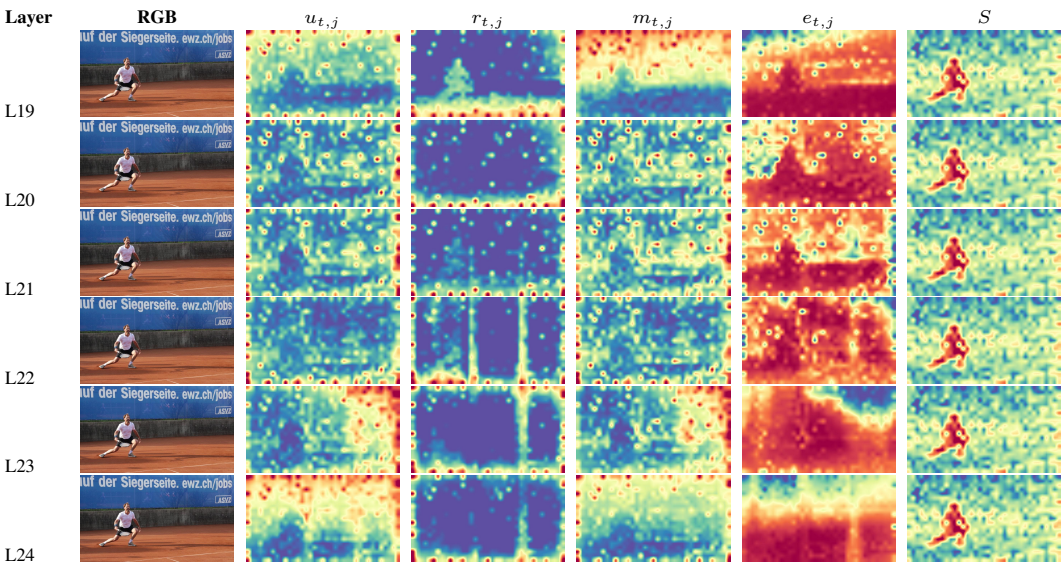


Figure 10: Layer-wise TTT4D motion cues on tennis for layers L19–L24.

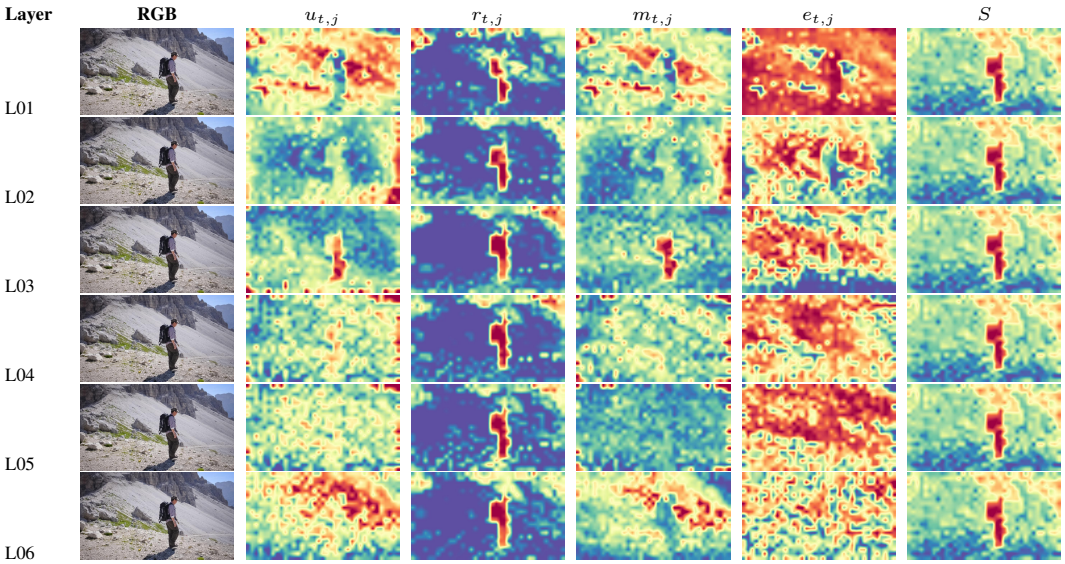


Figure 11: Layer-wise TTT4D motion cues on hike for layers L01–L06.

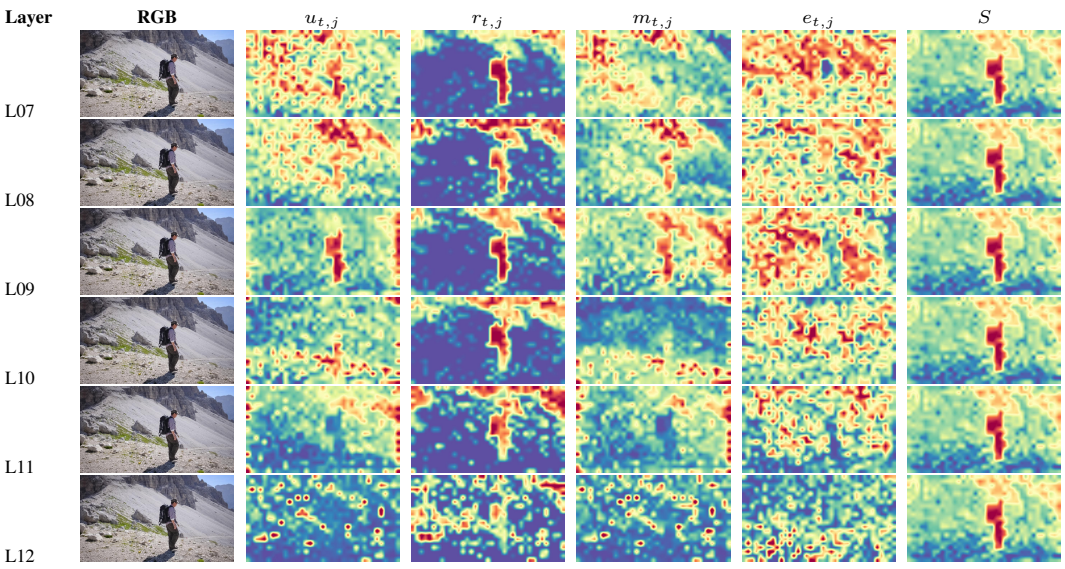


Figure 12: Layer-wise TTT4D motion cues on hike for layers L07–L12.

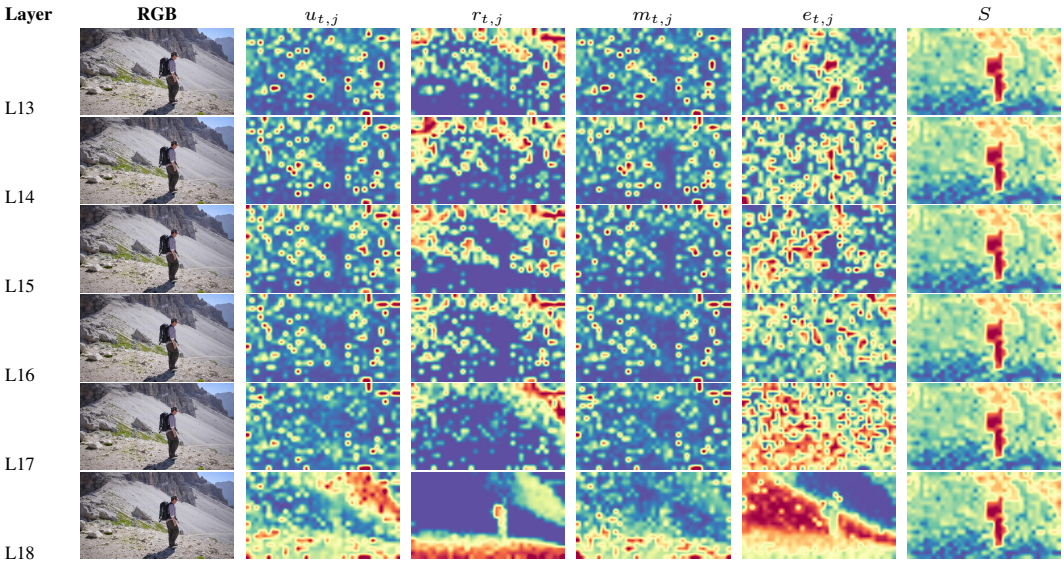


Figure 13: Layer-wise TTT4D motion cues on hike for layers L13–L18.

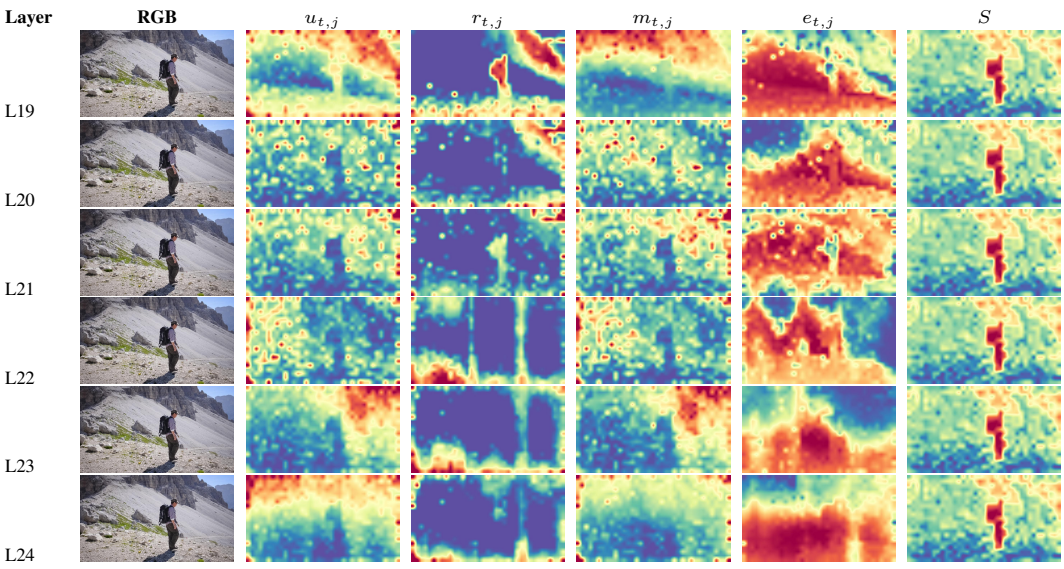


Figure 14: Layer-wise TTT4D motion cues on hike for layers L19–L24.