READING OUT CAMERA-DECOUPLED MOTION FROM FROZEN VIDEO DIFFUSION FEATURES

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding scene motion independently of camera movement is fundamental to robotics, autonomous navigation, and video understanding. Apparent motion in monocular video, however, entangles scene dynamics with camera movement. Video diffusion models generate coherent object motion under diverse camera movement, raising the question of whether their frozen features support camera-decoupled motion readout. We probe this question by keeping the video diffusion backbone frozen and training only a lightweight decoder for moving object segmentation (MOS) and motion magnitude estimation (MME). MOS captures the clip-level extent of independently moving objects, while MME estimates the per-pixel, scene-scale-normalized magnitude of world-space 3D displacement between consecutive frames. These readouts are strong enough to outperform dedicated moving object segmentation methods, while comparing favorably with leading feed-forward 4D reconstruction models in 3D motion magnitude estimation. Through controlled comparisons with geometric features, including synthetic-to-real transfer, we provide evidence that this capability is rooted in pretrained video diffusion features rather than decoder training alone. Together, these results establish frozen video diffusion features as effective representations for reading out camera-decoupled scene motion at both object and point levels.

1 INTRODUCTION

Humans naturally perceive motion in the world, distinguishing the motion of objects in the scene from their own viewpoint motion. This ability is also fundamental to robotics, autonomous navigation, video understanding (Li et al., 2026), and 3D/4D reconstruction (Wang et al., 2025; Zhang et al., 2025b). Motion observed in a video, however, entangles both scene dynamics and camera movement. As illustrated in Fig. 1 (left), even a static background appears to move as the camera changes viewpoint. Moreover, motion may be localized to particular object parts or occur only intermittently (Fig. 1, right). These factors make it challenging to identify independently moving objects and quantify their motion from RGB video.

Existing approaches draw on optical flow, image correspondences, object segmentation, and scene geometry, each emphasizing a different aspect of the visual world. These cues are valuable for motion perception, but do not by themselves specify which objects move independently of the camera or how much individual scene points move in world space. This raises the question of whether a single pretrained representation can support both forms of camera-decoupled motion readout.

Recent video diffusion models can generate realistic object movement alongside camera motion, with this capability continuing to improve with scale (Wan et al., 2025; Robbyant Team et al., 2026). Their ability to maintain coherent object structure as objects move across frames suggests that their learned representations may encode both scene motion cues and object structure. These observations make video diffusion models promising candidates for camera-decoupled motion perception.

Motivated by this, we investigate whether frozen video diffusion features can support camera-decoupled motion perception at both object and point levels. We find that a lightweight decoder is sufficient to extract fine-grained motion information from these features. At the object level, the decoder can segment the complete regions of objects moving independently of the camera; at the point level, it can estimate the magnitude of world-space 3D displacement between consecutive frames for each visible scene point. We refer to these two tasks as moving object segmentation (MOS)

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

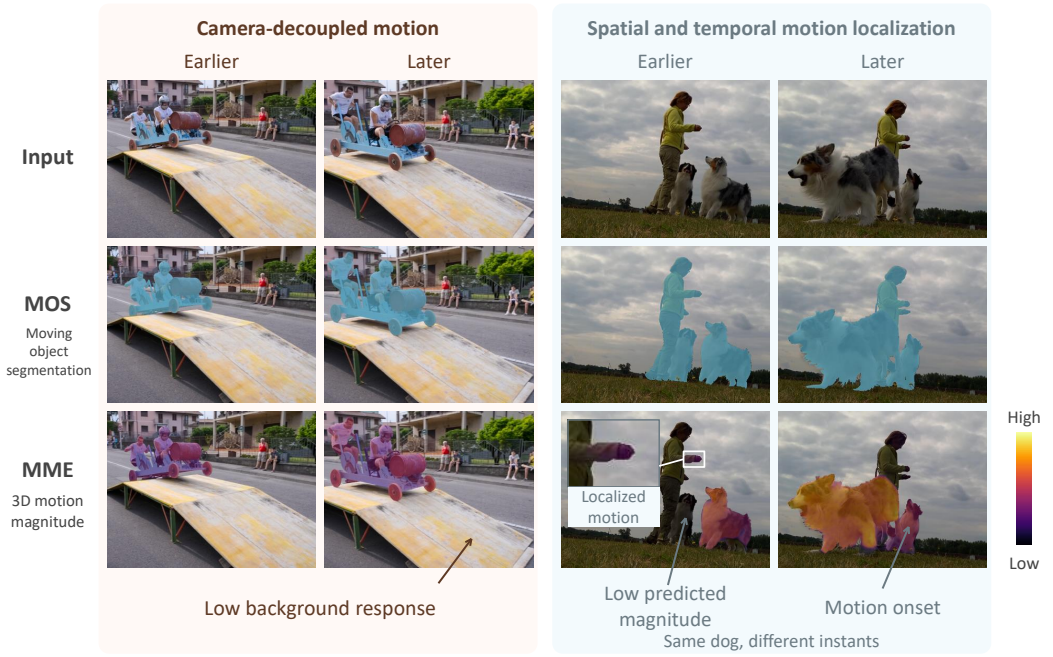

Figure 1: Given an input video, we decode two complementary descriptions of scene dynamics in a feed-forward pass from frozen video diffusion features. Moving object segmentation captures the complete regions of objects that move over time, including parts that are momentarily stationary, whereas motion magnitude describes instantaneous local motion intensity, such as partial motion and motion onset. The left example shows how our model distinguishes intrinsic object motion from apparent image motion induced by substantial camera viewpoint changes.

and motion magnitude estimation (MME), respectively. While MOS assesses object-level motion perception, MME further tests whether the frozen features support fine-grained physical motion estimation beyond binary moving-object membership. We further find that stronger video generative models yield more effective representations for these tasks. Motion readout works best at low noise levels and middle-to-late transformer layers.

In summary, our results show that frozen video diffusion features support both object-level motion segmentation and fine-grained 3D motion magnitude estimation. Our method achieves $\mathcal{J}$ scores of 92.0 on FBMS (Ochs et al., 2014), 85.0 on SegTrack v2 (Li et al., 2013), and 91.2 on DAVIS 2017-M (Pont-Tuset et al., 2017), the highest among the compared methods. For MME, it reduces clip-normalized mean absolute error relative to leading 4D reconstruction models (Luo et al., 2026; Zhu et al., 2026) by 22.3% on in-domain MOVi-E (Greff et al., 2022) and 39.5% on out-of-domain Spring (Mehl et al., 2023). Under the same training conditions, video diffusion features consistently outperform geometry-oriented features on both motion perception tasks. When decoders are trained only on synthetic data, video diffusion features retain strong performance on real videos, while geometry-oriented features degrade substantially, highlighting the stronger synthetic-to-real transfer of video diffusion features. Our contributions are summarized as follows:

- We show that camera-decoupled 3D motion magnitudes and moving-object masks can be read out directly from frozen video diffusion features through a lightweight decoder, enabling motion perception at both point and object levels.
- We validate the generalization of these motion readouts to real-world videos and systematically analyze motion representations across video generative backbones, transformer layers, and diffusion timesteps.
- We demonstrate the practical utility of these representations through state-of-the-art MOS performance and favorable results against leading feed-forward 4D reconstruction models in 3D motion magnitude estimation.

2 Related Work

2.1 Priors in Video Generative Models

Recent studies have uncovered a broad range of perceptual capabilities in video generative models (Vélez et al., 2025; Huang et al., 2025b; Wiedemer et al., 2025; Zhu et al., 2025). These capabilities have been accessed by prompting the generator (Wiedemer et al., 2025; Shrivastava et al., 2025; Kim et al., 2025), by post-training the backbone (Wang et al., 2026; Zhu et al., 2026), or, closest to our setting, by reading out frozen intermediate features (Vélez et al., 2025; Yuan et al., 2025; Huang et al., 2025b). Frozen-feature studies show that these representations encode strong cues for tracking (Nam et al., 2025; Yuan et al., 2025; Son et al., 2025) and for scene geometry (Huang et al., 2025b). However, tracking and optical flow capture apparent motion that entangles scene dynamics with camera movement, while geometry characterizes scene structure rather than which parts of it move. Whether frozen video generative features separate scene motion from camera motion has not been directly examined; we study this question through MOS at the object level and MME at the point level.

2.2 Moving Object Segmentation

Moving object segmentation (MOS) aims to identify objects that move independently of the camera, corresponding to the object-level component of intrinsic motion perception studied in this work. A major line of learning-based methods uses optical flow as a primary motion cue or self-supervisory signal (Yang et al., 2019; Meunier et al., 2023; Lian et al., 2023; Xie et al., 2022). However, optical flow typically captures motion between adjacent frames and mixes object motion with camera motion. Recent methods incorporate reconstruction-derived geometric features (He et al., 2026; Xie et al., 2026) or long-range point trajectories (Huang et al., 2025a) to support motion reasoning. Complementary objectness priors help recover complete object regions when motion cues are weak or spatially localized (Xie et al., 2024a; Huang et al., 2025a; Xie et al., 2024b; Goli et al., 2024). Beyond the representations used for motion inference, several recent methods employ SAM 2 (Ravi et al., 2025) video propagation to extend masks across frames and improve temporal consistency (Huang et al., 2025a; Xie et al., 2026). These methods combine specialized cues from different sources. In contrast, we read out moving-object masks directly from a single frozen video diffusion backbone.

2.3 World-Space 3D Motion Estimation

Moving object segmentation identifies independently moving objects but does not reveal how much they move in 3D. Estimating camera-decoupled motion magnitude requires relating the same scene points across time in a shared world frame, a capability central to monocular 4D reconstruction. Some methods estimate camera-relative scene flow (Liang et al., 2025), with extensions that separate camera and object motion (Qian et al., 2026). Dynamic point-map methods represent scene points across time in a shared reference frame, using image pairs (Sucar et al., 2025; Feng et al., 2025) or video clips (Sucar et al., 2026). Other approaches support time-conditioned queries of 3D scene structure and motion (Zhang et al., 2025a; Luo et al., 2026). Most closely related, MotionCrafter (Zhu et al., 2026) adapts a pretrained video diffusion model to jointly predict world-frame point maps and adjacent-frame scene flow. These methods train or fine-tune dedicated networks for explicit geometry and motion recovery. In contrast, we directly read out MME from frozen video diffusion features without explicit scene reconstruction, indicating how far each visible scene point has moved since the previous frame.

3 Method

As illustrated in Fig. 2, we extract spatiotemporal features from a frozen video diffusion backbone and train a lightweight decoder to jointly predict moving object segmentation and motion magnitude.

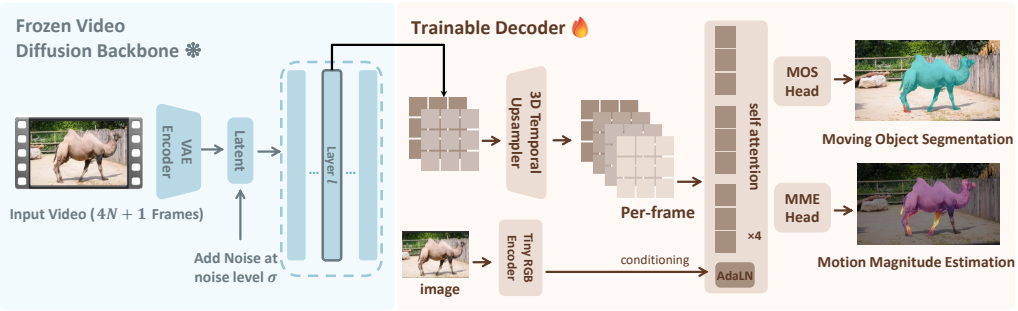

Figure 2: Overview of our framework. We add noise to the input video latent at diffusion noise level σ and extract the hidden representation from layer l of a frozen video diffusion backbone. Only the decoder is optimized: it first uses a 3D CNN to temporally upsample the extracted features, followed by a 4-layer refinement ViT (Dosovitskiy et al., 2021). Two task-specific heads then predict MOS and dense 3D scene-flow magnitude, which quantifies the intensity of local 3D motion.

3.1 Problem Formulation

Given a video $\mathcal{V} = \{I_t\}_{t=0}^{T-1}$, our framework predicts a binary moving-object mask $\hat{M}_t \in \{0,1\}^{H\times W}$ for each frame and a nonnegative motion-magnitude map $\hat{A}_t \in \mathbb{R}_{\geq 0}^{H\times W}$ for each consecutive-frame interval ($t \geq 1$). Moving Object Segmentation (MOS) identifies objects that move independently of the camera over the video clip and covers their complete visible regions. Motion Magnitude Estimation (MME) describes the 3D motion intensity of each visible scene point between consecutive frames. The two tasks characterize scene motion in terms of object extent and local displacement, respectively.

For $t \geq 1$, let $\mathbf{V}_{t-1\to t}(u,v) \in \mathbb{R}^3$ denote the displacement, in a shared world coordinate system, from frame $t-1$ to frame t of the scene point visible at pixel (u,v) in the current frame. Its MME value is defined as

$$A_t(u,v) = \frac{\|\mathbf{V}_{t-1\to t}(u,v)\|_2}{d_c}, \tag{1}$$

where d_c is the scene scale derived from the reference frame, as detailed in Appendix C.1.

3.2 Video Diffusion Feature Extraction

We extract dense spatiotemporal features from a frozen pretrained video diffusion model. We first encode the input video into latent space and perturb the latent at noise level σ, similar to Tang et al. (2023). Conditioned on an empty text prompt and the first video frame, the backbone performs a single denoising transformer forward pass, from which we extract the hidden activations at layer l, without running the full video generation process.

The backbone processes the video clip jointly, allowing the extracted features to use cross-frame context for motion readout. Backbone-specific feature extraction settings are provided in Appendix B.1.

3.3 Motion-Aware Decoder

As shown in Fig. 2, the decoder converts temporally compressed backbone features into frame-aligned dense predictions. Temporal unpacking first restores a feature map for each input frame while mixing spatial and temporal information. A per-frame ViT (Dosovitskiy et al., 2021) then refines these features spatially, with corresponding-frame RGB features from a lightweight trainable CNN supplied as conditioning through adaptive LayerNorm (AdaLN).

Two independent simplified DPT (Ranftl et al., 2021) heads fuse intermediate refinement features to predict MOS logits and nonnegative motion magnitudes, respectively. The video backbone remains frozen, while all decoder components are trained jointly. Architecture and temporal alignment details are provided in Appendix B.2.

Table 1: Quantitative comparison on moving object segmentation benchmarks. Methods are grouped according to whether inference uses cross-frame SAM 2 video propagation. The second column indicates whether a SAM image encoder (Kirillov et al., 2023; Ravi et al., 2025) is used. Our method provides competitive performance across multiple benchmarks, while offering a strong accuracy–efficiency trade-off. All baselines are tested on an RTX 4090 GPU.

Method	SAM Image Encoder	FBMS		SegTrack v2		DAVIS 2016-M		DAVIS 2017-M		Per-frame Runtime
		$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	$\mathcal{J}\uparrow$	$\mathcal{F}\uparrow$	
Without SAM 2 video propagation										
EM	No	57.9	56.0	55.5	–	76.2	74.3	–	–	–
RCF-Stage1	No	69.9	–	76.7	–	78.6	76.0	–	–	–
OCLR-flow	No	65.5	64.9	67.6	–	70.0	70.0	69.5	69.7	0.10s
OCLR-TTA	No	69.9	68.3	72.1	75.3	80.1	76.9	76.0	75.3	–
ABR	No	82.1	79.7	76.6	81.2	70.3	73.7	74.7	75.3	1.75s
FlowSAM	Yes	82.8	80.9	77.9	83.9	85.7	83.8	87.9	87.0	1.06s
GMOS-S	Yes	83.0	84.9	78.9	85.9	**90.0**	**93.2**	**90.7**	**93.1**	0.16s
Ours	No	**90.7**	**93.1**	**82.2**	**86.6**	84.8	86.9	87.1	88.5	0.18s
With SAM 2 video propagation										
RoMo	Yes	75.5	–	67.7	–	–	–	–	–	7.10s
SegAnyMo	No	78.5	82.9	76.5	85.6	89.3	89.8	90.0	89.0	3.00s
GeoMotion	No	72.5	78.5	77.3	84.3	83.5	84.3	84.3	82.9	0.59s
GMOS	Yes	91.9	93.0	82.0	88.3	88.7	**90.3**	90.0	90.6	0.32s
Ours	No	**92.0**	**93.9**	**85.0**	**90.0**	**89.7**	89.7	**91.2**	**91.0**	0.27s

3.4 Training Objectives

We train the decoder jointly for object-level segmentation and point-level motion magnitude estimation. For MOS, binary cross-entropy supervises the predicted logits using object-level motion labels. For MME, we measure the reconstruction error between the predicted magnitude $\hat{A}$ and the ground-truth magnitude A in log space to accommodate their wide dynamic range, and include a squared-magnitude regularizer.

The joint objective is

$$\mathcal{L} = \lambda_{\text{mos}}\mathcal{L}_{\text{mos}} + \lambda_{\text{mme}}\big(\mathcal{L}_{\text{rec}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}\big), \tag{2}$$

where $\mathcal{L}_{\text{rec}}$ is the log-space reconstruction loss averaged separately over high- and low-motion regions to balance their contributions, $\mathcal{L}_{\text{reg}}$ is the mean squared predicted magnitude, and $\mathcal{L}_{\text{mos}}$ is the MOS classification loss. Reconstruction constrains local motion intensity, while MOS supervision constrains complete object regions. Full loss definitions and validity handling are provided in Appendix B.3.

4 Experiments

4.1 Experimental Setup

Training data. We train the decoder on three datasets: two synthetic datasets, MOVi-E from Kubric (Greff et al., 2022) and our custom Kubric-Motion, and one real-video dataset, GMOS-2K (Xie et al., 2026). For the synthetic datasets, we derive world-space 3D motion and moving-object masks from the provided geometry and object annotations, enabling joint MOS and MME supervision. GMOS-2K is curated from five existing VOS datasets by filtering videos and annotating per-object motion intervals, and provides supervision only for MOS. We use the official GMOS-2K training split, which excludes the DAVIS sequences used for evaluation. Table 5 in the appendix summarizes the dataset sizes; processing details are provided in Appendix A.

Evaluation data. For moving object segmentation, we evaluate on DAVIS (Perazzi et al., 2016; Pont-Tuset et al., 2017), FBMS (Ochs et al., 2014), and SegTrack v2 (Li et al., 2013). For DAVIS, we follow the MOS evaluation protocol of SegAnyMo (Huang et al., 2025a), using DAVIS 2016-M and DAVIS 2017-M. For 3D motion magnitude evaluation, we use MOVi-E from Kubric (Greff et al., 2022) as the in-domain dataset, and Dynamic Replica (Karaev et al., 2023) and Spring (Mehl et al., 2023) as out-of-domain datasets. Since Dynamic Replica provides only sparse scene flow annotations, we compute metrics only on the annotated points.

Implementation details. We use Wan2.2-I2V-14B (Wan et al., 2025) as our default model. We extract features at flow-matching noise level $\sigma = 0.25$ and transformer layer $l = 30$. We train on 24-frame clips with supervision at resolutions containing 480×480 pixels per frame, while the frozen Wan image-to-video backbone processes the same clips at 480p. Unless otherwise specified, we train for 15 epochs with AdamW (Loshchilov & Hutter, 2019), learning rate 10^{-4} and weight decay 0.05. We extract features offline and train for 3 hours on 8 NVIDIA GeForce RTX 4090 GPUs, each with 48 GB of memory, using a batch size of 1 per GPU. For MOS inference, we process videos using sliding windows of up to 48 frames, while quantitative MME evaluation uses 24-frame clips. Further training details are provided in Appendix B.4.

4.2 Moving Object Segmentation

Baselines. We compare with representative flow-based MOS methods (Meunier et al., 2023; Xie et al., 2022; Lian et al., 2023; Xie et al., 2024a), as well as methods incorporating geometry, segmentation, or tracking priors (He et al., 2026; Xie et al., 2026; Goli et al., 2024; Huang et al., 2025a; Xie et al., 2024b). Several methods further use SAM 2 (Ravi et al., 2025) video propagation to refine their predictions and improve temporal consistency. We therefore report both raw predictions and results obtained by applying SAM 2 video propagation to our predicted masks, following GeoMotion (He et al., 2026).

Quantitative Results. Table 1 presents quantitative comparisons across four widely used moving object segmentation benchmarks. We report region similarity $\mathcal{J}$ (mask intersection-over-union) and contour accuracy $\mathcal{F}$ (boundary F-measure), with $\mathcal{J}\&\mathcal{F}$ denoting their arithmetic mean. Without a SAM image encoder or SAM 2 video propagation, our method achieves $\mathcal{J}/\mathcal{F}$ scores of $90.7/93.1$ on FBMS and $82.2/86.6$ on SegTrack v2, leading both metrics among the compared methods without propagation. Our raw predictions remain highly competitive on DAVIS 2016-M and DAVIS 2017-M. Other methods using SAM features achieve better results, possibly benefiting from stronger object completion and boundary refinement, especially on VOS-derived benchmarks. These raw results support effective readout of independently moving objects from frozen video diffusion features, without separate geometric or segmentation models. Adding SAM 2 video propagation consistently improves all metrics over our raw predictions and outperforms competing baselines on most metrics. Our raw model takes 0.18 s/frame including feature extraction (Table 1), using a single forward pass to an intermediate layer on temporally compressed latents.

Qualitative Analysis. Figure 3 compares moving-object masks across challenging scenes. Our method identifies distant moving objects and objects that appear briefly. It maintains complete coverage when the rabbit pauses, whereas some baselines miss much of its body. Our masks also preserve fine bicycle structures, including wheel openings, and remain complete during temporarily stationary moments. GMOS and SegAnyMo exhibit gaps between the rider and motorcycle, potentially related to their instance-wise SAM 2 propagation, whereas our method produces a continuous MOS mask. Our predictions also cover more penguins than GMOS and SegAnyMo. Some moving penguins in the background are unannotated, so their predicted masks should not automatically be treated as false positives.

4.3 Beyond Masks: Camera-Decoupled 3D Motion Magnitude

Metrics. We evaluate motion magnitude after least-squares alignment with a single nonnegative scale per clip. We report clip-normalized mean absolute error (cMAE), which divides each clip's MAE by its mean ground-truth magnitude, and threshold error rate e_{15}, the percentage of valid predictions whose absolute error exceeds 15% of the clip's maximum ground-truth magnitude. These



Figure 3: Qualitative comparison of moving object segmentation on DAVIS, FBMS, and SegTrack v2. Cyan overlays indicate moving foreground, and GT denotes ground truth. White boxes and zoomed insets show local details.

Table 2: Quantitative comparison of 3D scene-flow magnitude. DR denotes Dynamic Replica. Ours and 4RC use backward (B) scene flow, while MotionCrafter uses its native forward (F) scene flow. The two directions cover the same 23 adjacent frame pairs in each 24-frame clip.

Method	Dir.	MOVi-E		Spring		DR	
		cMAE↓	e_{15} ↓	cMAE↓	e_{15} ↓	cMAE↓	e_{15} ↓
MotionCrafter	F	1.304	2.01%	1.731	1.32%	1.377	11.69%
4RC	B	0.453	1.14%	1.252	1.80%	0.587	**5.36%**
Ours	B	**0.352**	**1.11%**	**0.757**	**1.23%**	**0.536**	6.33%

evaluation operations are distinct from normalization by d_c during training: scale alignment fits predictions to the evaluation ground truth, while cMAE normalizes the resulting error. Both metrics are averaged over sequences with valid scores. Details are provided in Appendix C.2.

Baselines. We select 4D reconstruction methods whose outputs directly yield world-space motion magnitudes without additional motion estimation, in a shared clip-level coordinate system up to a single scale factor. We compare with MotionCrafter (Zhu et al., 2026), which directly predicts adjacent-frame world-space 3D scene flow, and evaluate the magnitude of its flow vectors. We also compare with 4RC (Luo et al., 2026), whose time-conditioned 3D point trajectories are expressed in a shared clip-level coordinate system, allowing adjacent-frame motion magnitudes to be computed by temporal differencing.

Quantitative Results. Table 2 shows that our method achieves the lowest cMAE on all three benchmarks. On the in-domain MOVi-E benchmark, our method achieves the lowest cMAE and e_{15}, reducing cMAE by 22.3% compared with 4RC (0.352 versus 0.453). On out-of-domain Spring, our method performs best on both metrics, even though MotionCrafter includes Spring in its training data. On Dynamic Replica, which is unseen during our decoder training but included in 4RC's training data, our method achieves the lowest cMAE (0.536), compared with 0.587 for 4RC, although 4RC obtains lower e_{15} (5.36% versus 6.33%).

Qualitative Analysis. Figure 4 visualizes predicted 3D motion magnitudes, with warmer colors indicating larger motion. Our predictions capture localized motion at each instant, assigning larger magnitudes to rapidly moving limbs and lower magnitudes to nearly stationary regions such as the supporting foot. They also maintain low background responses under substantial camera movement.

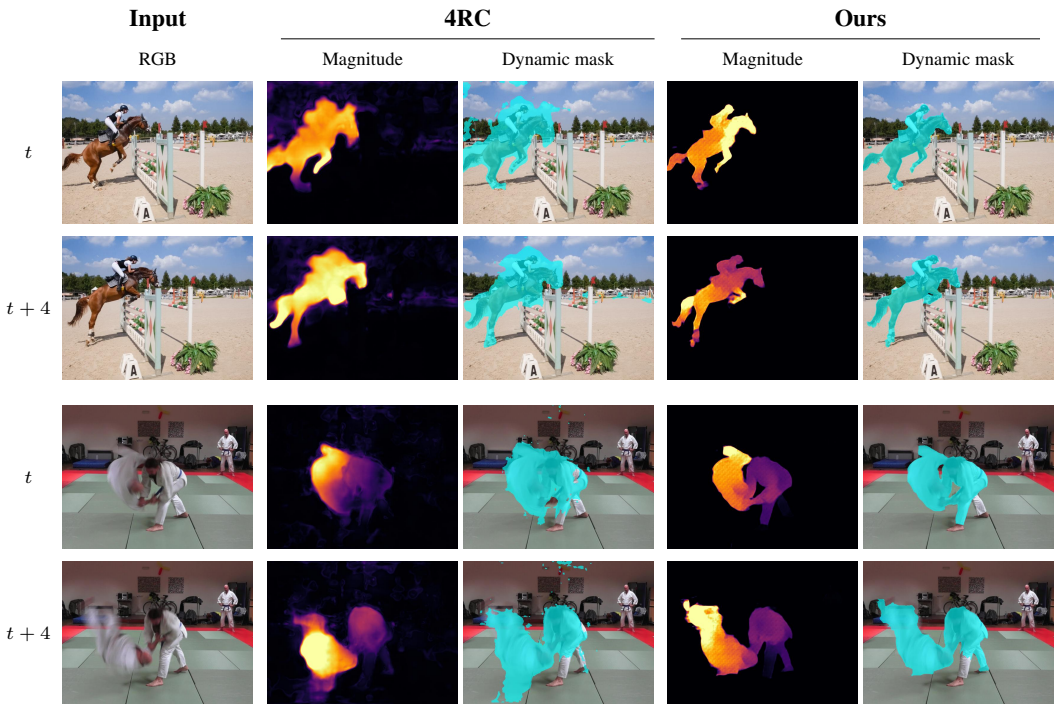

Figure 4: Qualitative comparison on DAVIS videos. Each scene is shown at frames t and $t+4$ in two consecutive rows. Columns show the RGB input, followed by 4RC and our predictions; each method provides a magnitude map and a dynamic mask obtained by thresholding its MME prediction.

Compared with the baseline, our predictions show sharper object boundaries and fewer spurious background responses.

4.4 Representation Analysis and Ablations

The backbone comparisons in Table 3 and the layer–noise sweeps in Figure 5 use an eight-epoch training budget.

How well do frozen features support motion readout? We compare frozen features from image, geometry, and video generative backbones to examine how camera-decoupled motion readout varies across model families and scales. Within each training regime, we match training data, losses, and optimization budgets; backbone-specific feature extraction settings are provided in Appendix B.1. As shown in Table 3, all evaluated video diffusion backbones outperform DINOv2 (Oquab et al., 2023) and VGGT (Wang et al., 2025) on both MOS and MME under both training regimes. The distinction is particularly pronounced in synthetic-to-real MOS transfer: without real-video supervision, DINOv2 and VGGT perform substantially worse, whereas video diffusion features retain much stronger performance on real FBMS videos. This pattern suggests that motion readouts from video diffusion features rely less on target-domain supervision, potentially benefiting from large-scale video pretraining. Within the Wan family, Wan2.2 consistently provides stronger readouts than Wan2.1 (Wan et al., 2025) on both tasks and under both training regimes. However, parameter count alone does not explain the ranking: CogVideoX-5B (Yang et al., 2025) trails the smaller Wan2.1-1.3B across all four comparisons. This comparison indicates that the accessibility of motion information depends on the pretrained representation, rather than model size alone. Overall, these results suggest that camera-decoupled motion is more readily read out from video diffusion features than from the evaluated image and geometry features, and that these readouts transfer more effectively from synthetic training data to real videos.

Where do dynamic cues emerge? We examine where dynamic cues are most readily read out within video diffusion transformers. Specifically, we jointly sweep transformer layers and noise

Table 3: Probing camera-decoupled motion in frozen visual backbones. We compare mixed synthetic and real decoder-training data with synthetic-only training. We report raw MOS on FBMS and motion magnitude estimation on MOVi-E.

Backbone	Mixed		Synthetic-only	
	FBMS $\mathcal{J}\&\mathcal{F}\uparrow$ (%)	MOVi-E cMAE↓	FBMS $\mathcal{J}\&\mathcal{F}\uparrow$ (%)	MOVi-E cMAE↓
DINOv2 (Oquab et al., 2023)	83.2	0.516	54.2	0.497
VGGT (Wang et al., 2025)	82.9	0.479	54.6	0.425
CogVideoX-5B (Yang et al., 2025)	83.4	0.465	77.5	0.403
Wan2.1-T2V-1.3B	86.9	0.441	80.4	0.387
Wan2.2-I2V-14B	**90.9**	**0.396**	**86.7**	**0.341**

Table 4: Component ablation study on moving object segmentation.

Variant	FBMS $\mathcal{J}\&\mathcal{F}\uparrow$	D17-M $\mathcal{J}\&\mathcal{F}\uparrow$
Full	**91.9**	**87.8**
w/o Real	86.6	77.0
w/o RGB	90.8	86.5
w/o MME	90.9	87.3

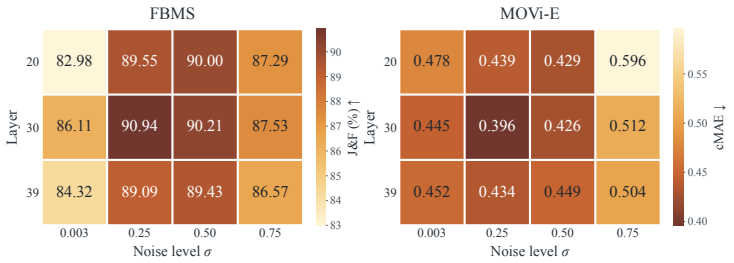

Figure 5: Joint layer–noise-level sweep for frozen Wan2.2 features: MOS on FBMS and MME on MOVi-E.

levels of Wan2.2 (Wan et al., 2025), evaluating MOS on FBMS and MME on MOVi-E. As shown in Figure 5, both tasks favor relatively low but nonzero noise levels, indicating that nearly noise-free inputs do not yield the most accessible motion representations. Across noise levels, the middle-to-late layer generally supports stronger readouts at both object and point levels than the other tested layers, with both readouts exhibiting similar layer and noise-level trends.

Component ablations. Table 4 examines the contributions of real-video supervision, the RGB encoder, and auxiliary motion-magnitude supervision to raw MOS predictions, without SAM 2 refinement or propagation. The synthetic-only variant retains strong performance on FBMS but shows a larger gap on DAVIS 2017-M. The synthetic-only model may also segment smoke and water splashes as moving foreground, even though these regions are not labeled as foreground in the benchmark (Appendix D). Real-video supervision may help bridge both appearance differences and the mismatch between local motion cues and object-level annotations. The RGB encoder improves mask prediction on both datasets, suggesting appearance cues may help recover fine spatial details attenuated by latent compression and noise injection. Auxiliary 3D motion magnitude supervision provides a small but consistent benefit.

5 Conclusion

We investigated the potential of pretrained video generative representations for dense, camera-decoupled motion perception. By freezing the generative backbone and training only a lightweight feed-forward decoder, our framework jointly predicts moving object masks and fine-grained motion intensity, quantified as the magnitude of 3D scene flow in a globally aligned world coordinate system. Experiments show that these features outperform the evaluated image and geometry backbones on both intrinsic-motion perception tasks. Layer and noise-level sweeps reveal similar readout trends across both tasks, favoring relatively low but nonzero noise levels and middle-to-late transformer layers. Overall, these findings position pretrained video generative models as a promising foundation for motion-centric perception beyond video synthesis.

AI USE STATEMENT

We used ChatGPT and Claude to discuss research ideas, assist with method and experimental design, and support the implementation of the proposed methods through code generation and debugging. We also used these tools for translation, language polishing, drafting passages, searching for and summarizing literature, and creating figures. The authors reviewed the manuscript sentence by sentence, checked references against the original sources, manually verified derivations, inspected and tested the code, and conducted experiments to validate the implementation. All AI-assisted contributions were reviewed by the authors, who take full responsibility for the final content of this work, including its text, methods, code, and results.

REPRODUCIBILITY STATEMENT

We describe our method in Section 3 and the experimental setup in Section 4.1. Appendix A details the synthetic data generation and annotation processing procedures. Appendix B provides the feature extraction settings, decoder architecture, loss implementation, and training configuration. Appendix C specifies the motion-magnitude normalization, evaluation-time scale alignment, metrics, and evaluation protocols. We will release the implementation, trained checkpoints, and evaluation scripts upon acceptance.

REFERENCES

Erwin Coumans and Yunfei Bai. PyBullet, a Python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google Scanned Objects: A high-quality dataset of 3D scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE, 2022.

Haiwen Feng, Junyi Zhang, Qianqian Wang, Yufei Ye, Pengcheng Yu, Michael J. Black, Trevor Darrell, and Angjoo Kanazawa. St4RTrack: Simultaneous 4D reconstruction and tracking in the world. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

Lily Goli, Sara Sabour, Mark Matthews, Marcus A. Brubaker, Dmitry Lagun, Alec Jacobson, David J. Fleet, Saurabh Saxena, and Andrea Tagliasacchi. RoMo: Robust motion segmentation improves structure from motion. *arXiv:2411.18650*, 2024.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3749–3761, 2022.

Xiankang He, Peile Lin, Ying Cui, Dongyan Guo, Chunhua Shen, and Xiaoqin Zhang. GeoMotion: Rethinking motion segmentation via latent 4D geometry, 2026. URL https://arxiv.org/abs/2602.21810.

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

Zixuan Huang, Xiang Li, Zhaoyang Lv, and James M. Rehg. How much 3D do video foundation models encode? *arXiv preprint arXiv:2512.19949*, 2025b.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. DynamicStereo: Consistent dynamic depth from stereo videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Seungwoo Kim, Khai Loong Aw, Klemen Kotar, Cristobal Eyzaguirre, Wanhee Lee, Yunong Liu, Jared Watrous, Stefan Stojanov, Juan Carlos Niebles, Jiajun Wu, and Daniel L. K. Yamins. Taming generative video models for zero-shot optical flow extraction, 2025. URL https://arxiv.org/abs/2507.09082.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE international conference on computer vision*, pp. 2192–2199, 2013.

Han Li, Si Liu, Zehao Huang, Dongxin Lyu, Longfei Xu, Jiahui Fu, Daxin Tian, Yuliang Xiu, and Naiyan Wang. ViSTR-Bench: Can MLLMs reason from continuous visual cues in dynamic scenes? *arXiv preprint arXiv:2607.20868*, 2026.

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 (CVPR)*, pp. 14582–14591, June 2023.

Yiqing Liang, Abhishek Badki, Hang Su, James Tompkin, and Orazio Gallo. Zero-shot monocular scene flow estimation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21031–21044, June 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Yihang Luo, Shangchen Zhou, Yushi Lan, Xingang Pan, and Chen Change Loy. 4RC: 4D reconstruction via conditional querying anytime and anywhere. *arXiv preprint arXiv:2602.10094*, 2026.

Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4981–4991, 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, 2023. doi: 10.1109/TPAMI.2022.3198480.

Jisu Nam, Soowon Son, Dahyun Chung, Jiyoung Kim, Siyoon Jin, Junhwa Hur, and Seungryong Kim. Emergent temporal correspondences from video diffusion transformers, 2025. URL https://arxiv.org/abs/2506.17220.

Peter Ochs, Jitendra Malik, and Thomas Brox. Segmentation of moving objects by long term video analysis. *IEEE transactions on pattern analysis and machine intelligence*, 36(6):1187–1200, 2014.

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.

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 (CVPR)*, June 2016.

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

Shenhan Qian, Ganlin Zhang, Shangzhe Wu, and Daniel Cremers. Flow4R: Unifying 4D reconstruction and tracking with scene flow, 2026. URL https://arxiv.org/abs/2602.14021.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12179–12188, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. SAM 2: Segment anything in images and videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026.

Ayush Shrivastava, Sanyam Mehta, Daniel Geng, and Andrew Owens. Point prompting: Counterfactual tracking with video diffusion models. *arXiv preprint arXiv:2510.11715*, 2025.

Soowon Son, Honggyu An, Chaehyun Kim, Hyunah Ko, Jisu Nam, Dahyun Chung, Siyoon Jin, Jung Yi, Jaewon Min, Junhwa Hur, and Seungryong Kim. Repurposing video diffusion transformers for robust point tracking, 2025. URL https://arxiv.org/abs/2512.20606v1.

Edgar Sucar, Zihang Lai, Eldar Insafutdinov, and Andrea Vedaldi. Dynamic point maps: A versatile representation for dynamic 3D reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7295–7305, October 2025.

Edgar Sucar, Eldar Insafutdinov, Zihang Lai, and Andrea Vedaldi. V-DPM: 4D video reconstruction with dynamic point maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=ypOiXjdfnU.

Pedro Vélez, Luisa F. Polanía, Yi Yang, Chuhan Zhang, Rishabh Kabra, Anurag Arnab, and Mehdi S. M. Sajjadi. From image to video: An empirical study of diffusion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16948–16958, October 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 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, 2025.

Letian Wang, Chuhan Zhang, Rishabh Kabra, Jasper Uijlings, Steven Waslander, Andrew Zisserman, Joao Carreira, Kaiming He, Misha Andriluka, Eduard Gabriel Bazavan, et al. Video generation models are general-purpose vision learners. *arXiv preprint arXiv:2607.09024*, 2026.

Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

Junyu Xie, Weidi Xie, and Andrew Zisserman. Segmenting moving objects via an object-centric layered representation. In *Advances in Neural Information Processing Systems*, 2022.

Junyu Xie, Weidi Xie, and Andrew Zisserman. Appearance-based refinement for object-centric motion segmentation. In *ECCV*, 2024a.

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, 2024b.

Junyu Xie, Tengda Han, Weidi Xie, and Andrew Zisserman. GMOS: Grounding moving object segmentation in 3D space and time. *arXiv preprint arXiv:2605.30352*, 2026.

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.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Tianyu Yuan, Yuanbo Yang, Lin-Zhuo Chen, Yao Yao, and Zhuzhong Qian. Denoise to track: Harnessing video diffusion priors for robust correspondence. *arXiv preprint arXiv:2512.04619*, 2025.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K. Barral, Raia Hadsell, Zoubin Ghahramani, Andrew Zisserman, Junlin Zhang, and Mehdi S. M. Sajjadi. Efficiently reconstructing dynamic scenes one D4RT at a time. *arXiv preprint arXiv:2512.08924*, 2025a.

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 *International Conference on Learning Representations*, volume 2025, pp. 82863–82886, 2025b.

Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8535–8546, 2025.

Ruijie Zhu, Jiahao Lu, Wenbo Hu, Xiaoguang Han, Jianfei Cai, Ying Shan, and Chuanxia Zheng. MotionCrafter: Dense geometry and motion reconstruction with a 4D VAE. *arXiv preprint arXiv:2602.08961*, 2026.

A Training Data and Annotation Processing

Table 5: Scale of the three training data sources. We report approximate annotated frame counts in the training splits, before temporal windowing and run-specific exclusions.

Dataset	Domain	Annotated frames	Supervision
MOVi-E	Synthetic	234K	MOS + MME
Kubric-Motion	Synthetic	23.5K	MOS + MME
GMOS-2K	Real	144K	MOS

A.1 Preprocessing Details

Kubric (Greff et al., 2022) directly provides RGB images, metric depth, instance segmentation, surface normals, normalized per-pixel object coordinates, optical flow, camera parameters, and per-frame rigid-object poses. We compute the camera intrinsics from the focal length, sensor width, and image resolution, and obtain the world-to-camera extrinsics by inverting the recorded camera poses. Depth and surface normals are used to identify invalid background regions and construct depth-confidence masks. For a pixel belonging to an object at frame t, its normalized object coordinate is converted to an object-local 3D point. Applying the object's poses at frames t and $t-1$ gives the world positions of the same physical surface point, $\mathbf{p}_t$ and $\mathbf{p}_{t-1}$. We define its 3D motion as

$$\mathbf{V}_{t-1\to t}(u,v) = \mathbf{p}_t(u,v) - \mathbf{p}_{t-1}(u,v). \tag{3}$$

This removes apparent motion caused only by camera movement. The first-frame motion target is set to zero and excluded from reconstruction supervision using a validity mask. To reduce scene-scale variation, we divide motion magnitudes by the mean camera-to-point distance of valid frame-0 points whose distances fall between the 10th and 40th percentiles. Moving-object masks are generated at the instance level. An instance is considered active in a frame if at least 30% of its valid pixels have normalized 3D motion magnitude above 0.005, or if the moving region contains a connected component of at least 100 pixels. An instance is classified as dynamic when it is active in at least three frames. All visible pixels of that instance are then labeled as moving throughout the clip, including frames in which it is temporarily stationary. Thus, the segmentation target represents clip-level moving-object identity, while the 3D magnitude target describes instantaneous previous-to-current motion.

A.2 Kubric-Motion Synthetic Pipeline

To enrich the motion and scene diversity of MOVi-E, we construct an additional synthetic motion dataset using Kubric (Greff et al., 2022), with rigid-body dynamics simulated by PyBullet (Coumans & Bai, 2016–2021) and images rendered using Blender Cycles. Each clip contains 24 frames at 12 fps. Foreground objects are sampled from Google Scanned Objects (Downs et al., 2022), and scene appearance is diversified using indoor and outdoor HDRI environments and procedurally generated floor materials. We randomize object identities, scales, initial positions, velocities, camera viewpoints, camera distances, and focal lengths. The generated clips include handheld camera motion, joint dolly and zoom, synchronized tracking shots, near-camera object intrusion, and foreground occlusion.

B Network Architecture and Training Details

B.1 Feature Extraction

All backbones remain frozen during feature extraction and training. For video diffusion models, we encode the input clip using the pretrained VAE, perturb its latents with Gaussian noise, and read the video-token activations after a selected transformer block in a single denoising forward pass. We use an empty text prompt throughout. Our default backbone is the low-noise expert of Wan2.2-I2V-14B, using layer $l = 30$ at $\sigma \approx 0.25$; its first-frame conditioning is constructed from the first causal VAE

latent and the corresponding mask. Wan2.1-T2V-1.3B uses layer $l = 22$ at the same noise level, without image conditioning. For CogVideoX-5B, we use layer $l = 32$ and native diffusion timestep 90, matching the noise-to-signal ratio of the Wan setting. Layer indices are zero-based, and these settings are fixed except in the layer and noise sweeps.

For DINOv2-L, we process frames independently and retain the final transformer block's patch tokens. For VGGT, we process the clip jointly and retain the final global-attention block's patch tokens, excluding camera and register tokens. These feature maps are bilinearly resampled to a stride-16 spatial grid while preserving the input field of view, matching the diffusion features' spatial layout. Temporally compressed diffusion features are restored to frame-level features by the temporal unpacker described below. For DINOv2 and VGGT, we instead use a comparable spatiotemporal CNN without temporal upsampling, keeping the trainable parameter counts similar.

B.2 Decoder Architecture

For the default Wan backbone, we prepend a copy of the first frame to each 24-frame clip, forming a 25-frame input that satisfies the $4N + 1$ temporal length shown in Fig. 2 ($N = 6$). This input produces seven temporal tokens. The temporal unpacker discards token 0, corresponding to the prepended first-frame copy, and applies LayerNorm and a linear projection to the remaining six tokens, followed by two temporal upsampling stages. Each stage uses a temporal convolution to expand the channels and rearranges the added channels into the temporal dimension, doubling the temporal length; this gives $6 \to 12 \to 24$ features aligned with the original 24 frames.

Factorized 3D residual blocks before and after upsampling mix features along the temporal and spatial dimensions separately. The restored features are refined by a four-layer per-frame ViT with 2D rotary positional embeddings.

A trainable CNN extracts features from each corresponding RGB frame and aligns them to the spatial grid of the video features. In each ViT block, a separate MLP maps the RGB features to scale and shift parameters that modulate the LayerNorm outputs of the self-attention and feed-forward sublayers. This RGB branch belongs to the trainable decoder and is separate from the backbone's first-frame image conditioning. Two DPT heads each fuse the intermediate features from all four refinement layers, producing MOS logits and motion magnitudes constrained to be nonnegative by ReLU, respectively.

B.3 Loss Implementation

Let A_i and $\hat{A}_i$ denote the ground-truth and predicted motion magnitudes at pixel i, respectively. We apply the log-space transform

$$g(a) = \log_{10}(\max(a, \epsilon)), \tag{4}$$

and separate valid pixels into high- and low-magnitude groups according to their ground-truth motion magnitudes. The log-space threshold $\tau = -2$ corresponds to $A_i > 10^{-2}$ in the high-magnitude group Ω_{h} and $A_i \leq 10^{-2}$ in the low-magnitude group Ω_{l}. Grouping depends on the motion magnitude over the current frame interval rather than MOS labels.

We average reconstruction errors within each group separately and sum the two averages, preventing the more numerous group from dominating the loss:

$$\mathcal{L}_{\mathrm{rec}} = \sum_{s \in \{\mathrm{h},\mathrm{l}\}} \frac{1}{|\Omega_s|} \sum_{i \in \Omega_s} \big(g(\hat{A}_i) - g(A_i)\big)^2, \tag{5}$$

with an empty-group term defined as zero.

Reconstruction uses valid annotations for $t \geq 1$. For clips with magnitude annotations, $\mathcal{L}_{\mathrm{reg}}$ averages squared predictions over all decoded pixels, including the auxiliary first-frame output, which receives no reconstruction supervision. MOS cross-entropy uses valid MOS labels; magnitude losses are omitted for samples without magnitude annotations.

B.4 Training Configuration

We use stratified sampling across training datasets, drawing 10,000 clips per epoch, corresponding to 1,250 optimization steps on eight GPUs with one clip per GPU. Mixed-data training samples

real and synthetic clips at an approximately 1:1 ratio; synthetic-only training samples exclusively from Kubric-Motion and MOVi-E. We use AdamW with learning rate 10^{-4} and weight decay 0.05, with two epochs of linear warmup followed by cosine annealing over a total of 15 epochs. The experiments in Table 3 and Figure 5 use the same learning-rate schedule but stop after eight epochs. We set $\epsilon = 10^{-4}$, $\lambda_{\text{reg}} = 0.01$, $\lambda_{\text{mos}} = 10$, and $\lambda_{\text{mme}} = 1$ for joint MOS–MME training.

C Motion Magnitude Normalization and Evaluation

C.1 Training-target Normalization

To construct normalized magnitude targets, we recover reference-frame 3D points from depth and camera intrinsics and compute their distances to the reference camera center. The scene scale d_c is the mean of valid distances between the 10th and 40th percentiles, shared across all frames and objects in the clip. This normalization is used only to construct training targets and does not require depth at inference.

C.2 Evaluation-time Alignment and Metrics

Let $\mathcal{V}_c$ be the valid evaluation samples in clip c, and let p_{ci} and g_{ci} denote predicted and ground-truth motion magnitudes. We fit one nonnegative scale using all valid samples in the clip:

$$s_c = \arg\min_{s \geq 0} \sum_{i \in \mathcal{V}_c} (s p_{ci} - g_{ci})^2. \tag{6}$$

The scale is shared across frames and objects, so evaluation measures scale-aligned magnitude rather than absolute metric calibration. Using $\tilde{p}_{ci} = s_c p_{ci}$, the clip-level normalized error and threshold error rate are

$$e_c = \frac{\sum_{i \in \mathcal{V}_c} |\tilde{p}_{ci} - g_{ci}|}{\sum_{i \in \mathcal{V}_c} g_{ci}}, \qquad e_{15}^{(c)} = \frac{1}{|\mathcal{V}_c|} \sum_{i \in \mathcal{V}_c} \mathbf{1}\left[|\tilde{p}_{ci} - g_{ci}| > 0.15 \max_{j \in \mathcal{V}_c} g_{cj} \right]. \tag{7}$$

The normalization and threshold are shared across the clip, not defined relative to each pixel's magnitude. Within each sequence, cMAE averages e_c equally over clips, whereas e_{15} pools predictions exceeding the threshold over all valid samples. Dataset scores then average valid sequence scores equally. We report e_{15} as a percentage with two decimal places; lower values are better for both metrics. Clips with negligible ground-truth motion are excluded from cMAE to avoid near-zero normalization; sequences without a valid cMAE score are omitted from that average.

C.3 Evaluation Protocol

We use the validation splits of MOVi-E and Dynamic Replica, and the publicly annotated training split of Spring. For Dynamic Replica, we obtain world-space displacements from the provided 3D trajectories and evaluate only at valid annotated points.

Ours and 4RC use backward motion, while MotionCrafter uses its native forward motion. We evaluate the same adjacent frame pairs against direction-specific ground truth and validity masks; visibility changes mean that the evaluated points may differ between directions. All methods use the same scale-alignment and aggregation rules.

D Discussion and Limitations

Motion predictions and benchmark annotations. Although our synthetic training data contain only rigid-body object motion, the synthetic-only model also segments water splashes and smoke as moving foreground, as shown in Figure 6. These regions move but are not labeled as foreground in the benchmark. The mixed-data model largely excludes them, producing masks that better match the annotations. These examples suggest that the lower MOS scores of the synthetic-only model may partly reflect what the benchmark labels as foreground, rather than a failure to detect motion.

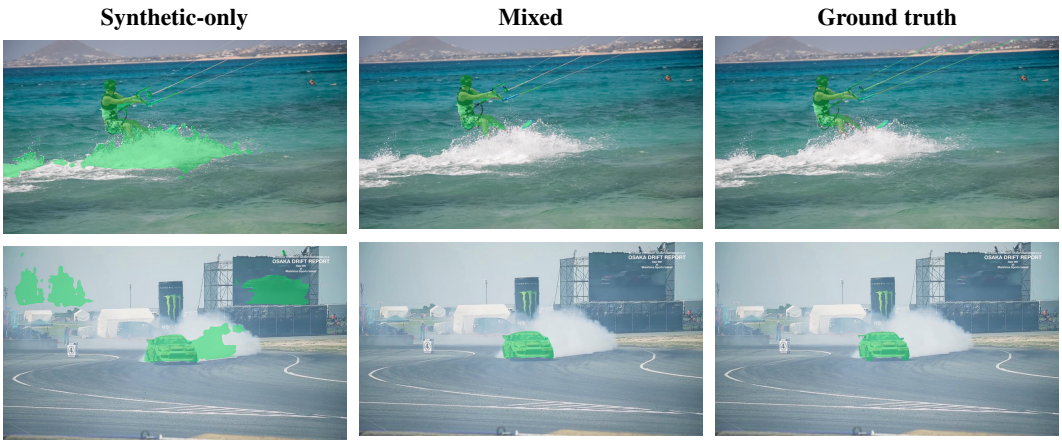

Figure 6: MOS predictions and annotations on DAVIS: *kite-surf*, frame 00024 (top), and *drift-chicane*, frame 00039 (bottom). Both models are shown without SAM post-processing. The synthetic-only model also segments water splashes and smoke outside the annotated object regions.

E Additional Qualitative Results

Figures 7 and 8 provide additional comparisons of 3D motion magnitude on Dynamic Replica and real-world DAVIS videos, illustrating localized motion responses and differences in background responses.

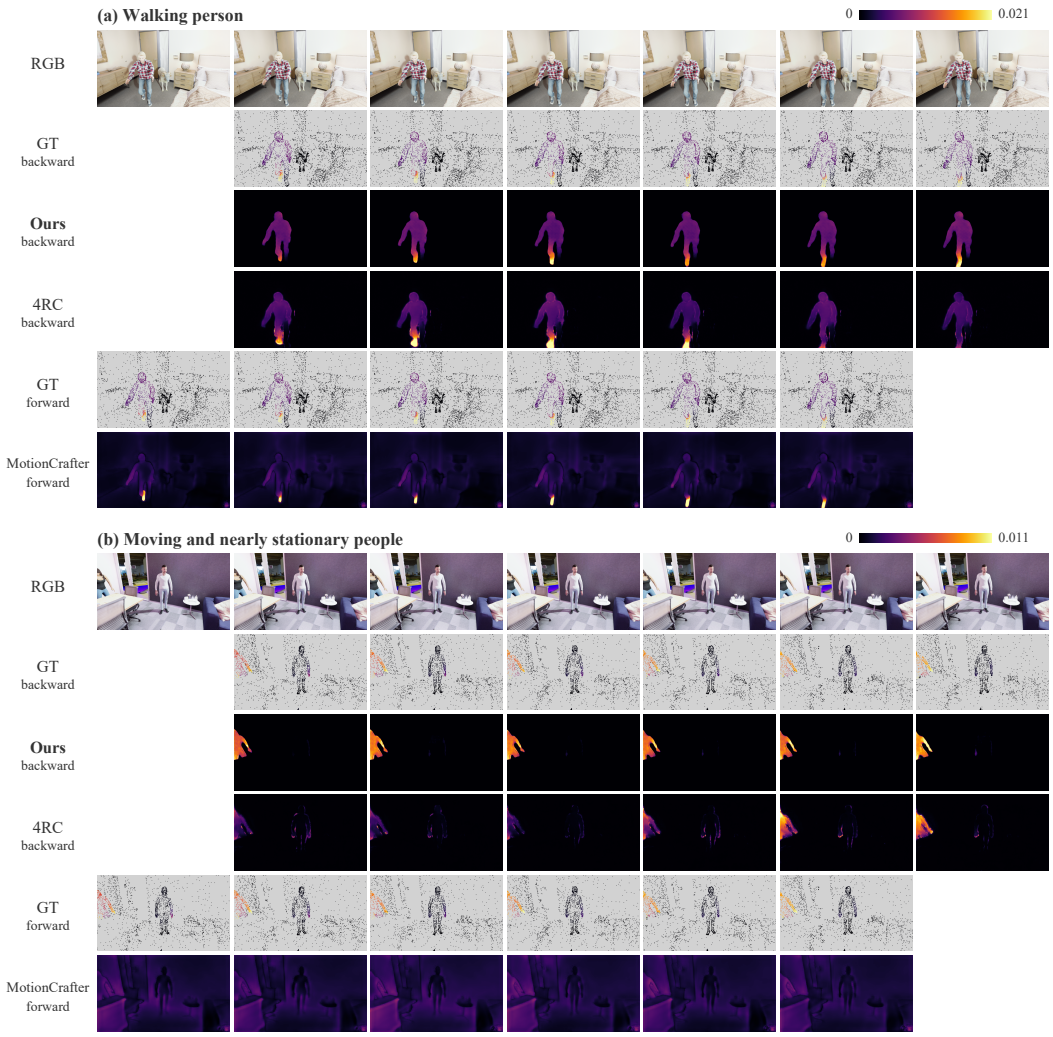

Figure 7: Additional 3D motion magnitude comparisons on Dynamic Replica. Each example shows seven consecutive frames. Ours and 4RC predict backward motion, while MotionCrafter predicts native forward motion; each is shown with its corresponding ground truth (GT). GT consists of sparse trajectory samples, enlarged for visibility without dense interpolation; predictions are displayed densely. We apply the clip-level scale alignment used in Table 2 and use a shared color scale across methods and frames within each example. Gray in the GT rows denotes unannotated regions, and blank cells omit motion intervals outside the displayed window. In these examples, the baselines exhibit residual responses on static backgrounds or nearly stationary people. Ours more selectively localizes motion in space and time while maintaining low responses on static scene regions under camera movement.



Figure 8: Additional 3D motion magnitude comparisons on real-world videos. Ours and 4RC predict backward motion, while MotionCrafter predicts forward motion; blank cells omit intervals outside the displayed window. Without 3D ground truth, predictions are normalized independently for visualization. The baselines exhibit residual responses on static backgrounds. Ours more selectively localizes motion in moving body parts while maintaining low background responses under camera movement.