Easi3R: Estimating Disentangled Motion from DUSt3R Without Training

Supplementary Material

In this **supplementary document**, we first present additional method details on temporal consistency dynamic object segmentation in Appendix A. Next, we conduct ablation studies of Easi3R in Appendix B and analysis limitations in Appendix C. Lastly, we report additional qualitative results in Appendix D. We invite readers to easi3r.github.io for better visualization.

A. Dynamic Object Segmentation

We have presented dynamic object segmentation for a single frame in Section 3.3, now we introduce how to ensure consistency along the temporal axis. Given image feature tokens $\mathbf{F}_0^t$ for frames at t, output from the image encoder, we concatenate them along the temporal dimension,

$$\bar{\mathbf{F}} = [\mathbf{F}_0^1; \mathbf{F}_0^2; \ldots; \mathbf{F}_0^T] \in \mathbb{R}^{(T\times h\times w)\times c} \tag{12}$$

where c is the feature dimension of the tokens. This allows us to apply k-means clustering to group similar features across frames, producing cluster assignments,

$$C = \text{KMeans}(\bar{\mathbf{F}}, k), \quad C^t(x,y) \in \{1,\ldots,k\}, \ \forall t,x,y \tag{13}$$

where k is the number of clusters, we use $k = 64$ for all experiments.

For each cluster $c \in \{1,\ldots,k\}$, we compute a dynamic score s_c by averaging the base dynamic attention values of all tokens within that cluster:

$$s_c = \frac{\sum_t \sum_{i,j} \mathbb{1}[C^t(x,y) = c] \cdot \mathbf{A}^{t=\text{dyn}}(x,y)}{\sum_t \sum_{x,y} \mathbb{1}[C^t(x,y) = c]} \tag{14}$$

where $\mathbb{1}[\cdot]$ denotes the indicator function. We then use these scores to generate a cluster-fused dynamic attention map, mapping each pixel's cluster assignment back to its corresponding dynamic score,

$$\mathbf{A}_{\text{fuse}}^{t=\text{dyn}}(x,y) = s_{C^t(x,y)} \tag{15}$$

The refined dynamic attention map $\mathbf{A}_{\text{fuse}}^{t=\text{dyn}} \in \mathbb{R}^{h\times w}$ is used to infer the dynamic object segmentation by,

$$\mathbf{M}^t(x,y) = \mathbb{1}[\mathbf{A}_{\text{fuse}}^{t=\text{dyn}}(x,y) > \alpha] \tag{16}$$

where α is an automatic image thresholding using Otsu's method [36]. This refinement enforces temporal consistency by ensuring similar features across frames receive consistent dynamic scores, as shown in Figure 8. The resulting dynamic object segmentation is further utilized in the second inference pass and global optimization.

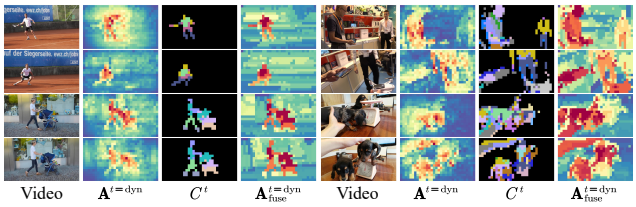


Figure 8. **Benefits of Cross-frame Feature Clustering.** We visualize the dynamic attention map $\mathbf{A}^{t=\text{dyn}}$, cluster assignments C^t, and cluster-fused dynamic attention map $\mathbf{A}_{\text{fuse}}^{t=\text{dyn}}$. Features from the DUSt3R encoder exhibit temporal consistency, as cluster assignments (C^t) remain unchanged across frames, thereby enhancing temporal consistency in dynamic segmentation ($\mathbf{A}^{t=\text{dyn}}$fuse) through clustering-guided temporal fusing. For better visual intuition, we invite readers to easi3r.github.io.

Table 6. **Ablation of Dynamic Object Segmentation** on DAVIS.

Backbone	Ablation	DAVIS-16		DAVIS-17		DAVIS-all	
		JM↑	JR↑	JM↑	JR↑	JM↑	JR↑
DUSt3R	w/o $\mathbf{A}_{\mu}^{a=\text{src}}$	45.1	45.2	42.8	39.9	42.2	38.5
	w/o $\mathbf{A}_{\sigma}^{a=\text{src}}$	42.3	50.0	35.0	37.0	30.9	28.3
	w/o $\mathbf{A}_{\mu}^{a=\text{ref}}$	33.3	28.4	31.5	27.9	32.5	29.7
	w/o $\mathbf{A}_{\sigma}^{a=\text{ref}}$	47.7	54.1	46.2	54.3	43.7	48.6
	w/o Clustering	40.0	38.5	38.3	38.3	34.3	30.5
	Full	**53.1**	**60.4**	**49.0**	**56.4**	**44.5**	**49.6**
MonST3R	w/o $\mathbf{A}_{\mu}^{a=\text{src}}$	47.2	51.5	44.4	46.7	40.9	41.5
	w/o $\mathbf{A}_{\sigma}^{a=\text{src}}$	49.7	60.1	48.7	57.8	44.9	49.6
	w/o $\mathbf{A}_{\mu}^{a=\text{ref}}$	46.4	54.0	47.4	55.9	45.3	50.7
	w/o $\mathbf{A}_{\sigma}^{a=\text{ref}}$	50.7	62.6	51.0	60.2	50.3	56.8
	w/o Clustering	45.5	46.7	45.3	48.1	42.1	43.5
	Full	**57.7**	**71.6**	**56.5**	**68.6**	**53.0**	**63.4**

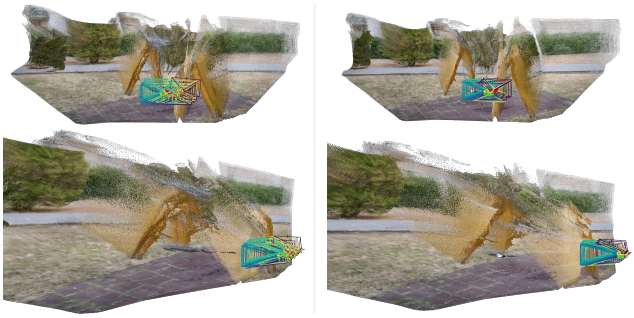
B. Ablation Study

Our ablation lies in two folds: dynamic object segmentation and 4D reconstruction. For dynamic object segmentation, as shown in Table 11 we ablate the contribution of four aggregated temporal cross-attention maps, $\mathbf{A}_{\mu}^{a=\text{src}}$, $\mathbf{A}_{\sigma}^{a=\text{src}}$, $\mathbf{A}_{\mu}^{a=\text{ref}}$, $\mathbf{A}_{\sigma}^{a=\text{ref}}$, and feature clustering. The ablation results show that (1) Disabling any temporal cross-attention map leads to a performance drop, indicating that all attention maps contribute to improved dynamic object segmentation; and (2) Features from the DUSt3R encoder exhibit temporal consistency and enhance dynamic segmentation through cross-frame clustering.

Table 7 presents ablation studies on 4D reconstruction, evaluating two key design choices: (1) the impact of two-branch re-weighting (applying attention re-weighting to both reference and source decoders) and (2) global alignment using optical flow with and without segmentation. The

Table 7. **Ablation Study of Camera Pose Estimation and Point Cloud Reconstruction** on the DyCheck dataset.

Backbone	Re-weighting	Flow-GA	Pose Estimation			Reconstruction					
			ATE↓	RTE↓	RRE↓	Accuracy↓ Mean	Accuracy↓ Median	Completeness↓ Mean	Completeness↓ Median	Distance↓ Mean	Distance↓ Median
DUSt3R	Ref + Src	✗	0.030	0.026	1.777	0.775	0.596	1.848	0.778	0.342	0.224
	Ref	✗	0.029	0.025	1.774	0.772	0.596	1.813	0.757	0.336	0.219
	Ref	w/o Mask	0.026	0.017	1.472	0.940	0.831	1.654	0.685	0.336	0.220
	Ref	w/ Mask	**0.021**	**0.014**	**1.092**	**0.703**	**0.589**	**1.474**	**0.586**	**0.301**	**0.186**
MonST3R	Ref + Src	✗	0.040	0.032	1.751	0.848	0.744	1.850	1.003	0.398	0.292
	Ref	✗	0.038	0.032	1.736	0.846	0.660	1.840	0.983	0.390	0.290
	Ref	w/o Mask	0.033	0.023	1.495	0.969	0.796	1.752	0.998	0.368	0.273
	Ref	w/ Mask	**0.030**	**0.021**	**1.390**	**0.834**	**0.643**	**1.661**	**0.916**	**0.350**	**0.255**



DUSt3R [64] Ours

Figure 9. **Limitation.** We visualize static reconstructions from two different viewpoints in the top and bottom rows. Easi3R improves camera pose estimation and point cloud reconstruction (top row), enhancing alignment in structures like swing supports through attention re-weighting and segmentation-aware global alignment. However, from another viewpoint (bottom row), Easi3R still produces floaters near object boundaries.

ablation results show that (1) Re-weighting only the reference view decoder outperforms re-weighting both branches. Since the reference and source decoders serve different roles, and the reference view acts as the static standard, this aligns with our design intuition (i); and (2) Incorporating segmentation in global alignment consistently improves 4D reconstruction quality.

C. Limitations

Despite strong performance on various in-the-wild videos, Easi3R can fail when the DUSt3R/MonST3R backbones produce inaccurate depth predictions. While Easi3R effectively improves camera pose estimation and point cloud reconstruction, as shown in Table 5 of the main paper, it provides clear improvements in completeness and distance metrics, which are measured on the global point cloud. However, a noticeable gap remains in depth accuracy, which

Video Easi3R MonST3R Video Easi3R MonST3R

Figure 10. **Dynamic masks in static scene.** Easi3R tends to reweight low confident regions in static scenes, leading to better pose estimation, as shown in Tab. 9.

Alignment	Method	Sintel Abs Rel ↓	Sintel δ<1.25 ↑	BONN Abs Rel ↓	BONN δ<1.25 ↑	KITTI Abs Rel ↓	KITTI δ <1.25 ↑
Per-sequence scale & shift	Marigold	0.532	51.5	0.091	93.1	0.149	79.6
	DepthAnythingV2	0.367	55.4	0.106	92.1	0.140	80.4
	NVDS	0.408	48.3	0.167	76.6	0.253	58.8
	ChronoDepth	0.687	48.6	0.100	91.1	0.167	75.9
	DepthCrafter	**0.292**	**69.7**	0.075	97.1	0.110	88.1
	Robust-CVD	0.703	47.8	-	-	-	-
	CasualSAM	0.387	54.7	0.169	73.7	0.246	62.2
	MASt3R	0.327	59.4	0.167	78.5	0.137	83.6
	Spann3R	0.508	50.8	0.157	82.1	0.207	73.0
	CUT3R	0.454	55.7	0.074	94.5	0.106	88.7
	DUSt3R	0.531	51.2	0.156	83.1	0.135	81.8
	MonST3R	0.333	59.0	0.066	96.4	0.157	73.8
	Easi3R$_{\text{dust3r}}$	0.435	59.1	0.085	91.1	0.155	76.1
	Easi3R$_{\text{monst3r}}$	0.316	59.3	**0.057**	**97.2**	**0.092**	**90.6**
Per-sequence scale	MASt3R	0.641	43.9	0.252	70.1	0.183	74.5
	Spann3R	0.622	42.6	0.144	81.3	0.198	73.7
	CUT3R	0.421	47.9	0.078	93.7	0.118	88.1
	DUSt3R	0.656	45.2	0.155	83.3	0.144	81.3
	MonST3R	0.378	55.8	0.067	96.3	0.168	74.4
	Easi3R$_{\text{dust3r}}$	0.577	51.9	0.086	90.3	0.170	74.2
	Easi3R$_{\text{monst3r}}$	**0.377**	**55.9**	**0.059**	**97.0**	**0.102**	**91.2**

Table 8. **Video Depth Evaluation**. We use the evaluation results from CUT3R for baselines.

is evaluated on per-view outputs. This is because our method focuses mainly on improving dynamic regions and global alignment rather than correcting depth predictions in static parts, as illustrated in Figure 9. We leave per-view depth correction for future work.

D. Addtional Results

More Evaluation. We evaluated on full-length sequences with downsampling only for GPU fit, leading to more dynamic and challenging motion than prior short-clip settings. For a more general evaluation, we also include MonST3R and CUT3R evaluation protocols. Tab. 8 and Tab. 9 confirm the effectiveness.

Category	Method	Sintel			TUM-dynamics			ScanNet (static)		
		ATE ↓	RPE trans ↓	RPE rot ↓	ATE ↓	RPE trans ↓	RPE rot ↓	ATE ↓	RPE trans ↓	RPE rot ↓
Pose only	DROID-SLAM	0.175	0.084	1.912	-	-	-	-	-	-
	DPVO	0.115	0.072	1.975	-	-	-	-	-	-
	Particle-SfM	0.129	**0.031**	**0.535**	-	-	-	0.136	0.023	0.836
	LEAP-VO	**0.089**	0.066	1.250	0.068	0.008	1.686	**0.070**	**0.018**	**0.535**
Pose & Depth	Robust-CVD	0.360	0.154	3.443	0.153	0.026	3.528	0.227	0.064	7.374
	CasualSAM	0.141	**0.035**	**0.615**	0.071	**0.010**	1.712	0.158	0.034	1.618
	MASt3R	0.185	0.060	1.496	**0.038**	0.012	**0.448**	0.078	0.020	**0.475**
	Spann3R	0.329	0.110	4.471	0.056	0.021	0.591	0.096	0.023	0.661
	CUT3R	0.213	0.066	0.621	0.046	0.015	0.473	0.099	0.022	0.600
	DUSt3R	0.417	0.250	5.796	0.083	0.017	3.567	0.081	0.028	0.784
	MonST3R	0.111	0.044	0.869	0.098	0.019	0.935	0.077	0.018	0.529
	Easi3R$_{dust3r}$	0.402	0.098	0.876	0.134	0.017	1.077	0.067	0.018	0.670
	Easi3R$_{monst3r}$	**0.110**	0.042	0.758	0.105	0.022	1.064	**0.061**	**0.017**	0.525

Table 9. **Camera Pose Evaluation.** We use the evaluation results from CUT3R for baselines.

Supervision	Method	Input			Output		DAVIS-16		DAVIS-17	
		RGB	Optical Flow	Point Tracks	Mask	4D Reconstruction	JM↑	JM-M ↑	JM↑	JM-M ↑
Supervised	SFL	✓	✓		✓		67.4	-	-	-
	SIMO	✓	✓		✓		67.8	-	-	-
	OCLR-flow	✓	✓		✓		72.0	70.0	-	69.9
	OCLR-TTA	✓	✓		✓		80.8	80.2	-	76.0
	FlowSAM	✓	✓		✓		87.1	85.7	-	-
	SegAnyMo	✓		✓	✓		**90.6**	**89.2**	-	**90.0**
	DAS3R	✓			✓	✓	54.2	51.6	57.4	55.5
Unsupervised	SAGE	✓	✓		✓		42.6	-	-	-
	CUT	✓	✓		✓		55.2	-	-	-
	FTS	✓	✓		✓		55.8	-	-	-
	CIS	✓	✓		✓		70.3	67.6	-	-
	Motion Grouping	✓	✓		✓		68.3	-	-	-
	EM	✓	✓		✓		69.3	76.2	-	-
	RCF-Stage1	✓	✓		✓		80.2	78.6	-	-
	RCF-All	✓	✓		✓		82.1	**81.0**	-	-
	LRTL	✓	✓	✓	✓		**82.2**	-	-	-
Zero-shot	MonST3R	✓	✓		✓	✓	64.3	61.4	56.4	59.0
	Easi3R$_{dust3r}$	✓			✓	✓	67.9	67.4	60.1	62.0
	Easi3R$_{monst3r}$	✓			✓	✓	**70.7**	**71.1**	**67.9**	**67.7**

Table 10. **Comparisons of Dynamic Object Segmentation** on DAVIS with 2D dynamic segmentation methods.

Ablation	Variants	DAVIS-16		DAVIS-17		DAVIS-all	
		JM↑	JR↑	JM↑	JR↑	JM↑	JR↑
Window Size	3	76.0	89.2	70.8	82.4	65.7	76.2
	5*	70.7	79.9	67.9	76.1	63.1	72.6
	7	66.9	76.9	63.9	73.3	61.0	68.8
Number of Clusters	16	67.4	79.1	64.6	73.7	60.7	65.2
	32	71.6	83.9	68.0	78.3	65.1	75.2
	64*	70.7	79.9	67.9	76.1	63.1	72.6
	128	69.9	79.9	66.3	76.2	62.9	73.1
Thresholding Values	0.5	61.6	64.5	61.0	65.2	61.6	67.8
	0.7	70.2	85.0	62.8	71.8	58.1	67.0
	Otsu's method*	70.7	79.9	67.9	76.1	63.1	72.6

Table 11. **More ablations on segmentation quality using DAVIS.** * denotes the value used in the submission.

Behavior for Static Scenes. Interestingly, our method also improves on static scenes (ScanNet in Tab. 9 and Fig. 10), owing to our attention reweighting.

Comparison with 2D Baselines. We also include a comparison with 2D baselines. As shown in Tab. 10, Easi3R achieves SOTA segmentation in a zero-shot manner with only the image as input.

More Ablation. We further ablate the default settings - window size of 5, 64 clusters, and Otsu's method in Tab. 11. Using the recent segmentation method SegAnyMo, pose accuracy improves by 9.62% and depth accuracy improves by 4.11% on the Sintel dataset.

Runtime Our method runs at almost the same speed as MonST3R. MonST3R runs at 0.33 FPS, while Easi3R achieves 0.31 FPS for 512×144 image resolution on an NVIDIA RTX 4090 GPU.

Qualitative Results We report additional qualitative results of disentangled 4D reconstruction in Figure 11, Figure 12 and Figure 13. We find that MonST3R tends to predict under-segmented dynamic masks, while DAS3R tends to predict over-segmented dynamic masks. CUT3R, although it produces more accurate depth estimation, is prone to being affected by dynamic objects, leading to misaligned static structures, unstable camera pose estimation, and ghosting artifacts due to the lack of dynamic segmentation prediction. In contrast, Easi3R achieves more accurate segmentation, camera pose estimation, and 4D reconstruction, resulting in renderings with better visual quality.

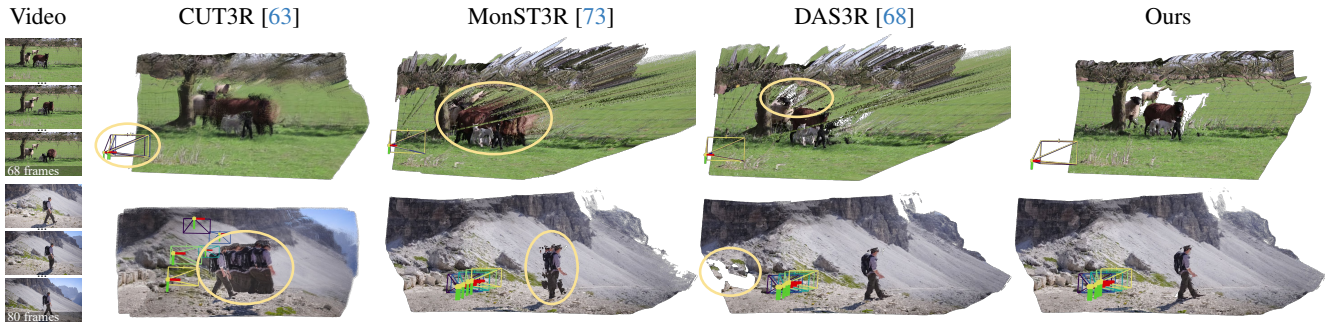


Figure 11. **Qualitative Comparison.** We visualize cross-frame globally aligned static scenes with dynamic point clouds at a selected timestamp. Notably, instead of using ground truth dynamic masks in previous work, we apply the estimated per-frame dynamic masks to filter out dynamic points at other timestamps for comparison. Top and bottom rows are Easi3R $_{\text{dust3r/monst3r}}$, respectively.

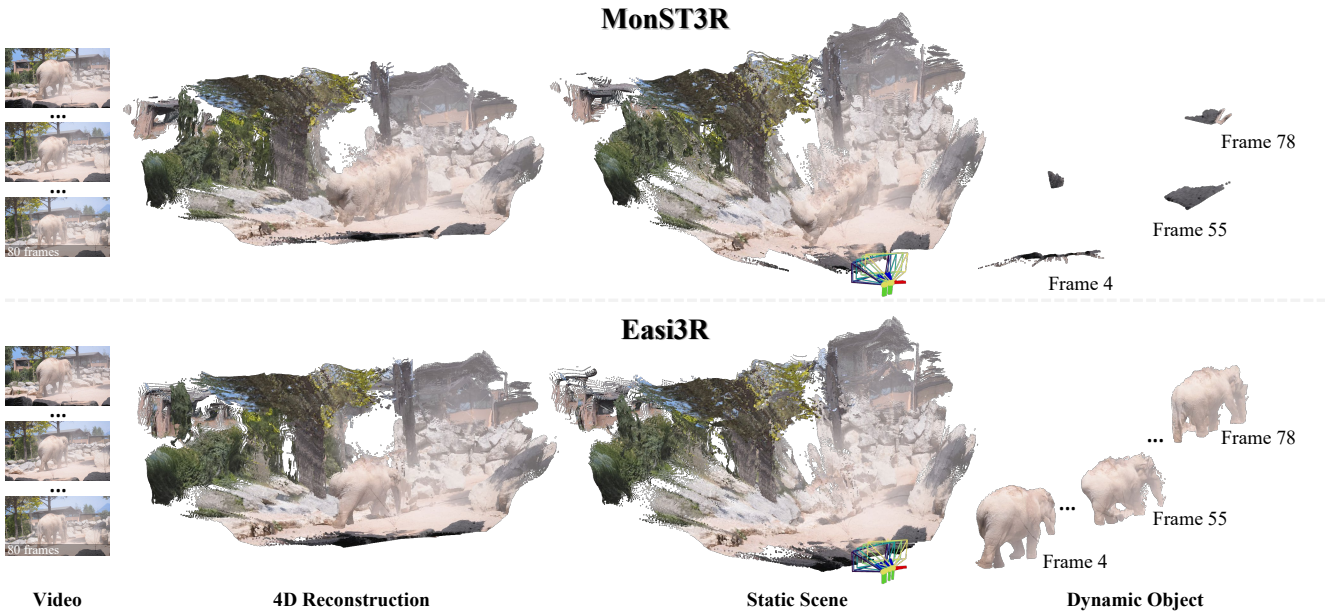


Figure 12. **Disentanglement vs. MonST3R [73].** We visualize the disentangled 4D reconstruction, static scene and dynamic objects at different frames. MonST3R tends to predict under-segmented dynamic masks.

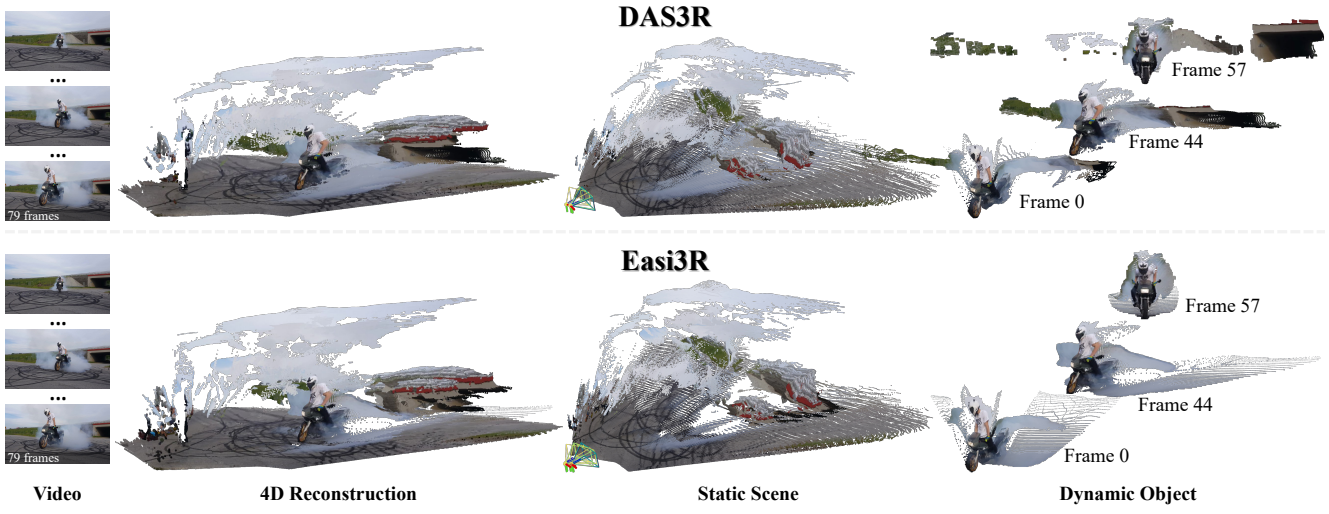


Figure 13. **Disentanglement vs. DAS3R [68].** We visualize the disentangled 4D reconstruction, static scene and dynamic objects at different frames. DAS3R tends to predict over-segmented dynamic masks.