2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Robust Multimodal Dynamic Object Segmentation

Zhe Xin[1*†], Hanzhi Chang[1*], Penghui Huang[1], Yinian Mao[1], Guoquan Huang[1,2]

***Abstract*— Dynamic object segmentation plays a critical role in many visual applications such as static scene reconstruction from dynamic videos. However, existing optical flow-based methods fail to ensure consistent static/dynamic segmentation along object boundaries, while 3D reconstruction-based approaches are highly sensitive to reconstruction errors. To address these limitations, we present a dynamic object segmentation framework that can generate both precise and complete dynamic masks by integrating multimodal cues including 2D point tracks, 3D reconstruction, and semantic information. We design a network combining Transformer architectures with feature clustering aggregation modules to perform static/dynamic classification of multimodal feature trajectories. It enables the model to adaptively determine which type of feature should dominate based on the characteristics of each scene, while also mitigating the impact of feature degradation. Additionally, we introduce a novel point-query-based SAM post-processing method capable of handling multiple objects within a single mask. Extensive experiments demonstrate that our approach achieves state-of-the-art performance in both dynamic object segmentation and static scene reconstruction tasks.**

I. Introduction

Dynamic object segmentation is essential in various domains including autonomous robotics, AR/VR, and video understanding [1]–[4]. This technique plays a crucial role in reconstructing static scenes from dynamic videos, as it enables the automatic filtering of moving objects while retaining static scene elements essential for 3D reconstruction. However, in real-world settings, due to complex object and camera motion patterns, environmental interferences, and variations in the scale of dynamic objects, dynamic object segmentation remains a challenging task.

Current work on dynamic object segmentation can be categorized into two paradigms: 2D optical flow based approaches and 3D-reconstruction-based techniques. 2D-based methods [5]–[8] typically estimate the optical flow from dynamic video sequences and perform static/dynamic classification at the pixel level. 3D reconstruction-based methods [9]–[11] first estimate depth maps or 3D point clouds from dynamic scenes, and subsequently derive dynamic masks through post-processing. The ViT architectures used in these methods can inherently capture semantic information, improving mask-object boundary alignment.

In order to generate both precise and complete dynamic object masks, relying solely on a single modality cue is often insufficient. For example, *2D optical flow or point tracking* directly reflects pixel-level movements that are intrinsically correlated with object motions. However, their pixel-wise nature fails to ensure consistent static/dynamic segmentation within object boundaries, and the lack of higher-dimensional information prevents the decoupling of camera and object motions. *3D reconstruction* provides approximate camera poses and 3D point map distributions, offering higher-dimensional perspective information for dynamic object segmentation. Nevertheless, its dynamic masks are highly sensitive to errors in depth estimation and 3D point map generation. Additionally, dynamic objects frequently violate epipolar constraints, leading to particularly unreliable 3D estimations for moving targets in certain scenarios. *Semantic information* serves as a valuable reference for maintaining segmentation consistency within objects. Current state-of-the-art methods often require SAM-based post-processing. However, they typically feed the entire mask as a single query to SAM, assuming all masked pixels belong to a single object, which is not always the case in real-world scenarios.

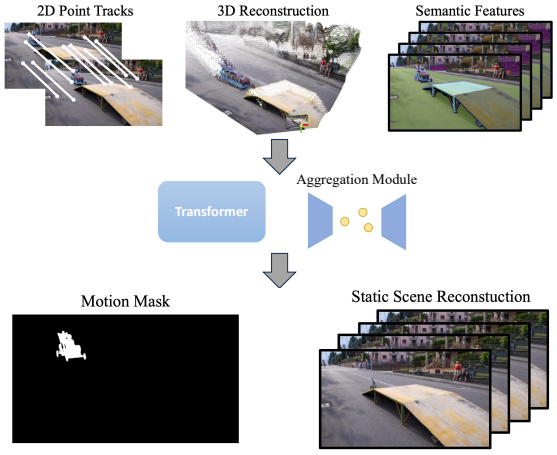

Fig. 1. **Overview.** Given a dynamic video containing several image frames, Our method leverages 2D point tracking, 3D reconstruction, and semantic priors as inputs, to accomplish the tasks of dynamic object segmentation and static scene reconstruction from dynamic videos.

To address the limitation of single-modal cues, in this paper, we design a unified network through the integration of optical flow, 3D reconstruction, and semantic information. Our approach takes semantic feature maps from SAM, 3D reconstruction results including depth maps, camera intrinsic/extrinsic parameters, and attention maps from MonST3R and 2D point tracks between multiviews as input, using feature trajectory classification paradigm to perform dynamic object segmentation, which includes a Transformer for multi-frame and multi-track information integration and a point-based aggregation network for feature clustering. To enhance

[1] Meituan UAV, Beijing, China. {xinzhe, changhanzhi, huangpenghui03, maoyinian}@meituan.com
[2] Dept. of Mechanical Engineering, Computer and Information Sciences, University of Delaware, Newark, DE, USA. ghuang@udel.edu
† denotes corresponding author.
* Authors contributed equally to this work.

mask quality in multi-object scenarios, we develop a SAM-enhanced post-processing technique that utilizes individual pixels rather than entire masks as SAM queries and iteratively refines initial masks. Experimental results demonstrate state-of-the-art performance in dynamic object segmentation. Furthermore, to validate the effectiveness of our method, we combine mask generation, pose estimation, and 3D Gaussian Splatting (3DGS) reconstruction to perform static scene reconstruction from dynamic videos. Experiments show superior performance compared to existing SOTA methods.

In summary, the main contributions of this paper include:

- We propose a dynamic object segmentation framework that integrates multimodal cues, including 2D point tracking, 3D reconstruction, and semantic information to achieve superior segmentation performance.
- We introduce a SAM-based post-processing methodology to address complex real-world dynamic scenarios involving multiple moving objects, significantly improving mask-based post-processing approaches.
- We conduct extensive validations demonstrating the state-of-the-art performance of the proposed method across multiple benchmarks in both the accuracy of motion masks and the quality of static scene reconstruction.

II. Related Work

Dynamic object segmentation focuses on predicting motion masks from video inputs. Traditional structure-from-motion methods, such as COLMAP, operate under the assumption that scenes are predominantly static, making them ineffective for videos containing dynamic objects. Alternative approaches [12]–[14] prioritize tracking and semantically segmenting dynamic objects but often rely on assumptions, such as distinguishing foreground object motion from a static background or pre-identifying mobile objects. These constraints limit their general applicability.

Classical methods typically employ optical flow estimation [7], [15], [16] and point tracking [17], [18] to separate moving objects from the background. To address these challenges, CasualSAM [19] jointly optimizes depth (using a pre-trained learned prior), camera poses, and motion masks. ParticleSfM [5] utilizes off-the-shelf optical flow and monocular depth estimators to create 3D tracks and trains a 3D motion classifier on synthetic data. Similarly, LEAP-VO [6] classifies tracks into static and dynamic components, enhances inputs with additional features, employs a refiner module, and uses a sliding window approach for global bundle adjustment to estimate camera poses. RoMo [20] incorporates unreliable epipolar geometry through Sampson error [21] and SAMv2 [22]. However, as these methods are trained exclusively on 2D data, they often struggle with issues such as imprecise optical flow, occlusions, and distinguishing object motion from camera motion, particularly in scenarios with faulty correspondences.

For point-map-based methods, DUSt3R [23] introduces a novel pose inference technique using a patch-based feed-forward network to predict global 3D coordinates. MonST3R [9] fine-tunes DUSt3R for dynamic scenes, integrating optical flow [24] with estimated pose and depth to achieve dynamic object segmentation. Building on this, DAS3R [11] trains a DPT [25] on top of MonST3R to enable feed-forward segmentation estimation. While effective, these approaches heavily rely on accurate pose and depth estimation to maintain reprojection consistency and require extensive training on diverse motion patterns to achieve robust generalization. Easi3R exploits attention layers from pre-trained MonST3R models to extract dynamic segmentation, it still cannot generalize to situations where MonST3R performs poorly, and fails to recover mask details without SAM-based post-processing due to the low resolution of attention maps.

III. Dynamic Object Segmentation with Multimodal Cues

Given a dynamic video clip containing N frames $\{I_i\}_{i=1}^N$, our goal is to estimate dynamic masks $\{M_i\}_{i=1}^N$ for each frame. Fig. 2 illustrates the overall system architecture. The primary components include multimodal feature extraction, dynamic mask estimation, and SAM-based post-processing. In the following, we will explain these modules in detail.

A. Multimodal Features Extraction

The framework effectively integrates 2D tracking, 3D reconstruction, and semantic information to achieve accurate dynamic mask predictions.

For 3D reconstruction, we employ MonST3R [9], which processes the entire video clip using its ViT architecture with global alignment to generate per-frame depth maps $\{D_i\}_{i=1}^N$, camera intrinsic parameters $\{K_i\}_{i=1}^N$, and camera extrinsic parameters $\{P_i|R_i,t_i\}_{i=1}^N$. Building upon insights from Easi3R [10] that the attention maps from the ViT decoder contain valuable information for identifying dynamic objects, we also compute and aggregate per-frame attention maps $\{A_i\}_{i=1}^N$. These features are then used in the feature trajectory construction process.

By extracting the self-attention maps and cross-attention maps from the ViT decoder layer, through averaging operations performed along query and layer dimensions, the corresponding attention maps are obtained for each image pair. Subsequently, we compute the mean and variance of attention maps across these pairs, and concatenate the results to generate the final attention map.

For semantic information, we utilize SAMv2 [22] to extract per-frame semantic feature maps $\{F_i\}_{i=1}^N$. For 2D tracking information, we sample 200×200 2D points from the first video frame and use them as queries into CoTracker [26] to obtain 2D tracking results $\{x_i^k\}_{i=1}^N$ for each pixel k.

B. Dynamic Mask Estimation

The multimodal features extracted in the preceding section are first converted into feature trajectories, which are then fed into the main model to classify each trajectory. These results are subsequently transformed into coarse dynamic

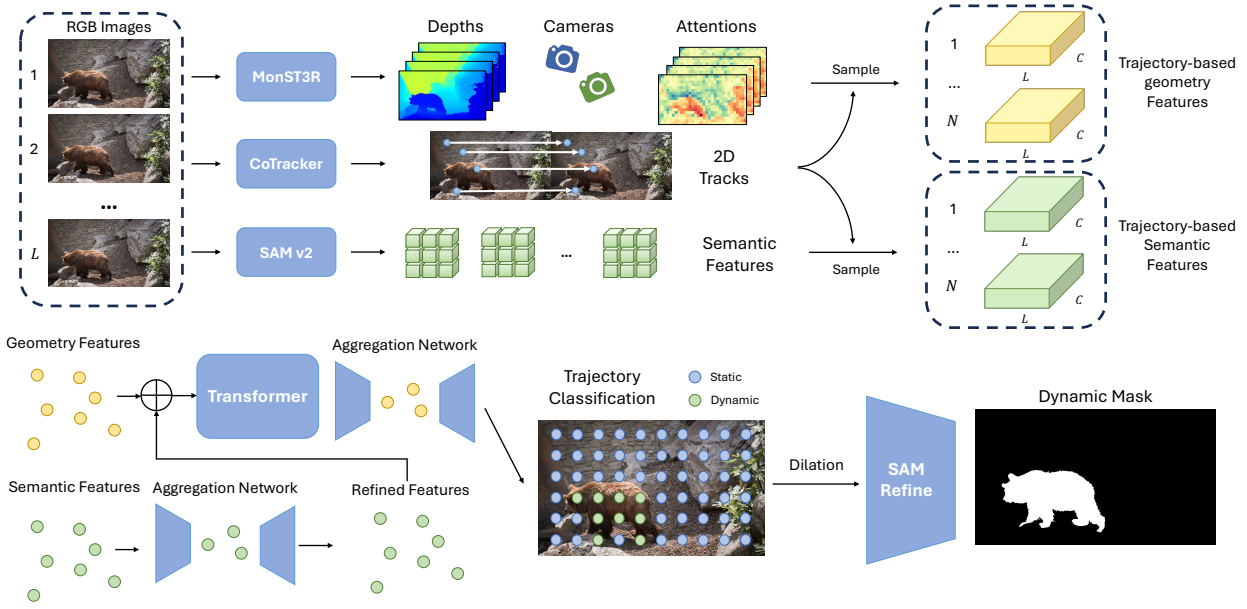

Fig. 2. **Overall Architecture.** Given a video clip with N frames, we first employ MonST3R [9] to perform coarse 3D reconstruction of the dynamic scene, obtaining depth information, camera parameters, and attention maps. Simultaneously, we use CoTracker [26] to densely sample 2D points on the images and obtain their motion trajectories across frames. These 2D point trajectories are then associated with corresponding 3D reconstruction features and semantic features are extracted from SAM [22] feature maps. These combined features are fed into our network consisting of a Transformer module and two aggregation networks. This network ultimately outputs static/dynamic classification results for each feature trajectory. To obtain the final high-quality dynamic masks, we apply our proposed SAM-based post-processing method to refine the initial dynamic masks.

masks using image dilation and refined by the SAM point-query-based iterative strategy.

1) Formulate Features Trajectories: We first convert input features into point-based trajectories, as this representation is more general and effectively captures information across the entire image sequence compared to two-frame flow models. For the k-th track $\{x_i^k\}_{i=1}^N$, we obtain per-frame depth values $\{d_i^k\}_{i=1}^N$ and attention values $\{a_i^k\}_{i=1}^N$ through interpolation, we then compute pixel position offsets Δx_i^k and depth offsets Δd_i^k between consecutive frames. Incorporating offset information provides a better representation of the changes between frames. Additionally, pixel position, depth, and offset are highly decoupled from the image content, which enhances the generalization capability of our method.

$$d_i^k, a_i^k = \text{Interpolation}(D_i, A_i) \tag{1}$$

$$\Delta x_i^k = x_{i+1}^k - x_i^k, \qquad \Delta d_i^k = d_{i+1}^k - d_i^k \tag{2}$$

Moreover, we employ camera parameters, including the focal length $f_i = (f_i^x, f_i^y)$ derived from intrinsic $\{K_i\}_{i=1}^N$ and extrinsics $\{P_i\}_{i=1}^N$. The transformation matrices $\{\Delta P_i | \Delta R_i, \Delta t_i\}_{i=1}^N$ between adjacent frames are also computed. Although camera information is inherently embedded in optical flow and depth, we find that explicitly providing these features as input improves performance, since this simplifies the learning difficulty for the model and enables more direct and effective utilization of camera motion.

The aforementioned features, such as 2D optical flow, depth, and camera poses, primarily describe the relationships of individual trajectories across images but fail to capture the consistency across multiple trajectories. Ignoring this consistency may lead to incomplete dynamic object segmentation. To address this limitation, we incorporate semantic features c_i^k obtained by interpolating the semantic feature maps. By combining coarse-grained attention features with fine-grained semantic features, our method achieves more accurate and complete dynamic masks. The final feature trajectories are formulated as follows, where $||$ means concatenation,

$$\mathbf{F}_{\text{traj}}^k = (x^k \,||\, \Delta x^k \,||\, d^k \,||\, \Delta d^k \,||\, c^k \,||\, a^k) \tag{3}$$

$$\mathbf{F}_{\text{sem}}^k = \text{Sample}(\{F_i\}_{i=1}^N) \tag{4}$$

2) Trajectory Classification: The semantic features of pixels belonging to the same object exhibit similar characteristics. To better capture pixel-wise relationships, we first utilize a point-based module Φ_{sem} to process and cluster semantic features $\mathbf{F}_{\text{sem}}^k$. This module clusters multiple feature trajectories using differentiable pooling operations and employs CNNs to facilitate interaction, thereby capturing global information in each cluster. The global information then aggregates with each local features in the cluster through unpooling operations to obtain the local-global context-aware features $\mathbf{F}_{\text{sem}}^{k'}$. This aggregation network unifies similar semantic features through feature clustering, providing meaningful semantic priors for subsequent processing.

Next, we concatenate $\mathbf{F}_{\text{traj}}^k$ with $\mathbf{F}_{\text{sem}}^{k'}$ and input them into a Transformer encoder to enable interaction in each trajectory and among all trajectories. These two types of

features are processed separately because $\mathbf{F}^k_{\text{traj}}$ primarily describes geometric properties with minimal correlation to image content, whereas $\mathbf{F}^{k'}_{\text{sem}}$ encodes texture attributes that are closely tied to image content.

Then, the refined features are fed into an aggregation network Φ acting as the decoder, where they are further clustered and fused. This process generates motion confidence scores for each trajectory, with the final classification determined via a sigmoid function. The entire procedure can be formally expressed as:

$$m^k = \text{Sigmoid}\left(\Phi\left(F^k_{\text{traj}} \,||\, \Phi_{\text{sem}}(F^k_{\text{sem}})\right)\right) \tag{5}$$

During training, ground truth labels m^k_{gt} are sampled from the ground truth dynamic mask of the first frame based on the 2D tracks x^k_1, and the cross-entropy loss is then computed as follows,

$$\mathcal{L} = \sum_k -m^k_{\text{gt}} \log(m^k) - (1 - m^k_{\text{gt}}) \log(1 - m^k) \tag{6}$$

Since the 2D tracks are sampled on a grid, during inference, after obtaining the motion confidence scores of trajectories, we apply a dilation operation to generate coarse dynamic masks.

C. SAM-based Post-processing

Recent dynamic segmentation methods [9]–[11] typically employ SAMv2 [22] to refine coarse dynamic masks. These approaches commonly input the entire mask as a query with a single object ID into SAMv2 to generate the final motion masks, which inherently assumes that all masked regions belong to the same object. However, this assumption is often violated in real-world scenarios with multiple dynamic objects, leading to incomplete or even entirely incorrect results.

Inspired by [27], we propose a point-query-based iterative SAM refinement strategy capable of handling multiple dynamic objects in a scene. The detailed procedure is outlined in Algorithm 1. Specifically, we first collect all pixels labeled as dynamic in the coarse mask M_{coarse} to form the dynamic pixel set X_{dyn}. Then, we randomly sample an individual pixel x as a point query for SAM. The resulting SAM-generated mask M_x is then evaluated based on its overlap ratio with M_{coarse}. If this ratio exceeds a predefined threshold β and the number of pixels in M_x meets the minimum requirement γ, we retain M_x in the final output M_{SAM} and remove all its constituent pixels from X_{dyn}. Otherwise, the sampled query pixel is eliminated from X_{dyn}. This iterative sampling continues until X_{dyn} is empty, after which the union of M_{SAM} and M_{coarse} is taken as the final dynamic mask M.

Algorithm 1: SAM-based Post-processing

Data: RGB Image I, Coarse Mask M_{coarse}, Overlapped Threshold β, Size Threshold γ
Result: Refined Final Dynamic Mask M

```
SAMSetImage(I);
X_dyn ← GetPixels(M_coarse);
M_SAM ← Zeros(M_coarse);
while Size(X_dyn) ≠ 0 do
    x ← RandomSample(X_dyn);
    M_x ← SAMPointQuery(x);
    if SumAll(M_x) ≥ γ then
        M_overlap ← Intersection(M_x, M_coarse);
        r ← SumAll(M_overlap) ÷ SumAll(M_x) ;
        if r ≥ β then
            M_SAM ← Union(M_SAM, M_x);
            X_overlap ← GetPixels(M_overlap);
            X_dyn ← RemovePixels(X_dyn, X_overlap);
        else X_dyn ← RemovePixels(X_dyn, x);
    else X_dyn ← RemovePixels(X_dyn, x);
M ← Union(M_SAM, M_coarse)
```

IV. Application to 3D Scene Reconstruction from Dynamic Videos

To demonstrate the effectiveness of the proposed multimodal dynamic object segmentation, we apply it to the problem of static 3D scene reconstruction from dynamic videos. In this problem, the input video is first converted into multiple video clips of length N using a sliding window approach. Each clip is then processed through a feature fusion network to obtain dynamic masks through feature extraction, trajectory classification and SAM-based post-processing. These dynamic masks are subsequently used to guide staticness-aware 3DGS reconstruction by filtering dynamic elements through static scene attributes. The rendering formulation of staticness-aware 3DGS is defined as follows,

$$\mathbf{c} = \sum_i c_i s_i \alpha'_i \prod_{j=1}^{i-1} (1 - s_j \alpha'_j) \tag{7}$$

$$\alpha'_i = \alpha_i \exp(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i)) \tag{8}$$

where $\mu_i,\ \Sigma_i,\ c_i,\ s_i,\ \alpha_i$ represents the position, covariance matrix, color, staticness, and opacity of the i-th 3DGS. The technical details of the 3DGS implementation follow the methodology described in DAS3R [11].

V. Experimental Results

To validate the proposed method, we conduct extensive experiments on both synthetic and real-world data.

Datasets. We evaluate our approach on the following three datasets that are widely used in the literature:

- PointOdyssey [28] is a synthetic dataset comprising 131 diverse indoor and outdoor scenes, with approximately 200k images. It includes dynamic objects, camera motion, and provides labels for camera poses and depth, making it well-suited for motion segmentation tasks.
- DAVIS2017 [29] consists of 90 videos featuring moving objects and cameras, which is relatively closer to the video sequences captured by robot cameras in the real world. Compared to its predecessor, DAVIS2016, this dataset introduces greater complexity by including multiple annotated objects per video, as well as more

challenging scenarios involving distractors, occlusions, smaller objects, and intricate structures.

- Sintel [30] is a synthetic dataset containing 23 sequences, renowned for its dynamic complexity. It presents highly challenging scenes with detailed motion patterns and interactions between dynamic and static elements, making it a valuable resource for evaluating motion segmentation methods.

Metrics. For dynamic object segmentation, we report accuracy, IoU, precision and recall metrics by comparing the predicted and ground-truth masks. For static scene reconstruction from dynamic videos, we report the PSNR and SSIM metrics by comparing the predicted and ground-truth RGB images.

Baselines. We benchmark our approach against the SOTA methods including the optical flow-based ParticleSfM (P-SfM) [5], end-to-end reconstruction based methods MonST3R [9] and Easi3R [10], and 3DGS-based DAS3R [11]. We use the official open-source implementations of these methods to evaluate the quality of motion masks. For reconstruction quality, we adopt the static background reconstruction with 3DGS in DAS3R and integrated the motion masks produced by different methods into it for comparison. For our methods, we use **Ours** to refer without any post-processing, **Ours(+SAM)** to refer the proposed point-query-based iterative SAM refinement strategy.

Implementation Details. We take video clips with $N = 10$ frames as input. In the main model, the Transformer we use consists of two encoder layers and two decoder layers, and the architecture of the aggregation networks we used based on OANet [31] with modifications. For the hyperparameters mentioned in the paper, we set $\alpha = 0.7$, $\beta = 0.3$, and $\gamma = 5$. We use the AdamW optimizer with a maximum learning rate of 0.001 and a batch size of 4 per GPU. The training stage is performed on $2\times$ A100-80G GPUs for 50 epochs with 1000 steps per epoch, using only the training split of the PointOdyssey dataset.

A. Motion Mask Accuracy

TABLE I. Motion Mask Accuracy of PointOdyssey dataset. **Acc** for accuracy, **Pre** for precision and **Rec** for recall in percentage.

Methods	Acc (mean)	IoU (mean)	Pre (mean)	Rec (mean)
P-SfM	88.50	27.39	56.59	34.99
MonST3R	85.27	18.61	59.59	22.11
Easi3R	88.17	40.01	59.45	57.49
DAS3R	92.22	66.54	76.62	79.64
Ours	93.70	63.37	73.45	81.42
Ours(+SAM)	93.69	64.38	73.79	82.23

1) Comparison on PointOdyssey: Detailed comparison results are shown in Tab. I. Except for ParticleSfM, all other methods were trained on this dataset. Our approach demonstrates superior performance in terms of accuracy and recall. Compared to ParticleSfM, which primarily relies on optical flow, and MonST3R, which mainly depends on depth estimation, our method effectively integrates 2D and 3D information, enabling it to better handle scenarios where both objects and the camera are in motion. Additionally, our approach remains robust even when one type of feature degrades, ensuring consistent motion recognition results.

Easi3R, which extracts attention layer features from MonST3R, suffers from accuracy limitations due to its dependence on implicit reconstruction precision. Furthermore, its lower resolution results in less accurate object contours. In contrast, we leverage SAM to extract more explicit semantic information and utilize feature clustering to achieve more comprehensive identification of dynamic objects. Combined with post-processing, our approach further enhances the refinement of object contours.

DAS3R builds on MonST3R by adding a motion mask head and training with pre-trained models using 3DGS. However, our method achieves superior results with significantly fewer training steps, demonstrating greater efficiency and effectiveness.

TABLE II. Motion Mask Accuracy of DAVIS2017 dataset. **Acc** for accuracy, **Pre** for precision and **Rec** for recall in percentage.

Methods	Acc (mean)	IoU (mean)	Pre (mean)	Rec (mean)
P-SfM	90.21	37.21	58.87	53.65
MonST3R	87.77	41.57	58.45	50.47
Easi3R	88.64	50.69	61.25	78.03
DAS3R	86.96	44.83	48.79	83.01
Ours	92.89	53.67	59.72	86.76
Ours(+SAM)	94.01	59.77	62.22	93.14

2) Comparison on DAVIS: The comparison results of DAVIS2017 are shown in Tab. II. DAVIS2017 is a dataset composed of real-world scenes, where our method consistently outperforms others across all metrics, demonstrating excellent generalization capabilities. By converting absolute information such as pixel positions, depth, and poses into relative representations like optical flow, scene flow, and camera motion, and representing them using point trajectories, our approach effectively reduces overfitting to specific scenes. Qualitative visualizations of the motion masks are shown in Fig. 3.

TABLE III. Motion Mask Accuracy of Sintel dataset. **Acc** for accuracy, **Pre** for precision and **Rec** for recall in percentage.

Methods	Acc (mean)	IoU (mean)	Pre (mean)	Rec (mean)
P-SfM	79.55	27.10	59.69	31.76
MonST3R	72.97	30.18	53.34	34.57
Easi3R	80.96	37.51	62.98	48.44
DAS3R	81.06	53.36	58.17	82.63
Ours	87.33	46.04	65.99	57.11
Ours(+SAM)	89.99	54.66	67.43	67.97

3) Comparison on Sintel: The accuracy of motion mask estimation is compared in Tab. III. On the Sintel dataset, the performance of all methods is suboptimal. While DAS3R achieves relatively high recall, its other metrics are notably low, suggesting a trade-off between quality for quantity. In contrast, our method demonstrates a more balanced performance across metrics. The MPI Sintel dataset is characterized by simple camera movements, unrealistic appearances, generally textureless scenes, and minimal 3D structure once

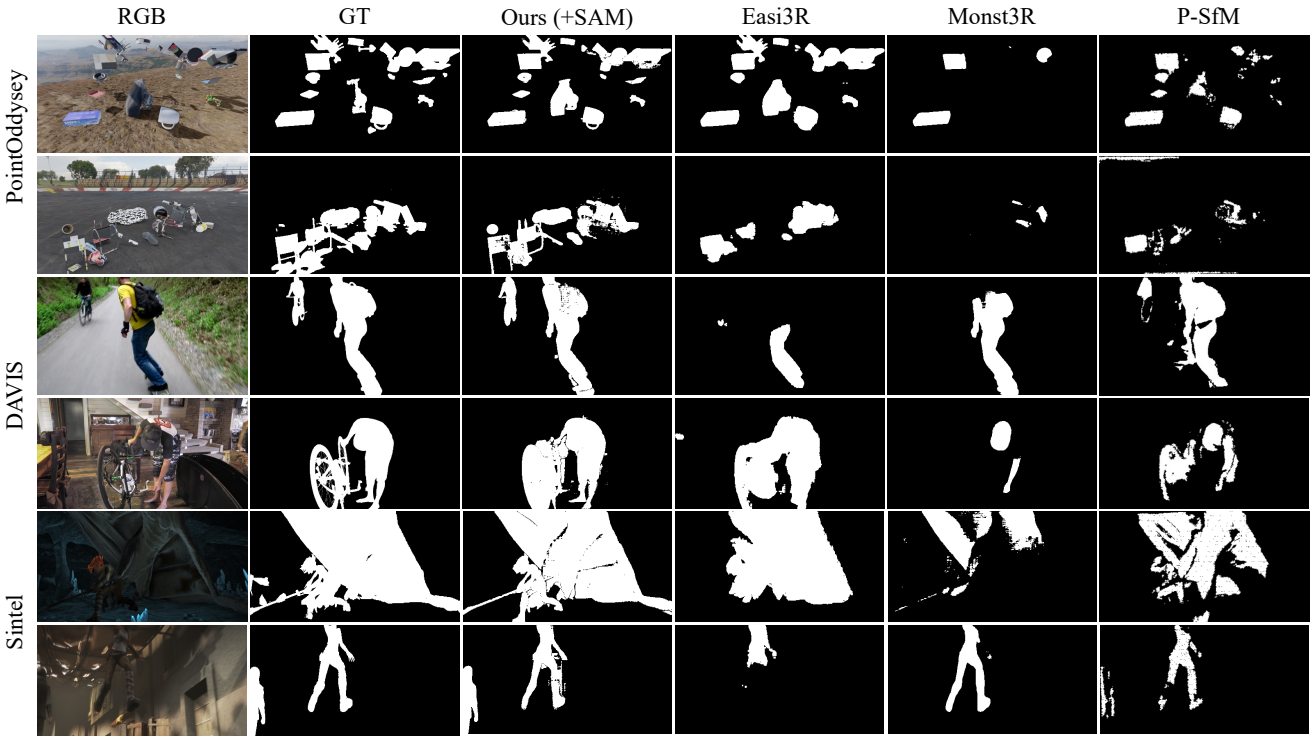

Fig. 3. **Visual Comparison of Dynamic Mask Prediction.** We provide quantitative comparisons for dynamic mask prediction tasks. Our method produces more precise and complete masks compared to other approaches.

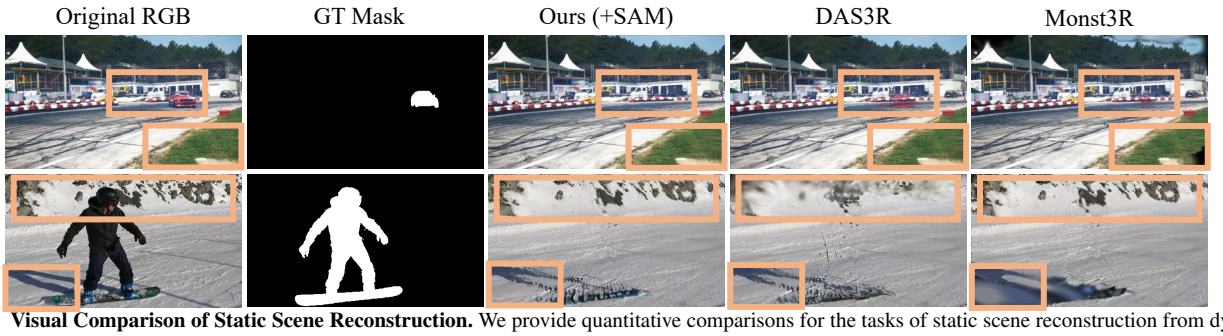

Fig. 4. **Visual Comparison of Static Scene Reconstruction.** We provide quantitative comparisons for the tasks of static scene reconstruction from dynamic videos using the DAVIS2017 dataset. Our method achieves more photorealistic novel view synthesis results compared to other methods, demonstrating its effectiveness in dynamic object segmentation.

dynamic objects are removed. These factors make scene reconstruction highly prone to failure. Moreover, the dataset frequently contains images dominated by dynamic objects, which presents significant challenges for dynamic object segmentation.

B. Static Scene Reconstruction

TABLE IV. Static scene reconstruction quality of DAVIS2017 dataset. The DAS3R split includes: blackswan, camel, car-shadow, dog, horsejump-high, motocross-jump, parkour, soapbox

Methods	DAS3R split		All	
	PSNR	SSIM	PSNR	SSIM
MonST3R	27.19	0.863	26.88	0.874
Easi3R	29.14	0.905	27.53	0.882
DAS3R	28.40	0.896	27.28	0.871
Ours(+SAM)	**30.60**	**0.933**	**28.90**	**0.903**

We follow the reconstruction pipeline of DAS3R to obtain camera poses, depth, motion masks, and other relevant information, which are then used as inputs to train a 3DGS model for static scenes. Using the camera poses from the training views, we render images without dynamic objects. The DAVIS dataset provides ground-truth motion masks, which we use as a baseline. To evaluate the quality of static scene reconstruction, we compare the rendered images obtained by training 3DGS with motion masks from other methods against those rendered using ground truth motion masks. Detailed quantitative comparison results are shown in Tab. IV. Our method achieves highest PSNR and SSIM on both the DAS3R split and all scenes of the DAVIS2017 dataset. We also provide qualitative comparisons in Fig. 4. The rendered images generated by our method exhibit fewer artifacts and more accurately capture and remove dynamic objects. Furthermore, the background is rendered with high clarity, indicating that the dynamic foreground and static

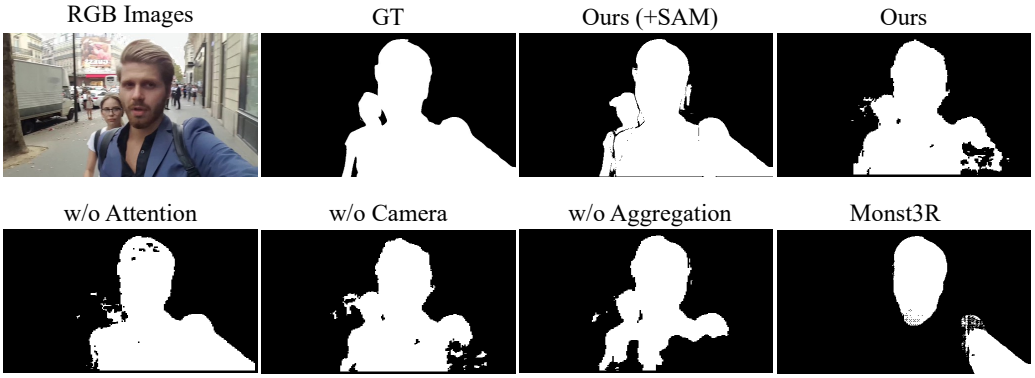

Fig. 5. **Visual Comparison in the Ablation Study.** We conduct ablation study on different feature combinations and processing strategies. Our full model achieves the best performance among all variants.

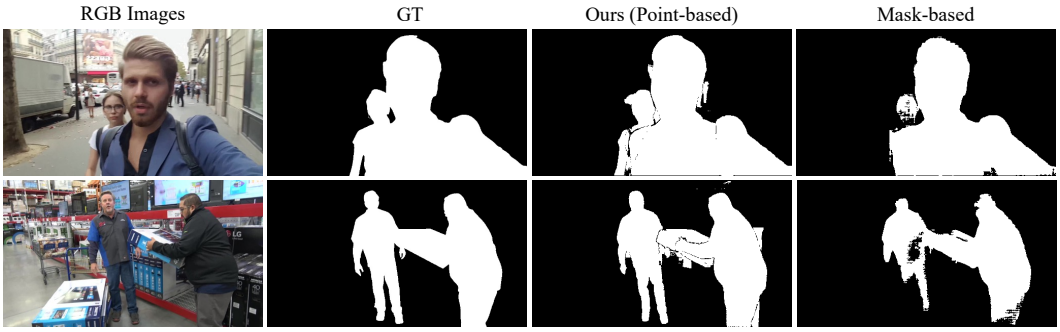

Fig. 6. **Comparison of Different SAM refinement strategy.** We show comparisons between ours and mask-based SAM refinement strategy. Previous mask-based refinement strategy suffers from incomplete mask generation due to the assumption that dynamic pixels belong to a single object.

background have been precisely and effectively separated.

C. Ablation Study

TABLE V. Motion Mask Accuracy of DAVIS2017 dataset with different feature combinations and processing strategy.

Features				Acc	Rec
Camera	Attention	Aggregation	SAM	mean	mean
✘	✔	✔	✘	92.11	84.22
✔	✘	✔	✘	87.67	68.20
✔	✔	✘	✘	92.51	84.46
✔	✔	✔	✘	92.89	86.76
✔	✔	✔	✔	94.01	93.14

We evaluate our method by testing different feature combinations on the motion mask accuracy, i.e., camera pose embeddings, attention features similar to Easi3R, feature aggregation modules and SAM refinements. Tab. V and Fig. 5 report the results tested on DAVIS2017. Attention features are directly extracted from MonST3R layers and camera poses are further optimized by bundle adjustment based on depths and confidences inferred from MonST3R. Although pose information is inherently embedded in optical flow and depth, explicitly providing poses as input features to the model still proves beneficial. This simplifies the learning process for the model and allows for more direct and effective utilization of camera motion. After clustering and aggregating the point track features with our aggregation module, the model can adaptively identify which type of feature should dominate based on the characteristics of each scene, while also mitigating the impact of feature degradation, such as imprecise optical flow and depth estimation. Moreover, SAM refinement can help the predicted mask better aligned with the object boundaries, resulting in a higher recall rate.

We also provide a comparison of different SAM refinement strategies, as shown in Fig. 6. It is evident that without SAM refinement, the predicted masks are incomplete, resulting in a lower recall rate. If the mask-based SAM refinement is applied, due to the assumption that all masked pixels belong to a single object, this strategy may not be suitable for all cases in the real world, and the output predicted masks may suffer from poor shapes.

D. Efficiency Analysis

We present the model sizes and inference time of our method and baseline models in Tab. VI. Compared with state-of-the-art (SOTA) methods, our approach features more compact model size and shorter inference time. We also provide the average inference times of the mask-based SAM refinement strategy used in previous works and our point-based refinement strategy. It is worth noting that the number of loops in our method is directly determined by the number of masked pixels in the coarse dynamic mask, therefore, the inference time may vary depending on the input video clips.

VI. Conclusions and Future Work

We propose a novel dynamic object segmentation framework by unifying multimodal cues, including 2D point tracks, 3D reconstruction results, and semantic features.

TABLE VI. Model Efficiency Analysis.

Model	Params (M)	Inference Time (s)
P-SfM	0.5377	14.0724
MonST3R	571.171	87.0266
Easi3R	571.171	37.0712
DAS3R	611.868	27.5518
Ours	2.798	0.2833
Mask-based SAM Refine	-	2.6565
Ours (Full Sequence)	-	32.4399

This framework enables the model to adaptively determine which type of feature should dominate based on the characteristics of each scene, while also mitigating the impact of feature degradation. Moreover, we introduce a novel point-query-based SAM post-processing method capable of handling multiple objects within a single mask. Extensive experiments demonstrate the effectiveness of our model. However, despite strong performance on various datasets, our method occasionally encounters challenges in scenarios where dynamic objects dominate the image. Addressing this limitation through more diverse training data and enhanced model refinements will be a key focus of our future work.

Acknowledgement. This work is supported by the Shenzhen Science andTechnology Program under Grant Nos. KJZD20230923115210021.

REFERENCES

[1] J. H. Hammer, M. Voit, and J. Beyerer, "Motion segmentation and appearance change detection based 2d hand tracking," in *2016 19th International Conference on Information Fusion (FUSION)*. IEEE, 2016, pp. 1743–1750.

[2] J. Klappstein, T. Vaudrey, C. Rabe, A. Wedel, and R. Klette, "Moving object segmentation using optical flow and depth information," in *Pacific-Rim symposium on image and video technology*. Springer, 2009, pp. 611–623.

[3] H. Rashed, A. El Sallab, S. Yogamani, and M. ElHelw, "Motion and depth augmented semantic segmentation for autonomous navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[4] C. Wang, B. Luo, Y. Zhang, Q. Zhao, L. Yin, W. Wang, X. Su, Y. Wang, and C. Li, "Dymslam: 4d dynamic scene reconstruction based on geometrical motion segmentation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 550–557, 2020.

[5] W. Zhao, S. Liu, H. Guo, W. Wang, and Y.-J. Liu, "Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild," in *European Conference on Computer Vision*. Springer, 2022, pp. 523–542.

[6] W. Chen, L. Chen, R. Wang, and M. Pollefeys, "Leap-vo: Long-term effective any point tracking for visual odometry," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 844–19 853.

[7] C. Yang, H. Lamdouar, E. Lu, A. Zisserman, and W. Xie, "Self-supervised video object segmentation by motion grouping," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7177–7188.

[8] T. Zhou, J. Li, S. Wang, R. Tao, and J. Shen, "Matnet: Motion-attentive transition network for zero-shot video object segmentation," *IEEE transactions on image processing*, vol. 29, pp. 8326–8338, 2020.

[9] J. Zhang, C. Herrmann, J. Hur, V. Jampani, T. Darrell, F. Cole, D. Sun, and M.-H. Yang, "Monst3r: A simple approach for estimating geometry in the presence of motion," *arXiv preprint arXiv:2410.03825*, 2024.

[10] X. Chen, Y. Chen, Y. Xiu, A. Geiger, and A. Chen, "Easi3r: Estimating disentangled motion from dust3r without training," *arXiv preprint arXiv:2503.24391*, 2025.

[11] K. Xu, T. H. E. Tse, J. Peng, and A. Yao, "Das3r: Dynamics-aware gaussian splatting for static scene reconstruction," *arXiv preprint arXiv:2412.19584*, 2024.

[12] B. Bescos, J. M. Fácil, J. Civera, and J. Neira, "Dynaslam: Tracking, mapping, and inpainting in dynamic scenes," *IEEE robotics and automation letters*, vol. 3, no. 4, pp. 4076–4083, 2018.

[13] L. Xiao, J. Wang, X. Qiu, Z. Rong, and X. Zou, "Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment," *Robotics and Autonomous Systems*, vol. 117, pp. 1–16, 2019.

[14] P. Ungermann, A. Ettenhofer, M. Nießner, and B. Roessle, "Robust 3d gaussian splatting for novel view synthesis in presence of distractors," in *DAGM German Conference on Pattern Recognition*. Springer, 2024, pp. 153–167.

[15] L. Lian, Z. Wu, and S. 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*, 2023, pp. 14 582–14 591.

[16] J. Xie, W. Xie, and A. Zisserman, "Segmenting moving objects via an object-centric layered representation," *Advances in neural information processing systems*, vol. 35, pp. 28 023–28 036, 2022.

[17] L. Karazija, I. Laina, C. Rupprecht, and A. Vedaldi, "Learning segmentation from point trajectories," *Advances in Neural Information Processing Systems*, vol. 37, pp. 112 573–112 597, 2024.

[18] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *European conference on computer vision*. Springer, 2010, pp. 282–295.

[19] Z. Zhang, F. Cole, Z. Li, M. Rubinstein, N. Snavely, and W. T. Freeman, "Structure and motion from casual videos," in *European Conference on Computer Vision*. Springer, 2022, pp. 20–37.

[20] L. Goli, S. Sabour, M. Matthews, M. Brubaker, D. Lagun, A. Jacobson, D. J. Fleet, S. Saxena, and A. Tagliasacchi, "Romo: Robust motion segmentation improves structure from motion," *arXiv preprint arXiv:2411.18650*, 2024.

[21] P. D. Sampson, "Fitting conic sections to "very scattered" data: An iterative refinement of the bookstein algorithm," *Computer graphics and image processing*, vol. 18, no. 1, pp. 97–108, 1982.

[22] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[23] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 697–20 709.

[24] Y. Wang, L. Lipson, and J. Deng, "Sea-raft: Simple, efficient, accurate raft for optical flow," in *European Conference on Computer Vision*. Springer, 2024, pp. 36–54.

[25] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 179–12 188.

[26] N. Karaev, I. Makarov, J. Wang, N. Neverova, A. Vedaldi, and C. Rupprecht, "Cotracker3: Simpler and better point tracking by pseudo-labelling real videos," in *Proc. arXiv:2410.11831*, 2024.

[27] Y. Tang, Y. Guo, D. Li, and C. Peng, "Spars3r: Semantic prior alignment and regularization for sparse 3d reconstruction," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 26 810–26 821.

[28] Y. Zheng, A. W. Harley, B. Shen, G. Wetzstein, and L. J. Guibas, "Pointodyssey: A large-scale synthetic dataset for long-term point tracking," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 19 855–19 865.

[29] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 724–732.

[30] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black, "A naturalistic open source movie for optical flow evaluation," in *European conference on computer vision*. Springer, 2012, pp. 611–625.

[31] J. Zhang, D. Sun, Z. Luo, A. Yao, L. Zhou, T. Shen, Y. Chen, L. Quan, and H. Liao, "Learning two-view correspondences and geometry using order-aware network," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5845–5854.