Long-RVOS: A Comprehensive Benchmark for Long-term Referring Video Object Segmentation

Supplementary Material

Table 7. Definitions of the video attributes.

Attribute	Full Name	Definition
POC	Partial Occlusion	The target object is partially occluded in the sequence.
FOC	Full Occlusion	The target object is fully occluded in the sequence.
OV	Out-of-view	The target leaves the video frame completely.
LRA	Long-term Reappearance	Target object reappears after disappearing for at least 100 frames.
VC	View Change	Viewpoint affects target appearance significantly.
ARC	Aspect Ratio Change	The ratio of bounding box aspect ratio is outside the range [0.5, 2].
SV	Scale Variation	The ratio of any pair of bounding-box is outside of range [0.5,2.0].
CM	Camera Motion	Abrupt motion of the camera.
MB	Motion Blur	The boundary of target object is blurred because of camera or object fast motion.

Table 8. The percentage (%) of videos featuring specific attributes.

Dataset	POC	FOC	OV	LRA	VC	ARC	SV	CM	MB
MeViS [9]	54.8	15.1	28.7	0.1	10.0	88.2	78.7	49.2	18.8
Long-RVOS (Ours)	**60.5**	**36.2**	**61.0**	**11.5**	**25.9**	**96.2**	**93.6**	**60.7**	**28.7**

7. More Dataset Statistics

To further highlight the challenges posed by Long-RVOS, we present a statistical analysis of video attributes, with definitions provided in Table 7. As shown in Table 8, compared to the current largest dataset MeViS [9], Long-RVOS involves numerous long-video challenges, including frequent object occlusion (POC, FOC, and OV) and long-term object disappearance-reappearance (LRA). In addition, the videos in Long-RVOS exhibit significant object motion (CM and MB) and appearance changes (VC, ARC and SV), making the dataset more akin to real-world scenarios.

8. More Implementation Details

Motion Extraction. Following Video-LaVIT [20], we rely on motion vectors instead of the expensive dense optical flow. Formally, given a video clip, we partition each frame into 16×16 non-overlapping macroblocks. Then, motion vectors $\vec{m}$ of the t-th frame are estimated by matching the macroblocks between the adjacent frames I_t and I_{t-1}:

$$\vec{m}(p,q) = \arg\min_{i,j} \|I_t(p,q) - I_{t-1}(p-i, q-j)\|, \quad (6)$$

where $I(p,q)$ denotes the pixel values of the macroblock at location (p,q), and (i,j) denotes the coordinate offset between the centers of the two macroblocks. Empirically, the extraction of motion vectors runs at 748 FPS on our device (Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz), enabling real-time processing of long videos.

Global Interaction. This module performs temporal attention over the inter-frame object features, enabling temporal reasoning and understanding. Since this is a common module in RVOS approaches [9, 27, 32], we follow the object-consistent temporal enhancer (OTE) of ReferDINO [27] rather than designing a new one from scratch. For clarity, we briefly revist OTE here. Given T-frame object features $\{O_t\}_{t=1}^{T}$ where $O_t \in \mathbb{R}^{N_q \times d}$, OTE utilizes the Hungarian algorithm [24] to align the N_q objects between adjacent frames based on the pairwise cosine similarity. After that, it performs temporal self-attention over the aligned object features and cross-attention with the sentence features $\widetilde{E}$. We refer the readers to the original paper [27] for additional details.

Training. Unlike previous RVOS methods, ReferMo relies only on the keyframe ground-truth annotations for efficient training. Formally, given a text description and a video of T_c clips, ReferMo outputs the prediction sequences $\{\mathbf{p}_i\}_{i=1}^{N_q}$ for the N_q object queries, where each sequence $\boldsymbol{p}_i = \{\hat{\boldsymbol{s}}_i^t, \hat{\boldsymbol{b}}_i^t, \hat{\boldsymbol{m}}_i^t\}_{t=1}^{T_c}$ describes the binary classification probability, bounding box and mask of the i-th object query on t-th keyframe. Our training pipeline follows the practice in previous approaches [27, 32, 48]. Suppose $\boldsymbol{y} = \{\boldsymbol{s}^t, \boldsymbol{b}^t, \boldsymbol{m}^t\}_{t=1}^{T_c}$ as the ground truth of keyframes, we select the prediction sequence with the lowest matching cost as the positive and assign the remaining sequences as negative. The matching cost is defined as follows:

$$\begin{aligned}\mathcal{L}_{\text{total}}(\boldsymbol{y}, \boldsymbol{p}_i) =& \lambda_{\text{cls}}\mathcal{L}_{\text{cls}}(\boldsymbol{y}, \boldsymbol{p}_i) + \lambda_{\text{box}}\mathcal{L}_{\text{box}}(\boldsymbol{y}, \boldsymbol{p}_i) \\ &+ \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}(\boldsymbol{y}, \boldsymbol{p}_i).\end{aligned} \quad (7)$$

Table 9. Comparison on Long-RVOS valid set. FPS is estimated at 360P on Nvidia A6000 GPUs, excluding the video loading time.

Method	Static			Dynamic			Hybrid			Overall			FPS
	$\mathcal{J}\&\mathcal{F}$	tIoU	vIoU	$\mathcal{J}\&\mathcal{F}$	tIoU	vIoU	$\mathcal{J}\&\mathcal{F}$	tIoU	vIoU	$\mathcal{J}\&\mathcal{F}$	tIoU	vIoU	
Without SAM / SAM2													
SOC [32] NeurIPS'23	38.7	73.1	34.9	37.8	74.6	34.1	37.8	74.3	34.5	38.1	74.0	34.5	53.8
MUTR [52] AAAI'24	44.1	73.5	40.3	42.0	75.2	38.9	43.5	74.6	40.2	43.2	74.4	39.8	20.4
ReferDINO [27] ICCV'25	52.5	**74.2**	48.2	46.7	**75.2**	42.9	49.3	**74.8**	45.4	49.6	**74.7**	45.6	46.4
With SAM / SAM2													
VideoLISA [1] NeurIPS'24	17.3	66.8	12.7	12.9	72.6	6.8	12.1	72.3	6.0	14.1	70.5	8.6	6.6
GLUS [28] CVPR'25	24.4	62.8	20.8	26.1	64.7	23.1	24.1	63.5	20.6	24.8	63.7	21.5	3.6
SAMWISE [7] CVPR'25	42.3	61.5	31.2	40.7	63.3	31.4	40.6	65.8	31.2	41.2	63.5	31.2	7.0
RGA3 [45] ICCV'25	21.1	61.0	15.4	22.3	62.8	17.5	21.1	61.8	16.4	21.5	61.8	16.4	8.7
ReferMo (Ours)	**56.7**	74.0	**49.4**	**50.7**	74.2	**43.4**	**53.7**	74.7	**47.4**	**53.7**	74.3	**46.8**	52.5

The matching cost is computed on individual frames and normalized by T_c. Here, $\mathcal{L}_{\text{cls}}$ is the focal loss that supervises the binary classification prediction. $\mathcal{L}_{\text{box}}$ sums up the L1 loss and GIoU loss. $\mathcal{L}_{\text{mask}}$ is the combination of DICE loss, binary mask focal loss and projection loss [44]. λ_{cls}, λ_{box} and λ_{mask} are scalar weights of individual losses. The model is optimized end-to-end by minimizing the total loss $\mathcal{L}_{\text{total}}$ for positive sequences and only the classification loss $\mathcal{L}_{\text{cls}}$ for negative sequences. The input frames are resized to have the longest side of 640 pixels and the shortest side of 360 pixels during both training and evaluation.

Inference. ReferMo produces instance mask for the referring object on keyframes and then employs SAM2 [37] for subsequent frames. Specifically, for the prediction sequences $\{\mathbf{p}_i\}_{i=1}^{N_q}$, we select the best sequence with the highest average classification score as follows:

$$\sigma = \underset{i\in[1,N_q]}{\arg\max} \frac{1}{T_c} \sum_{t=1}^{T_c} \hat{\boldsymbol{s}}_i^t \tag{8}$$

Then, the output mask sequence on keyframes is formed as $\{\boldsymbol{m}_\sigma^t\}_{t=1}^{T_c}$. For the t-th video clip, we use the keyframe prediction $\boldsymbol{m}_\sigma^t$ as the mask prompt for SAM2, which tracks the target across the subsequent frames within the clip. We train ReferMo on Long-RVOS dataset for 6 epochs, which take 24 hours on 8 Nvidia A6000 GPUs.

9. Validation Results

In Table 9, we present the benchmark results on Long-RVOS validation set. The results show that our ReferMo achieves consistent improvements over previous RVOS methods, especially those built on SAM or SAM2.

10. More Ablation Studies

Effectiveness of Gating Image-Motion Fusion. ReferMo employs the spatial-aware gating (SG) and channel-aware gating (CG) mechanisms in image-motion fusion to avoid undesired motion noise. As shown in Table 10, directly concatenating keyframe and motion features leads to a performance collapse. This is because RVOS requires per-frame fine-grain perception, while directly integrating motion features can introduce significant object-irrelevant noise. By applying these two gating strategies, ReferMo effectively alleviates such noise while preserving only the motion cues that highlight target objects, thereby yielding significant performance gains.

Table 10. Keyframe results of different image-motion fusion approaches.

SG	CG	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
		28.8	28.7	28.7
✓		46.9	46.3	46.6
	✓	49.5	50.4	50.0
✓	✓	50.3	51.8	51.1

Table 11. Overall $\mathcal{J}\&\mathcal{F}$ results for different description lengths.

Method	**<10**	**[10, 20]**	**>20**
RGA3 [45]	23.8	22.6	19.8
MUTR [52]	42.5	43.1	38.2
SAMWISE [7]	40.1	41.5	40.8
ReferDINO [27]	49.2	49.2	44.3
ReferMo (ours)	**53.6**	**53.6**	**48.5**

Effect of Description Length. We evaluate the impact of varying description lengths and present the results in Table R1. As description length increases, slight performance declines are observed across models. However, our ReferMo consistently outperforms existing methods across different description lengths.

Effect of Multi-event Videos. To explore the impact of event number in a video on model performance, we categorized the samples into single-event, two-event, and multi-event groups based on the keywords (e.g., *then*, *finally*, *ultimately*) in descriptions. As shown in Table 12, per-

Figure 8. Qualitative comparison of our ReferMo with the SOTA model ReferDINO [27]. ReferMo performs better in long-term object consistency and segmentation quality.

Table 12. Overall $\mathcal{J}\&\mathcal{F}$ results by event complexity.

Method	*Single-event*	*Two-event*	*Multi-event*
RGA3 [45]	23.0	22.6	19.2
MUTR [52]	42.7	40.0	36.0
SAMWISE [7]	40.6	41.7	38.0
ReferDINO [27]	48.4	44.7	37.2
ReferMo (ours)	**52.9**	**48.0**	**40.5**

Table 13. $\mathcal{J}\&\mathcal{F}$, $\mathcal{J}$, and $\mathcal{F}$ results on short-term benchmark.

Method	Ref-DAVIS17			MeViS		
	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}$	$\mathcal{F}$
DsHmp [17]	64.9	61.7	68.8	46.4	43.0	49.8
ReferDINO [27]	68.9	65.1	72.9	49.3	44.7	53.9
ReferMo (ours)	**69.6**	**65.9**	**73.4**	**50.2**	**45.5**	**54.8**

formance across models declines as the event number increases, yet our ReferMo consistently outperforms existing methods. Also, these results highlight the significance of our long-term benchmark for evaluating the capabilities of RVOS models in understanding complex event sequences.

Performance on Short-term RVOS Benchmarks. We follow the evaluation protocols of ReferDINO [27] and report the results in Table 13. The results confirm that our method remains competitive on existing short-term benchmarks.

11. Visualization

In Figure 8, we provide the qualitative comparisons with the SOTA model ReferDINO [27] on Long-RVOS. These examples involve multiple long-term challenges, such as object occlusion, disappearance-reappearance and view changes. The results clearly show the effectiveness of ReferMo in temporal consistency and segmentation quality.

12. Limitations and Future Work

In this work, we chose to begin with description-based RVOS because it is commonly used in current video applications and this task remains far from being solved. It is promising to broaden the benchmark scope to support more tasks, such as reasoning RVOS [1, 51], semi-supervised VOS [10, 36, 50], interactive VOS [23, 37] and audio-guided VOS [52]. Besides, while our benchmark covers a variety of objects, it currently does not include background stuff classes (e.g., sky and river), which could be incorporated in future work for covering more diverse scenarios.