# M$^3$-VOS: Multi-Phase, Multi-Transition, and Multi-Scenery Video Object Segmentation

Zixuan Chen[1*] Jiaxin Li[1,2*] Junxuan Liang[1] Liming Tan[1] Yejie Guo[1]
Cewu Lu[1,2] Yong-Lu Li[1,2†]
[1]Shanghai Jiao Tong University, [2]Shanghai Innovation Institute
{m13953842591, li_jiaxin, whitefork, spinningfever, gyj123, lucewu, yonglu_li}@sjtu.edu.cn

## Abstract

*Intelligent robots need to interact with diverse objects across various environments. The appearance and state of objects frequently undergo complex transformations depending on the object properties, e.g., phase transitions. However, in the vision community, segmenting dynamic objects with phase transitions is overlooked. In light of this, we introduce the concept of **phase** in segmentation, which categorizes real-world objects based on their visual characteristics and potential morphological and appearance changes. Then, we present a new benchmark, **Multi-Phase, Multi-Transition, and Multi-Scenery Video Object Segmentation (M$^3$-VOS)**, to verify the ability of models to understand object phases, which consists of **471** high-resolution videos spanning over 10 distinct everyday scenarios. It provides dense instance mask annotations that capture both object phases and their transitions. We evaluate state-of-the-art methods on M$^3$-VOS, yielding several key insights. Notably, current appearance-based approaches show significant room for improvement when handling objects with phase transitions. The inherent changes in disorder suggest that the predictive performance of the forward entropy-increasing process can be improved through a reverse entropy-reducing process. These findings lead us to propose ReVOS, a new plug-and-play model that improves its performance by reversal refinement. Our data and code will be publicly available at `https://zixuan-chen.github.io/M-cube-VOS.github.io/`.*

## 1. Introduction

Object understanding is crucial, especially for Embodied AI. Lately, large-scale datasets and data-driven methods [6, 19] have advanced research in object understanding, shifting from traditional category recognition [14] to deeper levels of comprehension, such as identifying higher-level affordances [10] and attributes [12], which help establishing the concepts of objects in interactions with environment.

In the real world, intelligent robots need to interact with various objects in diverse ways, during which objects exhibit a range of morphological and appearance changes influenced by their inherent characteristics, the nature of the environment, and the specific interactions occurring. Particularly, the **phase** characteristics of objects describe their intrinsic morphological features and change dynamics.

Objects with different phase characteristics exhibit distinct change features when subjected to the same interactions. For instance, placing solid coffee beans in a blender results in their transformation into a powdered form. In contrast, pouring milk into the blender causes it to splash and flow continuously during blending, potentially even leading to separation at the end. In addition to transformations **within** phase states, **cross**-state phase changes are common in the real world, endowing objects with properties they originally did not have. For example, dry ice rapidly sublimates, producing a lot of mist and water droplets, which will quickly frost at lower temperatures as Fig. 1. Therefore, a deep understanding of an object's intrinsic phase characteristics and the properties of phase transition is crucial for robots to perceive the world, manipulate objects, and fulfill various complex functions according to human needs.

Despite efforts to explore the visual understanding of object change characteristics from various perspectives, such as OCL [18], which focuses on changes in object usability, VOST [32] and VSCOS [37], which emphasize the segmentation of objects with appearance variations, most works remain limited to a single phase. They often overlook the understanding of object knowledge across phase transitions.

Thus, to address the absence of object phase transition understanding, we first categorize different objects based on their visual features and change characteristics commonly observed in daily life. Furthermore, we introduce **M$^3$-VOS**, a fine-grained text-visual annotated dataset for object

*The first two authors contribute equally.
†Corresponding author.

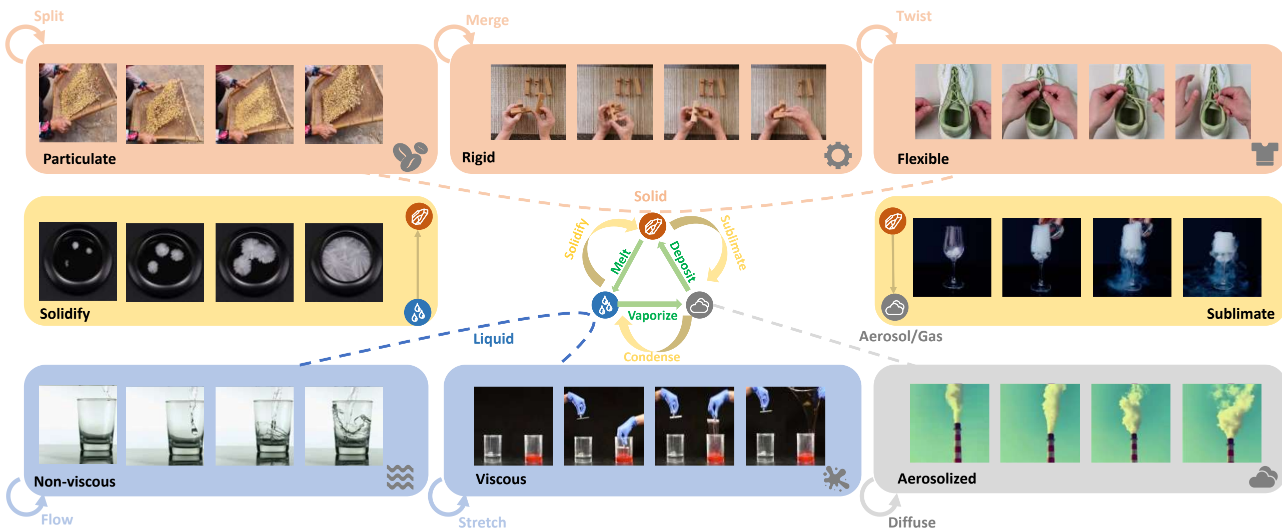



Figure 1. Objects in the real world have various phase states. Each phase has its unique transition (intra-phase transition), such as water flowing, shoelace twisting, smoke diffusing, and so on. Besides, numerous transitions occur across different phases (cross-phase transition), such as bromine liquefaction, dry ice sublimation, and water freezing.

knowledge understanding across multiple scenarios, including objects that encompass a variety of transformations, including appearance changes within phases and state transitions between phases. Motivated by SA-V [28], which was constructed using the pipeline combining SAM2-assisted manual annotation with SAM2-automatic generation, we designed our semi-annotate tool based on the Interactive Demo of Cutie [4]. Besides, we propose an effective method of model cross-validation to mitigate model bias in model-assisted annotation tools.

Given the data, we analyze the current understanding of object state changes in the field of computer vision from the perspective of *Video Object Segmentation (VOS)*. On one hand, VOS task requires the visual model to possess a comprehensive understanding of object phase knowledge to track objects that undergo significant changes in appearance and morphology during phase transitions. On the other hand, video object segmentation serves as an upstream task for other tasks, such as recognizing higher-level attributes and affordances and object manipulation tasks.

Equipped with our benchmark, we analyze state-of-the-art VOS algorithms [2, 4, 28, 36, 38]. We found significant room for improvement in the understanding of objects undergoing phase transitions. Additionally, by analyzing the performance differences across cases, we identified potential challenges in phase changes and proposed specific improvements, establishing a baseline **ReVOS** utilizing reverse memory, booster, and readout fusion to advance object segmentation with phase transformations.

We conclude in Sec. 6.2 by analyzing the different performances of the existing models facing different phase transitions. We hope this work will motivate further exploration into more robust object knowledge understanding. In summary, our main contributions are as follows:

1. We build a video object segmentation benchmark focused on object phase transitions, addressing the gap in understanding object phase transformations in this field.
2. We propose an efficient semi-automated annotation tool tailored for multi-phase object segmentation and a dual-model cross-validation method to address the common issue of model bias in model-assisted annotation tools.
3. To address the limitations of mainstream appearance-first models in understanding object phase transitions, we propose a bidirectional propagation module that enhances the model's performance in this task.

## 2. Related Work

### 2.1. Object Knowledge Understanding

Recently, computer vision has seen significant advancements in object understanding, primarily represented by two paradigms. The first is the classification, which has evolved from object recognition [14] to a deeper understanding of object attributes [12, 16, 17] and affordances [10]. The second focuses on pixel-level classification, encompassing tasks such as image grounding [22, 25] and segmentation [9, 35] of changing objects in videos [4, 28]. The former emphasizes an explicit understanding of object knowledge, where the model is expected to extract relevant information about objects directly from images [10, 12, 14]. In contrast, the latter highlights an implicit understanding of object knowledge, where the model determines the location of objects in an image based on given prompts and its implicit recognition of object knowledge [4, 22, 25, 28].

Data-driven foundation models [20] have achieved remarkable performance in both above paradigms and even

| Dataset | Videos | Frames | Avg len. (s) | Ann fps. | Text | Fluid | CPT |
|---|---|---|---|---|---|---|---|
| DAVIS'16 [24] | 20 | 1376 | 2.87 | 24 | ✓† | × | × |
| DAVIS'17 [26] | 90 | 6,265 | 2.90 | 24 | ✓† | × | × |
| YTVOS [33] | 982 | 25,812 | 4.38 | 6 | ✓† | × | × |
| VSCOS [37] | 98 | 44036 | 8.34 | 0.35 * | × | × | × |
| VOST [32] | 141 | 15,617 | 22.15 | 5 | × | × | × |
| $M^3$-VOS (Ours) | 471 | 202,577 | 14.34 | 30 | ✓ | ✓ | ✓ |

Table 1. Statistics of test set and val set in representative video segment datasets (†: $DAVIS_{16}$-RVOS, $DAVIS_{17}$-RVOS [13], Refer-YouTube-VOS [29] was built based on these datasets; *: VSCOS only annotated some frames in the valid set; Fluid: liquid or aerosol/gas phase); **CPT: Cross-Phase Transitions**. $M^3$-VOS focuses on different phase transitions in multi-scenes, motivating our design to have a higher video annotation frame rate.

demonstrate strong open-vocabulary abilities. However, very few studies focus on understanding object phase transitions, an essential aspect of object knowledge. There is a notable lack of effective benchmarks for testing this task.

In this work, we concentrate on object phases and the corresponding **intra-phase** and **cross-phase transitions**. We construct a benchmark for the second paradigm of object knowledge understanding. It aims to fill the gap and provide a platform to test models on this critical task.

### 2.2. Video Object Segmentation

Video object segmentation is the process of segmenting target foreground objects from the video background at the pixel level within a video sequence. The target foreground object is typically specified by providing a mask in the first frame, known as semi-supervised VOS [24, 26, 33]. This task requires the effective tracking of the target and a clear understanding of its boundaries, which demands that models have a thorough comprehension of the object's appearance [24, 33], dynamic properties [32, 37], and motion information [7] within the video, as well as other physical characteristics. This is precisely why we chose video object segmentation as the specific task to evaluate the ability of vision models to understand object phase transitions.

To prompt the development of the Video Object Segment, [24, 26, 33] constructs a series of early VOS benchmarks. Recently, several works have explored testing the open-world capabilities of VOS models from different perspectives: VOST [32] and VSCOS [37] focus on changes in object appearance, MeVis [7] focuses on the movement characteristics of objects. However, compared to the $M^3$-VOS dataset, these models are all limited to single-phase objects, typically dealing with solid objects. These objects have relatively simple transformation characteristics and highly predictable movement patterns.

There are a lot of VOS methods that achieve success and can be concluded into two classes:

**Memory-based VOS**. Semi-supervised Video Object Segmentation (SVOS) involves inference based on a

| Phase | | | Intra-Phase Transition |
|---|---|---|---|
| Solid | Particulate | | Split |
| | Non-Particulate | Rigid Body | Separate, Merge |
| | | Flexible Body | Twist, Stretch |
| Liquid | Viscous | | Stretch, Paint |
| | Non-Viscous | | Flow, Mix, Split, Paint |
| Aerosol / Gas | | | Diffusion |

Table 2. Phase definition and intra-phase transition according to visual characteristics and intrinsic transformation properties.

ground-true mask for the first frame, and it computes the masks of the rest of the frames. The most frequently used methods are memory-based, which means they store pixel-level or object-level features in their memory bank and compare them with features from newly arrived frames. Recent methods [2, 3] improve the memory bank by stratifying the memory bank into short and frequently updated memories and long and infrequently updated memories to better balance efficiency and performance. A recent method Cutie [4] introduces object memory and fuses it with pixel memory in an object transformer.

**Reverse propagation in VOS**. We propose a method that adopts a refinement module and optimizes the segmentation by reversely propagating the mask through the video sequence. Among the previous methods, there are not many that use backward mask propagation. The most similar method is DyeNet [15], which uses a re-identification module to predict a set of high-munificence masks and uses a bi-directionally proper, nation to the rest frames. Other methods that include elements of reverse optimization typically occur in bidirectional settings for sequences, *e.g.*, bi-directional ConvLSTMs [31]. In contrast, our method doesn't need a high-confidence mask and is implemented as a plug-and-play module as long as the backbone is a mask propagation-based architecture.

## 3. Preliminary

In this section, we introduce the concept of phase and use it to classify everyday materials. Then, we categorize the transformations of objects into **Intra-Phase Transitions** and **Cross-Phase Transitions**.

### 3.1. Phase Categories

Rather than defining the phases of objects from the microscopic perspective provided by chemistry or physics, which focuses on molecular spacing, we adopt a *macroscopic* approach to classify objects based on their appearance and dynamic characteristics. In our study, phase refers to an attribute that describes both visual characteristics and intrinsic transformation properties. Next, we categorize commonly encountered objects into three major types: *solid, liquid*, and *aerosol/gas*. Then we introduce the subdivided phase categories for each major class of materials, along with their respective transformation properties (Tab. 2).

For **solid materials**, particle size also influences their transformation characteristics. Therefore, we further subdivide solid materials into fragmentary or powder-like substances and non-fragmentary substances. For non-fragmentary substances, we categorize them further based on whether the material can easily undergo arbitrary deformation, dividing them into rigid and flexible objects.

For **liquid materials**, substances like melted cheese or melted rubber do not frequently exhibit fluidity in everyday situations. Instead, they demonstrate viscous properties, such as stringiness, when interacting with other materials. Therefore, we classify liquid materials based on their degree of fluidity and viscosity, dividing them into viscous and non-viscous substances.

For **aerosols/gases**, it not only includes common gaseous substances but also solid particles and liquid droplets suspended in the air. These materials possess the property of continuous diffusion in the atmosphere.

### 3.2. Phase Transition

In everyday life, objects undergo various changes through interactions with the environment or humans. Based on whether these changes cross different phase states, we categorize them into *intra-phase* and *cross-phase* transitions.

**Intra-Phase Transformations** primarily depend on the transformation characteristics inherent to the phase state of the object itself. Different phase states of materials often exhibit unique intra-phase transformations.

**Cross-Phase Transformations** are quite common in everyday life but are often overlooked. Due to the differences in phase states before and after the transformation, these objects exhibit distinctly different visual characteristics and transformation properties. Therefore, understanding such transformations poses a greater challenge.

In Supplementary, we provide more detailed definitions of these phases and phase transition.

## 4. Dataset Design and Construction

In this section, we introduce the data collection and annotation of M$^3$-VOS. The key steps include selecting representative videos, annotating them with instance masks and information text, and defining an evaluation protocol.

### 4.1. Video Collection

We chose to source our videos from YouTube and BiliBili, where there are lots of videos about intra-phase transition and cross-phase transition in different scenery: kitchen, outdoor, factory, school, farm, *etc*. In addition, we collect a small amount of data from Ego-Exo4D[11]. In total, **14 scenarios** as shown in Fig. 2a. In these scenarios, target objects undergo intra-phase transitions or cross-phase transitions as they engage in various interactions, including interactions with humans, other objects, and the environment.

Most of the Internet videos were captured using consumer cameras. To better capture rapid and dramatic phase transitions, such as balloon bursts or glass shattering, we also collected **high-frame-rate** video clips shot with high-speed cameras. For slower phase transitions, like ice melting or frost formation, we gathered a portion of **low-frame-rate** video clips taken with time-lapse techniques.

Next, we performed temporal cutting and spatial cropping to focus more on the process of phase transitions. The videos have an average length of 14.34 seconds. To further enrich the scenarios in our dataset, we collected a portion of videos filmed in the laboratory. We have carefully checked all the videos to avoid possible ethnic problems and will only provide access through video URLs along with downloadable preprocessing scripts.

### 4.2. Annotation Design

Each video clip is accompanied by rich annotations for target segmented objects, identifying the corresponding phase transitions. This includes a bilingual (Chinese and English) description of one or more target objects in each clip, the phases of each target object in the first and last frames, and pixel-level segmentation annotations with 30 fps.

In the video collection, we instructed the video collectors to use clear and accurate positional terms to describe each target object in the first frame. The descriptions are "*In the process of [specific action], the [specific object] which at [specific location] in the initial frame or whose color is [specific color].*" , providing accurate guidance for the volunteers responsible for the mask annotations.

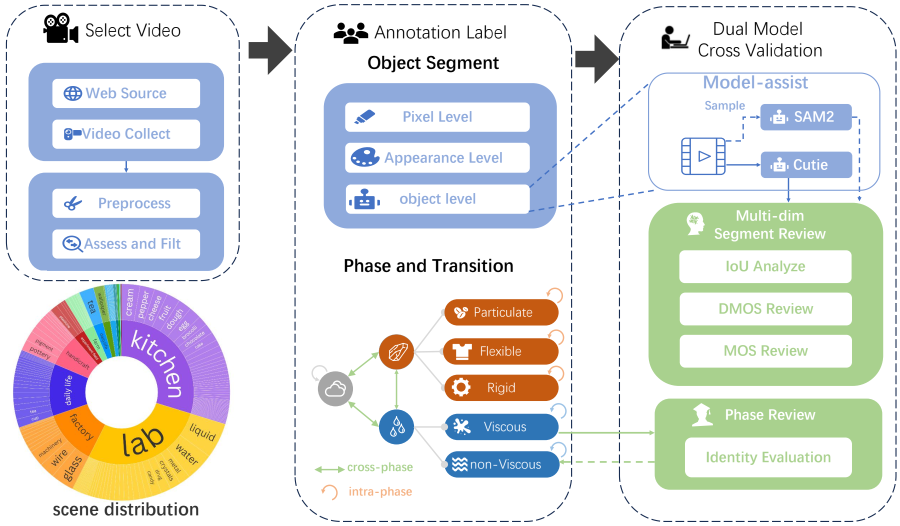
In the annotation, we instructed volunteers to use the information provided by the video collectors about the target objects and the semi-auto Annotation Tool (detailed in Sec. 4.3.1) to annotate the masks of the target objects in each frame of the video. For various *uncertain* void masks caused by unclear image quality or motion blur, We instructed volunteers to annotate the corresponding void masks, similar to the approach in the VOST [32]. In this way, we can disregard the influence of these pixels during the evaluation process. Volunteers were also required to select the *initial* and *final* phases of the target objects from Tab. 2 based on their states in the video and subsequently label the corresponding phase transformations.

### 4.3. Annotation Collection

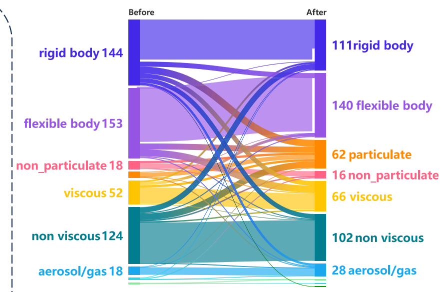
In this section, we introduce our semi-auto Annotation tool which efficiently annotates the masks of target objects in the videos and our annotation collection pipeline.

#### 4.3.1. Semi-Auto Multi-Level Annotation Tool

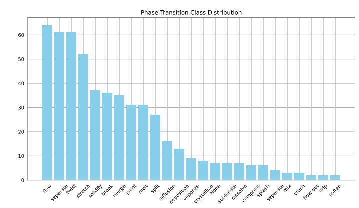
To efficiently annotate the masks of target objects in the videos, we employed a semi-automated multi-level annotation tool that utilizes a paint-erase tool (**pixel-level**), pixel




(a) Dataset construction. The core pipeline component is a multi-level semi-auto segmentation tool including Eraser Brush (**pixel-level**), Color Difference Mask (**appearance-level**), and Model-Assisted Annotation (**object-level**), enabling the efficient annotation of objects with complex phase transitions.




(b) Transition of phase.




(c) Transition category.

Figure 2. $M^3$-VOS characteristics. (a) $M^3$-VOS dataset pipeline including annotating and dual model cross-validation method for model bias. (b) Transition from before phase (left) to after phase (right). (c) Distribution of the phase transition in the $M^3$-VOS.

color difference assistance (**appearance-level**), and neural network-based annotation support (**object-level**).

Our annotation tool employs a *paint-and-erase* approach. Volunteers can use a brush and eraser tool, along with a magnification feature, to accurately annotate the masks of target objects and the void masks in the video. To achieve high-frame-rate mask annotation, we integrated the fully automated annotation tool provided by Cutie [4] based on RITM [30] for interactive image segmentation and Cutie for video object segmentation. After volunteers annotate masks for a specific frame using the paint-and-erase approach or point prompt method, they can utilize Cutie to propagate masks for that frame. Subsequently, the propagated pseudo-ground truth can be refined using the paint-and-erase approach. The *neural network based* automatic annotation tool provides good pseudo ground truth references for objects with clear contours. However, the tool often fails to provide satisfactory pseudo-ground truth for target objects like water vapor, thick smoke, and splashes, which have unclear outlines. Thus, our tool integrates a *color difference masking*, allowing users to select a specific pixel value in the image and set a range to create a color difference template for the video. Given a pixel $p_s$ and any other pixel $p_i$, the position of $p_i$ will be labeled according to

$$M(p_i) = \begin{cases} 1 \iff & \begin{array}{l} \Delta H_s(p_i) < 0.1\delta, \\ \Delta S_s(p_i) < \delta, \\ \Delta V_s(p_i) < \delta, \end{array} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

$$\Delta\phi_s(p_i) = |\phi(p_i) - \phi(p_s)|, \quad \phi \in \{H, S, V\}.$$

We label the pixel $p_i$ if and only if the color difference between $p_s$ and $p_i$ is within $0.1\delta$, and the differences in saturation and brightness are within $\delta$. Here, $\delta$ is a user-adjustable parameter. Our testing has shown that this functionality effectively annotates objects with unclear contours but distinct colors, especially for gases and liquids.

#### 4.3.2. Annotation Pipeline and Statistics

As illustrated in Fig. 2a, in our annotation pipeline, we hired 12 volunteers and provided them with 7 days of training to use the annotation tools on the videos we collected.

To ensure the quality of the data annotations, we employed 3 experienced reviewers to audit the annotated data. Each volunteer's masks were evaluated based on three criteria: tracking accuracy, completeness of annotation, and boundary stability, scored on a scale from 0 to 3 (detailed scoring criteria and the review result can be found in the Supplement). The final mask review score was the average of the two reviewers' ratings. If any score is below 2, we consider the mask Unqualified. In addition to the MOS reviewing, we employed a dual-model cross-validation approach to verify the annotated masks (detailed in Sec. 4.4). If the validated mask annotations do not meet the quality standards, we require the volunteer to annotate again until the review is approved.

Overall, we collect **202,577** masks, with an average track duration of **14.34**s. $M^3$-VOS covers **120+** categories of objects across **6** phases within a total of **14** scenarios, encompassing **23** specific phase transitions. We report additional statistics in Figs. 2b and 2c.

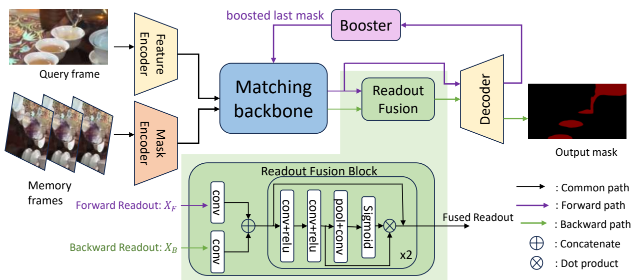



Figure 3. Overview of our ReVOS. ReVOS is a *plug-and-play* based on propagation-based matching backbones. ReVOS includes a Booster Module and a Readout Fusion Module. The forward and reverse pathway is shown in purple and green respectively.

## 4.4. Avoidance of Model Bias

Although using Cutie to generate pseudo-ground truth significantly improves the annotation efficiency, it's crucial to thoroughly verify whether the volunteers' refinements of the pseudo-ground truth can effectively mitigate the model bias introduced by Cutie. Therefore, we introduce our method for ensuring that the annotated masks are free from bias via dual-model cross-validation.

In this process, we sample around 20% video clips in M$^3$-VOS and require half of our volunteers to annotate the masks using another self-implement SAM2-assist tool. Meanwhile, they are required to label masks secondly using the Cutie-assist tool. Finally, we compare three kinds of masks from the two subjective qualitative and the objective quantitative aspects. In this way, we demonstrate that our dataset pipeline effectively mitigates the model bias and that the model bias of mask annotations in M$^3$-VOS is negligible. The details of the process and result can be found in the Supplementary.

## 4.5. Core Subset

To facilitate analysis of the challenge facing the phase transition in the VOS task, we extracted video clips that humans consider to be highly representative and comprehensive in variety. Finally, we split M$^3$-VOS into two subsets for evaluation, including full subset and core subset. The details can be found in the supplementary.

# 5. Method

In data collection and annotation, we observed that when objects undergo intra-phase or cross-phase transitions, their inherent *disorder* tends to increase. This hidden phenomenon makes mask predictions increasingly more difficult. If the final state of the object at the end of the video is well-annotated, *reverse-propagating* to predict the mask becomes more accurate. We hypothesize that this is because the reverse process involves a gradual reduction in disorder, which simplifies segmentation.

| Dataset | M$^3$-VOS | VOST | YouTube VOS-2019 val | DAVIS' 17 val |
|---|---|---|---|---|
| First half | 7.16 | 8.29 | 8.10 | 8.11 |
| Latter half | 7.18 | 8.32 | 8.13 | 8.12 |

Table 3. $h$ of the first half and the latter half of VOS datasets.

Given an object in a specific frame, we use a weighted combination of Local Binary Pattern (LBP) [23] of its mask and entropy of the original gray-scale image to describe its level of disorder. The overall level of disorder $h$ can be formulated as Eq. (2):

$$\begin{aligned} Q_\phi(i) &= -\sum_i H_\phi(i) log_2 H_\phi(i), \quad \phi \in \{L, O\}. \\ h &= \alpha Q_L(i) + \beta Q_O(j). \end{aligned} \tag{2}$$

where $Q_\phi(i)$ represents the entropy of the distribution $H_\phi(i)$. $H_L(i)$ represents the frequency of the LBP value $i$ and $H_O(j)$ represents the frequency of pixel value $j$. We calculated the average $h$ for the masks in the first and second halves of several datasets. We set $\alpha = 0.2$ and $\beta = 1.0$. The results are shown in Tab. 3. In all datasets, the $h_{LBP}$ values in the latter half are higher than those in the first half, which supports our hypothesis.

Given this, we propose a paradigm to improve performance using reverse-propagation. We introduce a *plug-and-play* framework named ReVOS based on mask propagation backbones (Fig. 3). First, the model performs a forward-propagating pipeline based on the matching backbone, where the last output mask will be boosted and reintroduced into the model. Then, the model will perform **reverse propagation** on a sliding window of previous frames. The object readout of the forward and reverse pipeline will be *fused* in the Readout Fusion Module to generate better results. In the following, we describe the three main contributions of ReVOS.

**Reverse Memory**. In the forward propagation of the mask, we maintain a sliding window of length $T$. Suppose the model needs to predict the $t$-th frame as the current time. The sliding window includes image information from $max(0, t-T)$ to $max(0, t-1)$ and the extracted forward memory readout. The reverse memory is similar to working memory in [2]. The difference is that it only stores high-resolution features for the reverse process, serving the same function as working memory in the reverse propagation. It is cleared at the beginning of each reverse propagation as we want the reverse memory to only collect information from the current reverse process.

**Booster**. If the model loses information about the mask for certain object parts during the forward process, this loss will be taken into the reverse process and continue to adversely affect the segmentation results. To address this issue, we implement a strategy to boost the mask during forward propagation. This enhancement aims to predict as many locations as possible, thereby reducing local mask loss. The boosted mask will be re-input into the matching backbone to generate the next frame mask. In the implementation, we output the final mask via

$$M = \sigma(\alpha X_{decode}), \tag{3}$$

where $\alpha$ is boosting factor, $\sigma$ is sigmoid function, $X_{decode}$ is the decoded logits, $M$ is the boosted mask.

**Readout Fusion**. In the forward and reverse processes, readout features will be extracted from the matching backbone. To obtain the final mask, we design a Fusion module to integrate readout features from the forward process, as shown in Fig. 3.

### 5.1. Implementation Details

We adopt Cutie-base [4] as our matching backbone, which is trained under "MEGA" setting with 5 datasets: YoutubeVOS [34], DAVIS [24], BURST [1], OVIS [27], and MOSE [8]. In training, we froze all the parameters of Cutie and only trained the Readout Fusion Module. We finetune our model with AdamW [21] optimizer. The learning rate is set to 1e-5. Finetuning takes about 75k iterations and we reduce the learning rate by 10 times after 60K and 67.5K iterations. The model is trained on 4 A100 GPUs for 10 hours.

## 6. Analysis

In evaluation, we adopt standard metric Jaccard index $\mathcal{J}$, newly proposed metrics $\mathcal{J}_{tr}$ [32] and $\mathcal{J}_{cc}$ [37]. $\mathcal{J}_{tr}$ represents the Jaccard index of the last 25% frames and $\mathcal{J}_{cc}$ represents the Jaccard index based on each connected component and takes an average over all components. For YouTubeVOS [34], we use its official metrics: $\mathcal{J}$ and $\mathcal{F}$ for seen and unseen categories. $\mathcal{G}$ is the averaged $\mathcal{J}\&\mathcal{F}$ for both seen and unseen classes.

### 6.1. Main Results

We compare ReVOS with SOTA approaches on our dataset and standard benchmarks: VOST validation [32], DAVIS 2017 validation [26] and YouTubeVOS validation [33]. For our dataset, we create two versions: M$^3$-VOS full (all cases) and M$^3$-VOS core (highly representative cases). For a fair comparison and open-vocabulary purpose, we choose only one checkpoint that each method performs the best on the DAVIS 2017 validation set and use it to test all other datasets. For DeAOT, we excluded a few long video cases as we encountered insufficient memory during inference.

In Tab. 4, ReVOS achieves better results than all SOTA methods, especially on our dataset. As we freeze the Cutie backbones during training, it is evident that the increase in our model's score relative to Cutie is entirely due to our new framework. Specifically, SAM2 achieved the best result on VOST. We assume it takes more training data similar to EPIC-kitchens [5] from which the VOST is constructed.

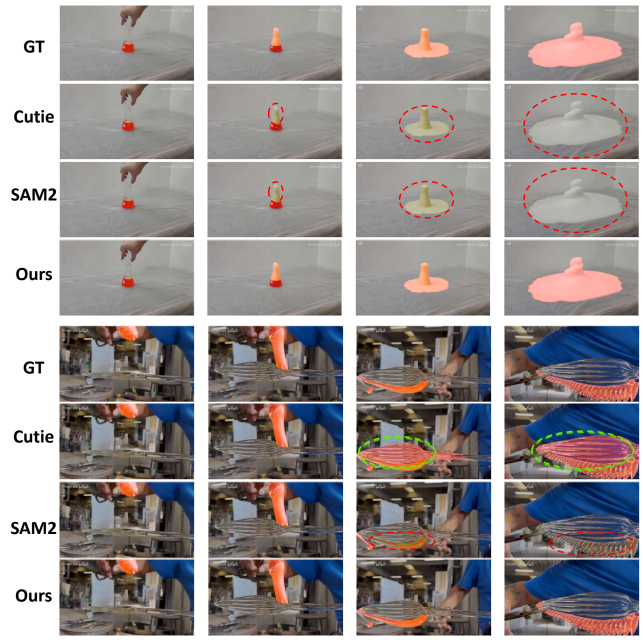



Figure 4. Visualization of qualitative results. (Red circle: false-negative region; Green circle: false-positive region). The highlighted color region is the predicted mask. Phase transitions, *e.g.*, reagent eruptions and glass solidification, challenge current models in boundary prediction, causing tracking failures or background interference. However, our reverse inference correction effectively addresses these issues.

### 6.2. Intra-Phase vs. Cross-Phase Transformations

We present the average model performance in 4 categories: Intra-Phase (solid), Intra-Phase (Liquid), Intra-Phase (Aerosol/Gas), and Cross-Phase, which are abbreviated as IS, IL, IG and CP in Tab. 5. In addition, we also present the performance of our model for the detailed phase transitions in Fig. 5.

Compared to pure Cutie, our Cutie-ReVOS improves the performance in all phase transitions. Compared to the SOTA, Cutie-ReVOS still demonstrated excellent performance facing these phase transitions. Especially when facing the cross-phase transition, Cutie-ReVOS achieves the best performance in all models we have evaluated.

### 6.3. Performance Differences across Benchmarks

Compared to existing benchmarks, M$^3$-VOS significantly expands the phase range of objects, introducing a series of new challenges for tracking and segmentation. For example, M$^3$-VOS includes liquids and granular solids, where target objects often exhibit rapid motion due to splashing or splitting. A large portion of fluids (both gases and liquids) also tend to be transparent, and phase transitions are

| | M$^3$-VOS full | | | M$^3$-VOS core | | | VOST val | | | DAVIS'17 val | | | YouTubeVOS-2019 val | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{G}$ | $\mathcal{J}_s$ | $\mathcal{F}_s$ | $\mathcal{J}_u$ | $\mathcal{F}_u$ |
| DeAOT * [36] | 72.5 | 65.2 | 61.2 | 62.3 | 55.1 | 51.4 | **45.6** | **33.2** | **35.1** | 82.7 | 79.8 | 60.9 | 86.2 | 85.6 | **90.6** | 80.0 | 88.4 |
| XMem [2] | 70.4 | 61.5 | 58.8 | 60.6 | 50.6 | 49.2 | 36.0 | 24.8 | 26.6 | 82.9 | 81.3 | 61.3 | 85.4 | 84.3 | 88.6 | 80.3 | 88.5 |
| RMem + DeAOT † [38] | 73.4 | 66.1 | 62.6 | 56.1 | 45.5 | 46.7 | 40.5 | 25.3 | 31.9 | 82.3 | 79.3 | 60.9 | 85.9 | 84.6 | 89.4 | 80.8 | 88.9 |
| SAM2 [28] | 69.5 | 57.8 | 58.2 | 61.0 | 49.9 | 48.9 | 44.1 | 28.0 | 32.7 | 85.5 | 82.8 | 64.1 | 85.2 | 83.7 | 80.6 | **87.9** | 88.5 |
| Cutie-base [4] | 74.6 | 64.6 | 64.3 | 64.6 | 52.0 | 53.9 | 40.8 | 25.1 | 31.6 | 85.6 | 84.6 | 63.9 | 86.8 | **85.9** | 90.4 | 81.6 | 89.3 |
| Ours | **75.6** | **66.5** | **65.2** | **66.3** | **55.8** | **55.5** | 41.0 | 25.3 | 31.7 | **86.0** | **84.8** | **64.2** | **86.8** | 85.3 | 89.8 | 82.1 | **89.9** |

Table 4. Comparison of semi-VOS methods. The metrics are $\mathcal{J}$, $\mathcal{J}_{tr}$, and $\mathcal{J}_{cc}$ for M$^3$-VOS (full, core), VOST, and DAVIS'17 ($^\dagger$: R50-DeAOT-L with restricted memory bank without temporal position embedding; *: SwinB-DeAOT-L; the grep region contains 469 objects. The Video clip of Other Objects is too long. DeAOT doesn't support long videos due to the memory limitation.)

| | IS | | | IL | | | IG | | | CP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ |
| DeAOT [36] | 71.6 | 64.7 | 59.8 | 72.1 | 65.0 | 59.9 | 80.0 | 74.3 | 67.9 | 73.8 | 66.0 | 65.3 |
| XMem [2] | 70.9 | 62.8 | 58.9 | 68.2 | 58.7 | 56.1 | 80.2 | 71.3 | 64.7 | 74.5 | 66.4 | 65.4 |
| Rmem + DeAOT [38] | 74.5 | **67.3** | **63.3** | 71.3 | 63.6 | 59.8 | **83.5** | **82.0** | **68.5** | 77.0 | 70.4 | 68.3 |
| SAM2 [28] | 64.7 | 52.6 | 54.7 | 69.8 | 57.7 | 57.6 | 67.6 | 57.4 | 56.1 | 74.2 | 63.9 | 64.0 |
| Cutie-base [4] | 72.3 | 61.7 | 62.5 | 74.1 | 64.1 | 63.3 | 75.8 | 64.3 | 63.6 | 77.5 | 69.7 | 69.0 |
| Ours | **73.0** | 63.2 | 63.3 | **75.6** | **66.4** | **64.2** | 76.8 | 66.5 | 65.4 | **78.1** | **70.8** | **69.6** |

Table 5. Performance of different VOS methods in phase transitions.

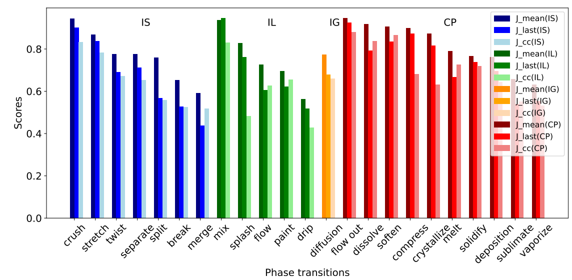



Figure 5. Results of ReVOS on phase transitions.

often accompanied by changes in the object's appearance and color. As Tab. 4 demonstrates, the performance of all existing models showed a significant decline.

Especially, comparing the full M$^3$-VOS full set and the M$^3$-VOS core subset, which balances the distribution of scene and phase transition, the existing modes have more obvious performance deficiencies. This phenomenon shows that the current models all have a certain degree of scenery bias and phase bias.

## 6.4. Ablation Study

We study various designs of our algorithm in the ablation. We report $\mathcal{J}$, $\mathcal{J}_{tr}$ and $\mathcal{J}_{cc}$ and FPS for M$^3$-VOS mid set.

**Hyperparameter Choices**. Tab. 6 compares our results with different choices of hyperparameters: sliding window length $T$ and reverse interval $L$. Note that $T = 0$ or $L = 0$ is equivalent to Cutie-base. $T$ is fixed to 30 when we vary the value of $L$ and vice versa. We find that a larger sliding window length means high performance but also implies a decrease in FPS. Surprisingly, a larger reverse interval also means higher performance with an increase in FPS. Tab. 7 shows that the performance improvement brought by the booster and readout fusion demonstrates the necessity of adding these modules.

| Setting | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ | FPS |
|---|---|---|---|---|
| *Sliding window length* | | | | |
| 10 | 75.6 | 66.6 | 65.2 | 15.3 |
| 30 | **75.7** | **66.4** | **65.3** | **15.4** |
| 60 | 75.4 | 66.3 | 65.0 | 10.4 |
| *Reverse interval* | | | | |
| 10 | 75.5 | 66.5 | 65.1 | 7.9 |
| 30 | **75.7** | 66.4 | 65.3 | 15.4 |
| 60 | 75.7 | **66.7** | **65.4** | **21.6** |

Table 6. Comparison with different hyperparameters.

| Setting | $\mathcal{J}$ | $\mathcal{J}_{tr}$ | $\mathcal{J}_{cc}$ |
|---|---|---|---|
| Only forward (Cutie) | 74.7 | 64.6 | 64.5 |
| Only backward | 75.4 | 66.4 | 64.9 |
| With readout fusion | **75.7** | **66.4** | **65.3** |
| Without booster | 74.2 | 64.2 | 63.9 |
| With booster | **75.7** | **66.4** | **65.3** |

Table 7. Ablation on Booster and Readout Fusion Module.

## 7. Discussion and Limitations

**Annotation Bias**. Annotation bias is inevitably introduced by annotators' tendencies, inherent biases in assisted tools, and ambiguous regions in videos. As discussed in Sec. 4.4, despite our implementation of multi-round, multi-level review processes, some discrepancies remain across annotation instances. Establishing a more standardized and rigorous data annotation review workflow could effectively mitigate annotation bias.

**Model Performance**. All models show limitations on our new benchmark with challenging object phase transitions. In future works, models may be advanced by leveraging the physical knowledge embedded in MLLMs.

## 8. Conclusion

In this work, we explored the ability of the visual models to understand object phase transitions through a video segmentation task, introducing a fine-grained text-visual annotated open-vocabulary benchmark M$^3$-VOS. It includes 14 scenes, encompassing 120+ objects across 6 phase states and 23 phase transitions (13 intra-phases, 10 inter-phases). To address the limitations of existing methods, we proposed a plug-and-play module ReVOS based on entropy theory, achieving decent improvements in M$^3$-VOS and comparable performance on other benchmarks. However, effectively understanding phase transitions remains a challenge. We believe this will pave the way for a new direction in research focused on a deeper understanding of objects.

## 8.1. Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grant No.62306175, CCF-Tencent Rhino-Bird Open Research Fund.

## References

[1] Ali Athar, Jonathon Luiten, Paul Voigtlaender, Tarasha Khurana, Achal Dave, Bastian Leibe, and Deva Ramanan. Burst: A benchmark for unifying object recognition, segmentation and tracking in video. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1674–1683, 2023. 7

[2] Ho Kei Cheng and Alexander G Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *European Conference on Computer Vision*, pages 640–658. Springer, 2022. 2, 3, 6, 8

[3] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Rethinking space-time networks with improved memory coverage for efficient video object segmentation. *Advances in Neural Information Processing Systems*, 34:11781–11794, 2021. 3

[4] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Joon-Young Lee, and Alexander Schwing. Putting the object back into video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3151–3161, 2024. 2, 3, 5, 7, 8

[5] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pages 720–736, 2018. 7

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[7] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. Mevis: A large-scale benchmark for video segmentation with motion expressions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2694–2703, 2023. 3

[8] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. Mose: A new dataset for video object segmentation in complex scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20224–20234, 2023. 7

[9] Hao-Shu Fang, Jianhua Sun, Runzhong Wang, Minghao Gou, Yong-Lu Li, and Cewu Lu. Instaboost: Boosting instance segmentation via probability map guided copy-pasting. In *ICCV*, 2019. 2

[10] James J Gibson. The ecological approach to the visual perception of pictures. *Leonardo*, 11(3):227–235, 1978. 1, 2

[11] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, Eugene Byrne, Zach Chavis, Joya Chen, Feng Cheng, Fu-Jen Chu, Sean Crane, Avijit Dasgupta, Jing Dong, Maria Escobar, and et al. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives, 2024. 4

[12] Phillip Isola, Joseph J Lim, and Edward H Adelson. Discovering states and transformations in image collections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1383–1391, 2015. 1, 2

[13] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part IV 14*, pages 123–141. Springer, 2019. 3

[14] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1, 2

[15] Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 90–105, 2018. 3

[16] Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. Symmetry and group in attribute-object compositions. In *CVPR*, 2020. 2

[17] Yong-Lu Li, Yue Xu, Xinyu Xu, Xiaohan Mao, and Cewu Lu. Learning single/multi-attribute of object with symmetry and group. *TPAMI*, 2022. 2

[18] Yong-Lu Li, Yue Xu, Xinyu Xu, Xiaohan Mao, Yuan Yao, Siqi Liu, and Cewu Lu. Beyond object recognition: A new benchmark towards object concept learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20029–20040, 2023. 1

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1

[20] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2024. 2

[21] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 7

[22] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions, 2016. 2

[23] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. 6

[24] 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*, pages 724–732, 2016. 3, 7

[25] Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models, 2016. 2

[26] 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:1704.00675*, 2017. 3, 7

[27] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip HS Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *International Journal of Computer Vision*, 130(8): 2022–2039, 2022. 7

[28] 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. *arXiv preprint arXiv:2408.00714*, 2024. 2, 8

[29] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pages 208–223. Springer, 2020. 3

[30] Konstantin Sofiiuk, Ilia A. Petrov, and Anton Konushin. Reviving iterative training with mask guidance for interactive segmentation, 2021. 5

[31] Hongmei Song, Wenguan Wang, Sanyuan Zhao, Jianbing Shen, and Kin-Man Lam. Pyramid dilated deeper convlstm for video salient object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 715–731, 2018. 3

[32] Pavel Tokmakov, Jie Li, and Adrien Gaidon. Breaking the" object" in video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22836–22845, 2023. 1, 3, 4, 7

[33] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation, 2018. 3, 7

[34] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 7

[35] Wenqiang Xu, Yonglu Li, and Cewu Lu. Srda: Generating instance segmentation annotation via scanning, reasoning and domain adaptation. In *ECCV*, 2018. 2

[36] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. *Advances in Neural Information Processing Systems*, 35:36324–36336, 2022. 2, 8

[37] Jiangwei Yu, Xiang Li, Xinran Zhao, Hongming Zhang, and Yu-Xiong Wang. Video state-changing object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20439–20448, 2023. 1, 3, 7

[38] Junbao Zhou, Ziqi Pang, and Yu-Xiong Wang. Rmem: Restricted memory banks improve video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18602–18611, 2024. 2, 8