Segment Anything Across Shots: A Method and Benchmark

Hengrui Hu, Kaining Ying, Henghui Ding*

Institute of Big Data, College of Computer Science and Artificial Intelligence, Fudan University, China
hhding@fudan.edu.cn

Abstract

This work focuses on multi-shot semi-supervised video object segmentation (MVOS), which aims at segmenting the target object indicated by an initial mask throughout a video with multiple shots. The existing VOS methods mainly focus on single-shot videos and struggle with shot discontinuities, thereby limiting their real-world applicability. We propose a transition mimicking data augmentation strategy (TMA) which enables cross-shot generalization with single-shot data to alleviate the severe annotated multi-shot data sparsity, and the Segment Anything Across Shots (SAAS) model, which can detect and comprehend shot transitions effectively. To support evaluation and future study in MVOS, we introduce Cut-VOS, a new MVOS benchmark with dense mask annotations, diverse object categories, and high-frequency transitions. Extensive experiments on YouMVOS and Cut-VOS demonstrate that the proposed SAAS achieves state-of-the-art performance by effectively mimicking, understanding, and segmenting across complex transitions.

Code and Data — https://henghuiding.com/SAAS/
Extended Version — https://arxiv.org/abs/2511.13715

1 Introduction

Semi-supervised video object segmentation (VOS) (Caelles et al. 2017) aims to segment and track the target object throughout a video sequence, given its mask in the first frame as a prompt. This task has received increasing attention (Ravi et al. 2024) in the research community because of its broad applicability in human–robot interaction, video editing, autonomous driving, and annotation assistance, *etc*.

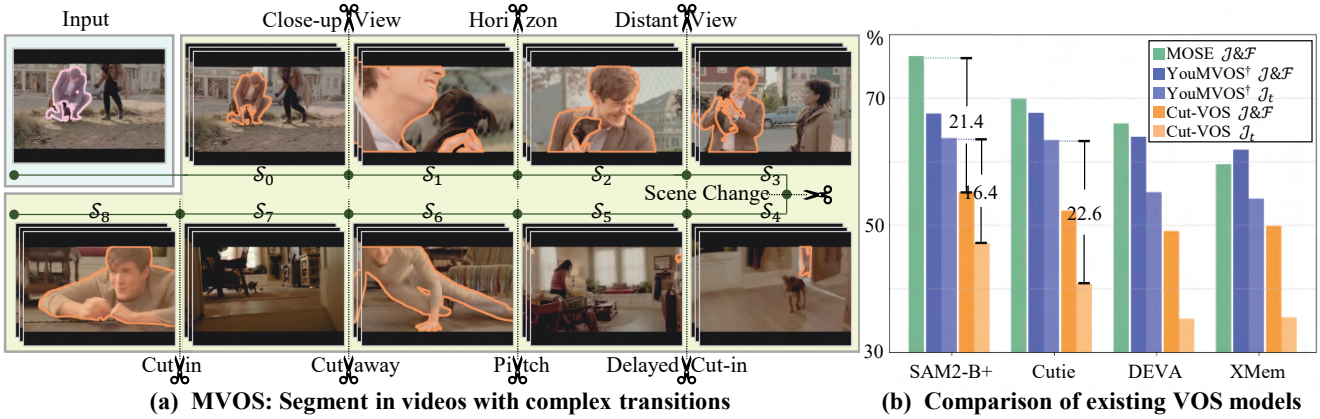
Despite notable progress, existing VOS methods predominantly focus on single-shot videos, overlooking the increasing prevalence of multi-shot videos (see Figure 1 (a)) in real-world Internet content. This oversight on **multi-shot video object segmentation** (**MVOS**) has led to a widening gap between academic research and practical deployment. The current representative VOS methods, *e.g*. XMem (Cheng and Schwing 2022), DEVA (Cheng et al. 2023), Cutie (Cheng et al. 2024), and SAM2 (Ravi et al. 2024) exhibit a notable performance degradation when exposed to complex shot transitions. As shown in Figure 1 (b), SAM2-B+ suffers a 21.4% $\mathcal{J}\&\mathcal{F}$ drop on the MVOS benchmark compared to MOSE (Ding et al. 2023b), highlighting their limitations in the applications of edited videos, multi-camera systems, and high-mobility platforms.

To our knowledge, YouMVOS (Wei et al. 2022) is currently the only dataset that supports MVOS. However, upon reviewing the playlists provided in their dataset, we find that the dataset falls short in fully reflecting the challenges of MVOS task. Specifically, the dataset contains only sparse shot transitions, exhibits a limited diversity of object categories with a predominant focus on humans, and lacks screening or categorization of transition types, as shown in Figure 2. Furthermore, the mask annotations of YouMVOS have not been open-sourced to date, making it unavailable for subsequent model development and training.

To address the lack of multi-shot training data, we propose the **T**ransition **M**imicking Data **A**ugmentation (**TMA**) strategy, which simulates diversiform shot transitions on single-shot datasets to enable effective multi-shot segmentation training without relying on native multi-shot annotations. Meanwhile, the deficiencies of previous methods in complex multi-shot videos, as shown in Figure 1 (b), prompt us to develop a specialized cross-shot segmentation method, **S**egment **A**nything **A**cross-**S**hot (**SAAS**), equipped with transition detection and comprehension modules. These modules jointly detect and interpret shot transitions using adjacent frames along with background context, guided by two auxiliary training objectives. Additionally, we introduce a training-free memory refinement mechanism through a local memory bank that stores fine-grained object features to enhance segmentation quality across transitions.

To fairly evaluate cross-shot segmentation performance and better reflect the complexity of real-world multi-shot videos, we introduce a new MVOS benchmark, **C**omplex M**u**lti-sho**t** **V**ideo **O**bject **S**egmentation (**Cut-VOS**), containing 10.2K instance masks for 174 unique objects in 100 videos. Compared to YouMVOS, the proposed Cut-VOS provides 1.6× higher shot transition frequency and 3× more object categories. The transition types are manually screened to ensure greater diversity and difficulty. For qualitative comparison, we build YouMVOS† test split, a manually annotated version, as they don't release mask annotations. We sample and annotate 30 videos across 10

*Corresponding author.



(a) MVOS: Segment in videos with complex transitions **(b) Comparison of existing VOS models**

Figure 1: This work focuses on an underexplored task of multi-shot video object segmentation (MVOS). As shown in (a), the significant variations in object appearance, spatial location, and background across shots pose major challenges in MVOS. We introduce Cut-VOS, a challenging MVOS benchmark with high transition diversity to support this task. As shown in (b), on Cut-VOS, SAM2-B+ exhibits a 21.4% $\mathcal{J}\&\mathcal{F}$ drop compared to the challenging single-shot MOSE dataset and a 16.4% $\mathcal{J}_t$ drop compared to YouMVOS†, a sampled MVOS dataset YouMVOS annotated by our team strictly following its original protocol. The metric $\mathcal{J}_t$ specifically measures cross-shot segmentation performance, further highlighting the difficulty of Cut-VOS.

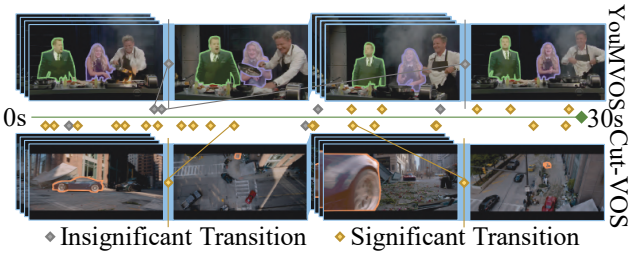


Figure 2: The comparison between YouMVOS and our proposed Cut-VOS benchmark. Cut-VOS is distinguished from YouMVOS by frequent, significant transitions and more variety in complex scenarios.

genres from the playlist, strictly following their protocol. All the experiments in this paper are conducted on the reconstructed version. Compared to YouMVOS, the models perform significantly worse on Cut-VOS, as shown in Figure 1 (b), indicating a substantial difficulty gap. Extensive experiments demonstrate that SAAS achieves consistent improvements across both YouMVOS and Cut-VOS.

Overall, the key contributions of this work are as follows:

- We introduce a new VOS training strategy, **T**ransition **M**imicking Data **A**ugmentation (TMA), to alleviate data sparsity by simulating shot transitions, thereby promoting the model's multi-shot segmentation capacity using only single-shot datasets.
- To the best of our knowledge, the proposed **SAAS** is the first semi-supervised VOS method specialized for multi-shot videos. It incorporates online transition detection, transition comprehension, and local visual cue encoding. Extensive experiments demonstrate its robustness and effectiveness in complex multi-shot scenarios.
- To facilitate future research in MVOS, we introduce **C**omplex **Mu**lti-sho**t** **V**ideo **O**bject **S**egmentation (**Cut-VOS**) dataset, which will become the first fully open-sourced MVOS benchmark with mask annotations upon publication. Cut-VOS provides diverse object categories and carefully curated transition types to evaluate cross-shot tracking performance.

2 Related Work

Video Object Segmentation. Video object segmentation (VOS) (Caelles et al. 2017; Lin, Qi, and Jia 2019; Huang et al. 2020; Ding et al. 2023a, 2025b; Ying, Hu, and Ding 2025; Ding et al. 2025a; Liu et al. 2025) aims at tracking and segmenting the objects in a video sequence, given the mask in the first frame. Early methods (Xiao et al. 2018; Perazzi et al. 2017) are mostly fine-tuning-based. They model inter-frame correlations via fine-tuning during inference. Matching-based methods (Cheng et al. 2018; Duarte, Rawat, and Shah 2019; Duke et al. 2021) generate an object prototype embedding from the conditional frame, performing pixel-level matching to classify each pixel as foreground or background. Propagation-based methods (Han et al. 2018; Hu et al. 2018; Jabri, Owens, and Efros 2020; Wang et al. 2019) leverage the previous frames and predictions to guide the segmentation on the current frame. For better use of historical information, recent methods introduce a memory bank to compress and store previous frames. For example, XMem (Cheng and Schwing 2022) conducts multiple granularities of memories, while Cutie (Cheng et al. 2024) enriches the memory bank with object-specific queries. Most recently, SAM2 (Ravi et al. 2024) extends SAM (Kirillov et al. 2023) to the video domain, yielding a remarkable improvement via a robust memory architecture and large-scale training. However, these previous methods only focus on single-shot videos, lacking solid cross-shot tracking capacity, which leads to their limited applications. This work aims to generalize VOS

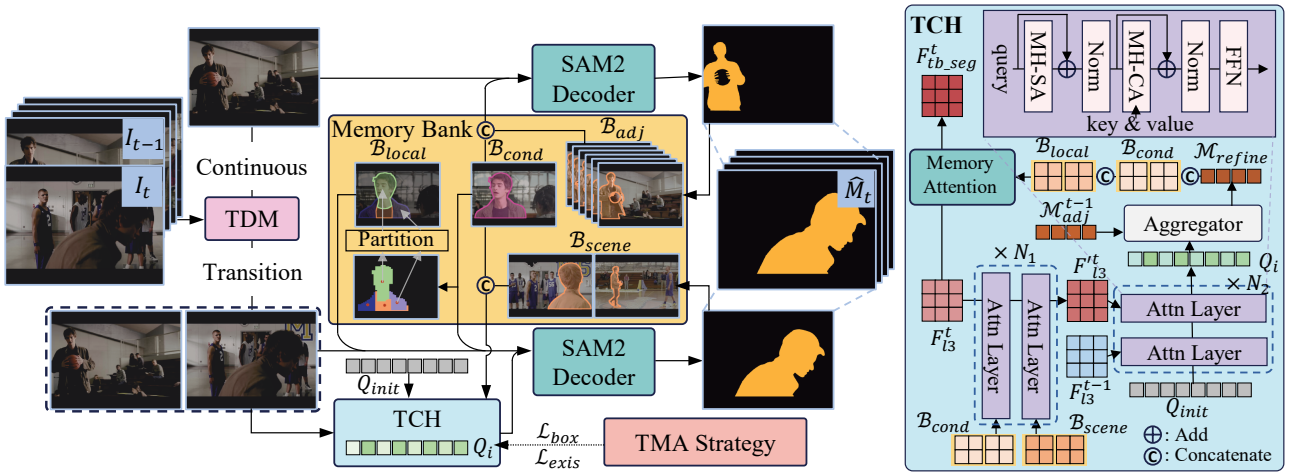


Figure 3: The overall pipeline of our proposed Segment Anything Across Shots (SAAS) method, consisting of three new components, Transition Detection Module (TDM), Transition Comprehension Module (TCH), and local memory bank $\mathcal{B}_{local}$. Transition Mimicking Augmentation (TMA) is employed to train the model by synthesizing high-quality multi-shot training samples using annotated single-shot videos.

to multi-shot videos, bridging the gap between the current research and practical requirements.

Multi-shot Video Understanding. Multi-shot videos, which circulate on the internet at an increasingly large scale, have gradually attracted the attention of the computer vision community. Most early works (Canny 1986; Jacobs et al. 2004; Qian, Liu, and Su 2006) aim to detect the shot boundaries with manual features. With the development of deep learning, some methods (Hassanien et al. 2017; Soucek and Lokoc 2024; Bouyahi and Ayed 2020; Wang et al. 2021) adapt 3D-CNN (Ji et al. 2012) and dilated filter (Chen et al. 2017; Yu, Koltun, and Funkhouser 2017) to improve model accuracy. Meanwhile, some works collect multi-shot videos in their video captioning benchmarks (Xu et al. 2016; Krishna et al. 2017; Zhou, Xu, and Corso 2018), asking the model to generate video descriptions. Recently, Shot2Story (Han et al. 2023) and MMBench-Video (Fang et al. 2024) posed more fine-grained questions, requiring clip-wise understandings to answer. MUSES (Liu et al. 2021a) focuses on the multi-shot temporal event localization task which requests dense frame labels. However, these works still lack the exploration of pixel-level instance segmentations (Ying et al. 2022, 2023). This paper specifically targets fine-grained segmentation in multi-shot videos.

3 Methodology

3.1 Overview

Figure 3 shows an overview of our approach, which contains the proposed Transition Mimicking Data Augmentation (TMA) training strategy and a transition-aware method, Segment Anything Across Shots (SAAS), built upon the SAM2 to generalize VOS to multi-shot videos. Given a video $\mathcal{V} = \{I_t\}_{t=1}^{T}$ with T frames, and the first frame I_0 with ground truth mask M_0, SAAS firstly applies SAM2 image encoder to extract multi-level visual features $\{F_{li}^t\}_{i=1,2,3}$. At each timestep t, SAAS introduces the Transition Detection Module (TDM) to detect if a shot transition occurs and subsequently directs to diverse segmentation strategies. For the detected transitions, the following Transition Comprehension Module (TCH) further comprehends them, generates compressed transition state representation Q_i, thereby refining previous memories. To capture the local fine-grained features of objects, we also propose the local memory bank B_{local}, to partition the target and store corresponding information unsupervisedly. The conditional memories from B_{cond} and features stored in B_{local} are then concatenated to generate the features prepared to be segmented $F_{tb\_seg}^t$, used to finally predict $\hat{M}_t$ by the mask decoder. The entire architecture is trained via the TMA strategy, with two additional objectives.

3.2 Transition Mimicking Augmentation

One of the most critical challenges for MVOS is the lack of available training data. To address this issue, we propose **T**ransition **M**imicking Data **A**ugmentation (**TMA**), a new strategy which synthesizes quality-approved multi-shot training samples from annotated single-shot videos by simulating diverse transitions. TMA enables the effective MVOS training utilizing existing single-shot VOS datasets, significantly alleviating data scarcity.

We show some primary patterns involved in TMA in Figure 4. TMA maintains a conventional 8-frame continuous sampling strategy in previous VOS works with a probability $1 - p_{trans}$, otherwise performs a transition mimicking operation. Specifically, TMA conducts a single transition (as shown in (a), (b), and (d)) with a probability p_{once}, otherwise applies multiple transitions (as depicted in (c)). For each expected transition, TMA employs a well-defined framework with several control random variables to generate different transition patterns. For example, case (a) retains

Figure 4: Some visualization cases of our proposed TMA strategy. (a) Random strong transforms. (b) Single transition across different temporal segments from the same video. (c) Multiple transitions, conducting a case with *cut in* and *cut away*. (d) Single transition to another video, with random replication and gradual translations.

a continuous 8-frame sampling but applies strong transformations, including horizon flipping, random scaling, and random affine on posterior frames after the transition. This pattern simulates common view transitions, like *close-up view* or *distant view*. Case (b) cuts to a different segment from the same video, with a higher probability of sampling more further frames. The substantial temporal gap between the two clips often results in significant changes in object poses and camera viewpoints. Case (c) cuts to an unrelated video and cuts back later, like the *cut away* and *cut in* transitions. Case (d) cuts to an unrelated video while replicating the object with a random, gradual translation, simulating the *scene change* and the *delayed cut in* transitions effectively. TMA fully combines these patterns to preserve data richness while carefully avoiding ambiguous samples and anomalous noises. More details are offered in the appendix.

3.3 Transition Detection and Comprehension

Transition Detection Module. SAAS employs a lightweight **T**ransition **D**etection **M**odule (**TDM**) to detect different shot segments and occurring transitions in video sequences. Inspired by previous shot boundary detection methods (Tu et al. 2017; Soucek and Lokoc 2024), we conduct a dilated convolution pyramid (Chen et al. 2018, 2019) as TDM. At each timestep t, TDM predicts a probability score for current frame I_t, directing to different pipelines:

$$\hat{p}_{i,tr} = \text{Sigmoid}(\mathcal{F}_{\text{TDM}}(F^t, F^{t-i}_{i=1,2,...,N})), \quad (1)$$

where $\mathcal{F}_{\text{TDM}}$ indicates the main network of TDM which uses the adjacent N frames for detection. When $\hat{p}_{i,tr} < \tau_{tr}$, SAAS passes through the SAM2 segmentation pipeline (the upper part in Figure 3) directly, and only encodes the memory $\mathcal{M}_t$ into the bank $\mathcal{B}_{adj}$. Otherwise, SAAS recognizes the transition occurs and adopts a transition segmentation strategy instead (the down part). Extracted features F^t and F^{t-1}, along with few memory banks, are fed to the TCH. TCH compresses them to refine the memory tokens, followed by the segmentation head to achieve a cross-shot segmentation. Meanwhile, the memory $\mathcal{M}_t$ is encoded and stored in a special memory bank $\mathcal{B}_{scene}$ instead, used to establish a necessary spatial scene understanding in TCH.

Transition Comprehension Module. SAAS builds a **T**ransition **C**ompre**h**ension Module (**TCH**) to firstly associate stored scene information and then integrate adjacent frames to fully comprehend the occurring transition. Specifically, TCH reads out the background scene information from the banks $\mathcal{B}_{cond}$ and $\mathcal{B}_{scene}$. $\mathcal{B}_{scene}$ stores representative memories for the most closed N_s shots. These memories are used to build an entire scene understanding, subsequently integrated into F^t_{l3} via stacked attention layers, attaining F'^t_{l3}. Then, a trainable vector Q_{init} passes through the module, sufficiently interacts with the features of the previous frame and the current frame to comprehend the current transition:

$$Q^n_i = \text{Attn}(\text{Attn}(Q^{n-1}_i, F'^t_{l3}), F^{t-1}_{l3})), \quad (2)$$

where $Q^0_i = Q_{init}$, $n = \{1, 2, ..., N_2\}$. Attn represents a standard attention layer (Vaswani et al. 2017), consisting of a multi-head cross-attention, a multi-head self-attention, and a feed forward layer following previous ViT works (Dosovitskiy et al. 2020; Liu et al. 2021b) with a RoPE positional encoding (Su et al. 2024). To validate the process of transition state modeling, we incorporate two additional auxiliary objectives: presence prediction and bounding box regression. Presence prediction requires the model to predict the presence of the object on the next frame from the transition state representation Q_i, supervised by a BCE loss $\mathcal{L}_{exis}$. For the bounding box regression objective, the model learns a mapping from the previous bounding box and Q_i to the post-transition bounding box, adopting a MCE loss $\mathcal{L}_{box}$. Simple MLP stacks suffice for these objectives.

Subsequently, an attention-based aggregator is introduced to decode the transition state Q_i to refine the previous memory $\mathcal{M}^{t-1}_{adj}$. This decoding strategy ensures seamless compatibility with SAM2's well-trained segmentation head. The final refined memories are concatenated with memories from $\mathcal{B}_{cond}$ and $\mathcal{B}_{local}$ and fed to SAM2's memory attention module to prepare the features to be segmented $F^t_{tb\_seg}$.

3.4 Local Memory Bank

In a significant proportion of transitions, local object details can serve as critical segmentation cues, like the clothing of a person or the painted markings on a vehicle. Previous VOS methods struggle to actively capture and recognize such fine-grained features. Informed by such an observation, SAAS introduces a local memory bank $\mathcal{B}_{local}$ to capture and store the target's local details. Inspired by previous works (Song et al. 2019; Liang et al. 2022; Lyu, Zhong, and Zhao 2024), SAAS constructs a minimum spanning tree (MST) on the masked deepest feature map of the conditional frame $M_0 \odot F^0_{l3}$ to simultaneously preserve semantic clustering and spatial structural information. By pruning low-weight edges in the tree, the target is unsupervisedly

Dataset	#Videos	#Objects	#Masks	#Shots	Trans. Frequency	Obj. Categories	Available
YouMVOS	200	492	431.0K	13.4K	0.222/s*	4	✗
YouMVOS-test	30	78	64.6K*	2.4K	0.222/s*	4	✗
Cut-VOS (ours)	100	174	10.2K	648	0.346/s	11	✓

Table 1: The basic statistics for the Cut-VOS benchmark. * denotes the number is estimated via the corresponding description in the paper. Cut-VOS has 1.6× higher transition frequency and 3× more categories than the YouMVOS test split.

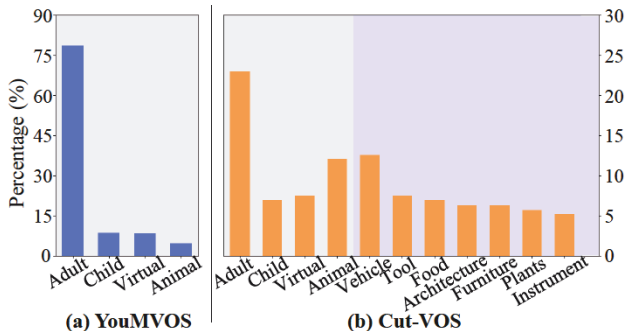


Figure 5: Comparison of object categories. Cut-VOS contains 4 categories in YouMVOS and 7 new categories.

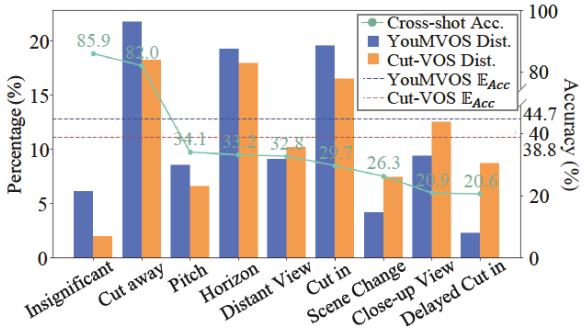


Figure 6: The average accuracies of different transition types on the SAM2-B+ model and their distribution across two benchmarks. The drop in expected accuracy shows Cut-VOS's more challenging nature.

partitioned into multiple semantically coherent sub-regions on a low-resolution map. SAAS further adopts the center point of each partition as a positive point prompt, the rest as the negative to segment these sub-regions and extract corresponding fine-grained features at a high resolution. These detailed features are compressed as complementary object pointers and preserved in the local memory bank $\mathcal{B}_{local}$, which is leveraged to guide the cross-shot segmentation when a transition is detected. Notably, we set a proportion threshold τ_p (2.5% in a common setting) to filter out too small objects, preventing over-partitioning them.

4 Cut-VOS Benchmark

4.1 Video Collection and Annotation

The new challenging multi-shot video object segmentation (MVOS) benchmark, Complex Multi-shot Video Object Segmentation (Cut-VOS), collects large amounts of high-quality multi-shot videos from mainstream community media. The videos and objects are carefully selected to ensure the data samples are unambiguous. The detailed object and transition distributions are shown in Figure 5 and Figure 6.

For mask annotation, our research team organizes and trains a cohort of highly responsible annotators and validators, establishing a robust annotation pipeline. Each annotated video undergoes a dual-review verification to ensure annotation quality assurance. For videos that are discovered with uncertain object correlations, we reconvened discussions to determine whether to keep or filter them.

4.2 Dataset Statistics

Overall, Cut-VOS contains 100 videos, 174 annotated objects, and 10.2K high-quality masks, as shown in Table 1. Cut-VOS outperforms the existing YouMVOS-test in three main aspects: 1) More videos and objects representing more diverse scenarios. 2) Carefully screened, multiple types of transitions with a 1.6 times higher frequency reaching 0.346/s, which makes the Cut-VOS more challenging. 3) 11 diversiform object categories which cover the YouMVOS as depicted in Figure 5, containing 62% actors and 38% static objects. These characteristics make the Cut-VOS benchmark more complex and better aligned with real-world scenarios.

4.3 Transition Analysis

To better analyze the latent challenges in the MVOS task, we classify all shot transitions into 9 different categories: *cut in, cut away, delayed cut in* as existence types, and *close up view, distant view, pitch transformation, horizon transformation, scene change, insignificance* as view types. In specific cases, we allow the coexistence of an existence type and a view type in one transition. Please refer to the appendix for detailed explanations and visualized examples.

We test the tracking accuracy on different transition types with the SAM2-B+ model to pinpoint existing bottlenecks. As shown in Figure 6, SAM2 performs well on *cut away* and *insignificance*, shows moderate competencies on *pitch* and *horizon* types, but drops ruinously on *delayed cut-in*, *close-up view*, and *scene change* types (lower than 27%). The observation indicates that previous methods can recognize the object disappearing, but struggle with matching targets with abrupt visual appearance and absolute position shifts. Cut-VOS filters out simple *insignificance* and long duration *cut away*, involving more difficult transitions to make the benchmark more challenging. Compared to YouMVOS, the significant decrease of $\mathbb{E}_{Acc}$ (44.7% to 38.8%) reflects the challenges brought by screened complex transitions.

Method	Venue	Param.(M)	FPS	YouMVOS				Cut-VOS			
				$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}_t$	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}_t$
XMem	ECCV'22	62.2	**45**	61.7	62.1	61.9	54.2	48.4	51.4	49.9	35.5
DEVA	ICCV'23	<u>61.2</u>	37	63.3	64.5	63.9	55.2	47.3	50.8	49.1	35.3
Cutie	CVPR'24	**35.0**	<u>40</u>	67.3	68.1	67.7	63.4	51.0	53.6	52.3	40.8
Cutie*	CVPR'24	**35.0**	<u>40</u>	67.9	68.8	68.4	64.7	50.0	52.7	51.4	40.0
SAM2-B+	ICLR'25	80.9	22	67.6	67.6	67.6	63.7	54.0	56.4	55.2	47.2
SAM2-L	ICLR'25	224.0	15	69.9	70.3	70.1	68.5	58.3	60.6	59.4	50.7
SAM2-B+*	ICLR'25	80.9	22	68.7	69.1	68.9	64.1	53.9	55.9	54.9	46.8
SAM2-L*	ICLR'25	224.0	15	69.7	70.7	70.2	68.4	57.6	60.3	58.9	50.4
Cutie+TMA	-	**35.0**	<u>40</u>	69.1	70.0	69.6	65.4	52.0	55.0	53.5	43.1
SAAS-B+ (Ours)	AAAI'26	92.5	21	<u>73.4</u>	<u>73.7</u>	<u>73.5</u>	<u>68.9</u>	<u>59.4</u>	<u>61.9</u>	<u>60.7</u>	<u>53.1</u>
SAAS-L (Ours)	AAAI'26	235.6	14	**74.0**	**74.4**	**74.2**	**69.6**	**60.5**	**63.6**	**62.0**	**54.0**

Table 2: Main results on YouMVOS and Cut-VOS benchmarks. * denotes the model is directly trained on the YTVOS dataset without extra data augmentation. Bold and underlined indicate the best and the second-best performance in the tested methods.

ID	$\mathcal{B}_{local}$	TMA	TCH	$\mathcal{J}\&\mathcal{F}$	$\mathcal{J}_t$
I	✗	✗	✗	55.2	47.2
II	✓	✗	✗	57.6	49.4
III	✗	✓	✗	58.0	50.7
IV	✓	✓	✗	58.8	52.0
V	✗	✓	✓	<u>60.1</u>	<u>52.8</u>
VI	✓	✓	✓	**60.7**	**53.1**

Table 3: The ablation study on different modules.

5 Experiments

Benchmark Setting. We benchmark the proposed SAAS and existing methods on Cut-VOS and YouMVOS under the semi-supervised VOS setting. Following previous works (Ding et al. 2023b; Ying et al. 2025), we compute $\mathcal{J}\&\mathcal{F}$ to quantify the region similarity and the contour accuracy of predictions. Besides, we additionally measure the cross-shot tracking capacity by computing region similarity $\mathcal{J}_t$ on post-transition frames. Given the ground truth shot set $\mathcal{S}$, for each shot $\mathcal{S}_i$ we calculate intersection over union (IoU) on the first frame I^i_{tr} and the frame where the object firstly appears I^i_{app} (defined as the first frame too if the object isn't present in the shot) separately, to accommodate different existence transitions, especially *delayed cut in*. Then $\mathcal{J}_t$ is defined as:

$$\mathcal{J}_t = \frac{1}{|\mathcal{S}|}\sum_{i\in|\mathcal{S}|}\frac{\text{IoU}(\hat{M}^i_{tr}, M^i_{tr}) + \text{IoU}(\hat{M}^i_{app}, M^i_{app})}{2}, \quad (3)$$

where M^i denotes the ground truth mask on I_i and $\hat{M}^i$ represents the predicted one. In all of the following experiments, we report both $\mathcal{J}\&\mathcal{F}$ and $\mathcal{J}_t$ as metrics.

Implementation Details. Our method is build upon SAM2 framework, with MAE-pretrained (He et al. 2022) Hiera (Bolya et al. 2023; Ryali et al. 2023) serving as image encoders. We initialize SAM2 original modules with their official weights, firstly freeze other parameters, and train our transition detection module on IACC.3 (Awad et al. 2017) and ClipShots (Tang et al. 2018) shot boundary detection datasets. In the following main training phase, we unfreeze all parameters and train the model for 30 epochs on YTVOS (Xu et al. 2018) with TMA enabled. We set the number of sampling frames as 8 for the base-plus setting and 6 for the large. We enable focal, dice, iou, and CE losses in original SAM2, along with our proposed $\mathcal{L}_{box}$ and $\mathcal{L}_{exis}$. The weights of $\mathcal{L}_{box}$ and $\mathcal{L}_{exis}$ are both set as 0.5. We employ AdamW as the optimizer, with the learning rate decaying from 5e-6 to 5e-7 during training. All experiments are conducted on 4 NVIDIA RTX-A6000 (48G) GPUs.

5.1 Main Results

As shown in Table 2, we conduct exhaustive experiments on existing VOS methods (Cheng and Schwing 2022; Cheng et al. 2023, 2024; Ravi et al. 2024) and our proposed SAAS on YouMVOS and Cut-VOS benchmarks. For SAM2 and SAAS, we test the base-plus setting and the large setting, respectively. The result shows that SAAS-B+ and SAAS-L outperform corresponding SAM2 methods and other existing VOS methods on two benchmarks across both $\mathcal{J}_t$ and $\mathcal{J}\&\mathcal{F}$ metrics, demonstrating its superiority. All reported data are calculated as the average of three runs.

From the table, we observe that training on YTVOS with TMA disabled (marked by *) results in a marginal improvement on YouMVOS (0.7% $\mathcal{J}\&\mathcal{F}$ on Cutie and 1.3% $\mathcal{J}\&\mathcal{F}$ on SAM2-B+). This strategy, however, suppressed methods' performance by 0.3% to 0.9% on Cut-VOS. The finding reveals that some videos from YouMVOS insufficiently represent MVOS difficulties, as they exhibit characteristics similar to conventional single-shot videos. In contrast, directly training on single-shot clips offers diminishing returns for Cut-VOS, which is specifically collected for MVOS.

The experimental result illustrates the effectiveness and robustness of the SAAS method. SAAS-B+ reaches 73.5% $\mathcal{J}\&\mathcal{F}$, 68.9% $\mathcal{J}_t$ on YouMVOS(vs. 67.6% and 63.7%) and 60.7% $\mathcal{J}\&\mathcal{F}$, 53.1% $\mathcal{J}_t$ on Cut-VOS(vs. 55.2% and 47.2%). Compared to SAM2-L, SAAS-L also attains consistent improvements of $\mathcal{J}\&\mathcal{F}$ (from 59.4% to 62.0%) and $\mathcal{J}_t$ (from 50.7% to 54.0%). Notably, SAAS has virtually no degradation in inference speed due to efficient designs. Cutie+TMA method, compared to Cutie and Cutie*, reaches 69.6% (vs. 68.4%) and 53.5% (vs. 52.3%) $\mathcal{J}\&\mathcal{F}$ on two benchmarks, showing great generalization of TMA strategy.

(a) **Delayed cut in** and **Position shift**

(b) **Crowded scene** and **Complex relations**

Figure 7: Qualitative comparison of some representative cases from Cut-VOS between the SAAS and the SAM2 methods. (a) shows a case with a delayed cut in transition and an abrupt position shift of target objects. (b) demonstrates SAAS's better capacity in a crowded scene with complex relations. SAAS coherently segments the target object among ten similar objects.

In the following ablation study, we offer more detailed data to further corroborate TMA and other modules' advantages.

5.2 Ablation Studies

We analyze the validation of our modules via rigorous ablation studies, shown in Table 3. The ablation studies maintain the same implementation as the main experiments, employing the base-plus setting and uniformly tested on the Cut-VOS benchmark. In Table 3, we mainly study the effectiveness of different modules. Compared to baseline model I, the local memory bank $\mathcal{B}_{local}$ and TMA (model II and III) improve $\mathcal{J}\&\mathcal{F}$ by 2.4% and 2.8% respectively, while TMA plus TCH (V) achieves a 4.9% $\mathcal{J}\&\mathcal{F}$ increase. For more ablation studies and hyperparameters analysis in detail, please refer to the appendix.

5.3 Qualitative Results

Figure 7 presents several representative visualized examples and corresponding segmentation results of SAM2 and SAAS models. Case (a) shows a delayed cut in transition, one of the most difficult types, and a classical abrupt position shift of the target object, with a similar appearance distractor appearing at the same position. SAM2 misses the target man (orange) when he reoccurs in shot 2, and incorrectly segments one another man with the same clothing (green) in shot 3, whereas our method successfully segments them. In case (b), we highlight a crowded scene with complex relations between multiple similar objects. SAM2 model struggles to match different instances correctly, leading to flickering predictions. In contrast, by effectively capturing detail cues and establishing scene understanding, SAAS predicts high-quality masks for the object of interest consistently. These examples demonstrate the superiority of our approach in complex multi-shot videos. A few more qualitative analyses are involved in the appendix.

6 Conclusion

We introduce **TMA**, a new training strategy that mitigates MVOS data sparsity by mimicking different transitions on single-shot datasets, and **SAAS**, a new MVOS method performing robust multi-shot segmentation capacity on complex edited videos. Meanwhile, we present a complex multi-shot benchmark, **Cut-VOS**, enabling evaluation and facilitating future research in MVOS. Extensive experiments demonstrate that our proposed strategy and method achieve state-of-the-art performance on MVOS benchmarks.

Limitations. Our method still struggles with extreme appearance changes of the target. For example, the same person with different clothing and hairstyles. The proposed TMA can't simulate this type effectively, and captured local cues may not help. This reflects one of the key challenges for MVOS: the model is required to both match unlike targets and distinguish similar distractors, demanding reducing the reliance on pure visual feature matching and requiring a stronger reasoning ability, which is to be further explored.

Acknowledgments

This project was supported by the National Natural Science Foundation of China (NSFC) under Grant No. 62472104.

References

Awad, G.; Butt, A. A.; Fiscus, J.; Joy, D.; Delgado, A.; Mcclinton, W.; Michel, M.; Smeaton, A. F.; Graham, Y.; Kraaij, W.; et al. 2017. Trecvid 2017: evaluating ad-hoc and instance video search, events detection, video captioning, and hyperlinking. In *TRECVID 2017*. NIST.

Bolya, D.; Ryali, C.; Hoffman, J.; and Feichtenhofer, C. 2023. Window attention is bugged: How not to interpolate position embeddings. arXiv:2311.05613.

Bouyahi, M.; and Ayed, Y. B. 2020. Video scenes segmentation based on multimodal genre prediction. *Procedia Computer Science*, 176: 10–21.

Caelles, S.; Maninis, K.-K.; Pont-Tuset, J.; Leal-Taixé, L.; Cremers, D.; and Van Gool, L. 2017. One-shot video object segmentation. In *CVPR 2017*, 221–230. IEEE.

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6): 679–698.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4): 834–848.

Chen, L.-C.; Papandreou, G.; Schroff, F.; and Adam, H. 2019. Rethinking atrous convolution for semantic image segmentation. arXiv 2017. arXiv:1706.05587.

Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; and Adam, H. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV 2018*, 801–818. Springer.

Cheng, H. K.; Oh, S. W.; Price, B.; Lee, J.-Y.; and Schwing, A. 2024. Putting the object back into video object segmentation. In *CVPR 2024*, 3151–3161. IEEE.

Cheng, H. K.; Oh, S. W.; Price, B.; Schwing, A.; and Lee, J.-Y. 2023. Tracking anything with decoupled video segmentation. In *ICCV 2023*, 1316–1326. IEEE.

Cheng, H. K.; and Schwing, A. G. 2022. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *ECCV 2022*, 640–658. Springer.

Cheng, J.; Tsai, Y.-H.; Hung, W.-C.; Wang, S.; and Yang, M.-H. 2018. Fast and accurate online video object segmentation via tracking parts. In *CVPR 2018*, 7415–7424. IEEE.

Ding, H.; Liu, C.; He, S.; Jiang, X.; and Loy, C. C. 2023a. MeViS: A large-scale benchmark for video segmentation with motion expressions. In *ICCV 2023*, 2694–2703.

Ding, H.; Liu, C.; He, S.; Jiang, X.; Torr, P. H.; and Bai, S. 2023b. MOSE: A new dataset for video object segmentation in complex scenes. In *ICCV 2023*, 20224–20234. IEEE.

Ding, H.; Liu, C.; He, S.; Ying, K.; Jiang, X.; Loy, C. C.; and Jiang, Y.-G. 2025a. MeViS: A multi-modal dataset for referring motion expression video segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(12): 11400–11416.

Ding, H.; Ying, K.; Liu, C.; He, S.; Jiang, X.; Jiang, Y.-G.; Torr, P. H.; and Bai, S. 2025b. MOSEv2: A more challenging dataset for video object segmentation in complex scenes. *arXiv preprint arXiv:2508.05630*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929.

Duarte, K.; Rawat, Y. S.; and Shah, M. 2019. Capsulevos: Semi-supervised video object segmentation using capsule routing. In *ICCV 2019*, 8480–8489. IEEE.

Duke, B.; Ahmed, A.; Wolf, C.; Aarabi, P.; and Taylor, G. W. 2021. Sstvos: Sparse spatiotemporal transformers for video object segmentation. In *CVPR 2021*, 5912–5921. IEEE.

Fang, X.; Mao, K.; Duan, H.; Zhao, X.; Li, Y.; Lin, D.; and Chen, K. 2024. Mmbench-video: A long-form multi-shot benchmark for holistic video understanding. *Advances in Neural Information Processing Systems*, 37: 89098–89124.

Han, J.; Yang, L.; Zhang, D.; Chang, X.; and Liang, X. 2018. Reinforcement cutting-agent learning for video object segmentation. In *CVPR 2018*, 9080–9089. IEEE.

Han, M.; Yang, L.; Chang, X.; and Wang, H. 2023. Shot2story20k: A new benchmark for comprehensive understanding of multi-shot videos. arXiv:2312.10300.

Hassanien, A.; Elgharib, M.; Selim, A.; Bae, S.-H.; Hefeeda, M.; and Matusik, W. 2017. Large-scale, fast and accurate shot boundary detection through spatio-temporal convolutional neural networks. *arXiv preprint arXiv:1705.03281*.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *CVPR 2022*, 16000–16009. IEEE.

Hu, P.; Wang, G.; Kong, X.; Kuen, J.; and Tan, Y.-P. 2018. Motion-guided cascaded refinement network for video object segmentation. In *CVPR 2018*, 1400–1409. IEEE.

Huang, X.; Xu, J.; Tai, Y.-W.; and Tang, C.-K. 2020. Fast video object segmentation with temporal aggregation network and dynamic template matching. In *CVPR 2020*, 8879–8889. IEEE.

Jabri, A.; Owens, A.; and Efros, A. 2020. Space-time correspondence as a contrastive random walk. *Advances in Neural Information Processing Systems*, 33: 19545–19560.

Jacobs, A.; Miene, A.; Ioannidis, G. T.; and Herzog, O. 2004. Automatic Shot Boundary Detection Combining Color, Edge, and Motion Features of Adjacent Frames. In *TRECVID 2004*, volume 2004, 197–206. NIST.

Ji, S.; Xu, W.; Yang, M.; and Yu, K. 2012. 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1): 221–231.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *ICCV 2023*, 4015–4026. IEEE.

Krishna, R.; Hata, K.; Ren, F.; Fei-Fei, L.; and Carlos Niebles, J. 2017. Dense-captioning events in videos. In *ICCV 2017*, 706–715. IEEE.

Liang, Z.; Wang, T.; Zhang, X.; Sun, J.; and Shen, J. 2022. Tree energy loss: Towards sparsely annotated semantic segmentation. In *CVPR 2022*, 16907–16916. IEEE.

Lin, H.; Qi, X.; and Jia, J. 2019. Agss-vos: Attention guided single-shot video object segmentation. In *ICCV 2019*, 3949–3957. IEEE.

Liu, C.; Ding, H.; Ying, K.; Hong, L.; Xu, N.; Yang, L.; Fan, Y.; Gao, M.; Chen, J.; Miao, Y.; et al. 2025. LSVOS 2025 Challenge Report: Recent Advances in Complex Video Object Segmentation. *arXiv preprint arXiv:2510.11063*.

Liu, X.; Hu, Y.; Bai, S.; Ding, F.; Bai, X.; and Torr, P. H. 2021a. Multi-shot temporal event localization: a benchmark. In *CVPR 2021*, 12596–12606. IEEE.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021b. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV 2021*, 10012–10022. IEEE.

Lyu, H.; Zhong, T.; and Zhao, S. 2024. Gtms: A gradient-driven tree-guided mask-free referring image segmentation method. In *ECCV 2024*, 288–304. Springer.

Perazzi, F.; Khoreva, A.; Benenson, R.; Schiele, B.; and Sorkine-Hornung, A. 2017. Learning video object segmentation from static images. In *CVPR 2017*, 2663–2672. IEEE.

Qian, X.; Liu, G.; and Su, R. 2006. Effective fades and flashlight detection based on accumulating histogram difference. *IEEE Trans. Circuit Syst. Video Technol.*, 16(10): 1245–1258.

Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; et al. 2024. Sam 2: Segment anything in images and videos. arXiv:2408.00714.

Ryali, C.; Hu, Y.-T.; Bolya, D.; Wei, C.; Fan, H.; Huang, P.-Y.; Aggarwal, V.; Chowdhury, A.; Poursaeed, O.; Hoffman, J.; et al. 2023. Hiera: A hierarchical vision transformer without the bells-and-whistles. In *ICML 2023*, 29441–29454. PMLR.

Song, L.; Li, Y.; Li, Z.; Yu, G.; Sun, H.; Sun, J.; and Zheng, N. 2019. Learnable tree filter for structure-preserving feature transform. *Advances in Neural Information Processing Systems*, 32.

Soucek, T.; and Lokoc, J. 2024. Transnet v2: An effective deep network architecture for fast shot transition detection. In *ACMMM 2024*, 11218–11221. ACM.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Tang, S.; Feng, L.; Kuang, Z.; Chen, Y.; and Zhang, W. 2018. Fast video shot transition localization with deep structured models. In *ACCV 2018*, 577–592. Springer.

Tu, C.; Zhang, Z.; Liu, Z.; and Sun, M. 2017. TransNet: Translation-Based Network Representation Learning for Social Relation Extraction. In *IJCAI 2017*, 2864–2870. IJCAI Inc.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Wang, T.; Feng, N.; Yu, J.; He, Y.; Hu, Y.; and Chen, Y.-P. P. 2021. Shot boundary detection through multi-stage deep convolution neural network. In *MMM 2021*, 456–468. Springer.

Wang, Z.; Xu, J.; Liu, L.; Zhu, F.; and Shao, L. 2019. Ranet: Ranking attention network for fast video object segmentation. In *ICCV 2019*, 3978–3987. IEEE.

Wei, D.; Kharbanda, S.; Arora, S.; Roy, R.; Jain, N.; Palrecha, A.; Shah, T.; Mathur, S.; Mathur, R.; Kemkar, A.; et al. 2022. Youmvos: an actor-centric multi-shot video object segmentation dataset. In *CVPR 2022*, 21044–21053. IEEE.

Xiao, H.; Feng, J.; Lin, G.; Liu, Y.; and Zhang, M. 2018. Monet: Deep motion exploitation for video object segmentation. In *CVPR 2018*. IEEE.

Xu, J.; Mei, T.; Yao, T.; and Rui, Y. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR 2016*, 5288–5296. IEEE.

Xu, N.; Yang, L.; Fan, Y.; Yue, D.; Liang, Y.; Yang, J.; and Huang, T. 2018. Youtube-vos: A large-scale video object segmentation benchmark. arXiv:1809.03327.

Ying, K.; Ding, H.; Jie, G.; and Jiang, Y.-G. 2025. Towards omnimodal expressions and reasoning in referring audio-visual segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 22575–22585.

Ying, K.; Hu, H.; and Ding, H. 2025. MOVE: Motion-guided few-shot video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11632–11642.

Ying, K.; Wang, Z.; Bai, C.; and Zhou, P. 2022. Isda: Position-aware instance segmentation with deformable attention. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2619–2623. IEEE.

Ying, K.; Zhong, Q.; Mao, W.; Wang, Z.; Chen, H.; Wu, L. Y.; Liu, Y.; Fan, C.; Zhuge, Y.; and Shen, C. 2023. Ctvis: Consistent training for online video instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 899–908.

Yu, F.; Koltun, V.; and Funkhouser, T. 2017. Dilated residual networks. In *CVPR 2017*, 472–480. IEEE.

Zhou, L.; Xu, C.; and Corso, J. 2018. Towards automatic learning of procedures from web instructional videos. In *AAAI*. AAAI Press.