

Video Object Segmentation in Panoptic Wild Scenes

Yuanyou Xu^{1,2†}, Zongxin Yang¹, Yi Yang^{1‡}

¹ReLER, CCAI, Zhejiang University

²Baidu Research

{yoxu, zongxinyang, yangyics}@zju.edu.cn

Abstract

In this paper, we introduce semi-supervised video object segmentation (VOS) to panoptic wild scenes and present a large-scale benchmark as well as a baseline method for it. Previous benchmarks for VOS with sparse annotations are not sufficient to train or evaluate a model that needs to process all possible objects in real-world scenarios. Our new benchmark (VIPOSeg) contains exhaustive object annotations and covers various real-world object categories which are carefully divided into subsets of thing/stuff and seen/unseen classes for comprehensive evaluation. Considering the challenges in panoptic VOS, we propose a strong baseline method named panoptic object association with transformers (PAOT), which associates multiple objects by panoptic identification in a pyramid architecture on multiple scales. Experimental results show that VIPOSeg can not only boost the performance of VOS models by panoptic training but also evaluate them comprehensively in panoptic scenes. Previous methods for classic VOS still need to improve in performance and efficiency when dealing with panoptic scenes, while our PAOT achieves SOTA performance with good efficiency on VIPOSeg and previous VOS benchmarks. PAOT also ranks 1st in the VOT2022 challenge. Our dataset and code are available at <https://github.com/yoxu515/VIPOSeg-Benchmark>.

1 Introduction

Video object segmentation (VOS) is a fundamental task in computer vision. In this paper, we focus on semi-supervised video object segmentation, which aims to segment all target objects specified by reference masks in video frames. Although VOS has been well studied in recent years, there are still limitations in previous benchmark datasets. Firstly, previous VOS datasets only provide limited annotations. The annotations of commonly used datasets for VOS, YouTube-VOS [Xu *et al.*, 2018] and DAVIS [Pont-Tuset *et al.*, 2017]

are spatially sparse, with a few objects annotated for most video sequences. Secondly, the classes of YouTube-VOS only include countable thing objects. While in the real world, many scenes may contain dozens of objects and other stuff classes like ‘water’ and ‘ground’. Obviously these datasets can’t cover such scenarios. As a consequence, previous datasets are not able to train VOS models thoroughly and evaluate models comprehensively.

To this end, we study VOS in panoptic scenes as panoptic VOS and present a dataset named Video Panoptic Object Segmentation (VIPOSeg). VIPOSeg is built on VIPSeg [Miao *et al.*, 2022], a dataset for video panoptic segmentation. We re-split the training and validation set and convert the panoptic annotations in VIPSeg to VOS format. Beyond classic VOS, we make thing/stuff annotations for objects available as a new panoptic setting. VIPOSeg dataset is qualified to play the role of a benchmark for panoptic VOS. First, VIPOSeg provides annotations for all objects in scenes. Second, a variety of object categories are included in VIPOSeg. The large diversity of classes and density of objects help to train a model with high robustness and generalization ability for complex real-world applications. For model evaluation, we divide object classes into thing/stuff and seen/unseen subsets. A model can be comprehensively evaluated on these class subsets. In addition, VIPOSeg can also evaluate the performance decay of a model as the number of objects increases.

Challenges also emerge when a model tries dealing with panoptic scenes (Figure 1). The large number of objects causes occlusion and efficiency problem, and various scales and diversity of classes require high robustness. In order to tackle the challenges, we propose a strong baseline method for panoptic object association with transformers (PAOT), which uses decoupled identity banks to generate panoptic identification embeddings for thing and stuff, and uses a pyramid architecture with efficient transformer blocks to perform multi-scale object matching. PAOT achieves superior performance with good efficiency and ranks 1st in both short-term/real-time tracking and segmentation tracks in the VOT2022 challenge [Kristan *et al.*, 2023].

In summary, our contributions are three-fold:

- We introduce panoptic VOS, and present a new benchmark VIPOSeg, which provides exhaustive annotations and includes seen/unseen and thing/stuff classes.

[†]Yuanyou Xu worked on this at his Baidu Research internship.

[‡]Yi Yang is the corresponding author.

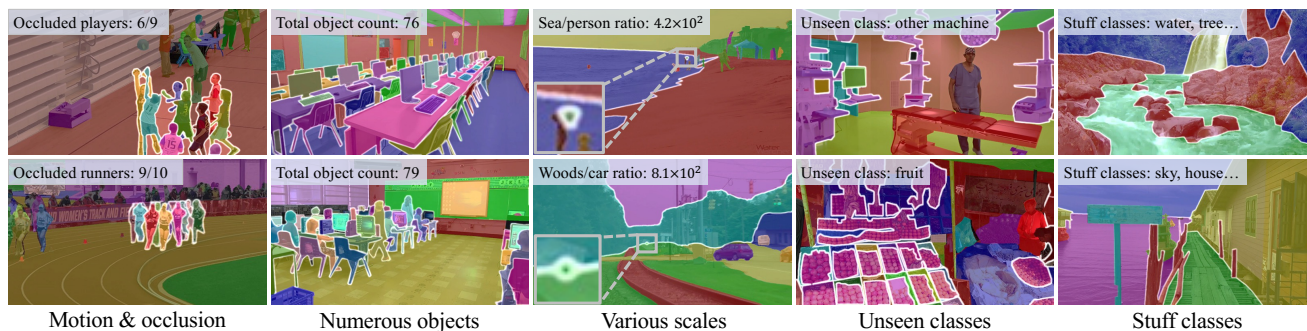


Figure 1: The figures illustrate challenges of video object segmentation in panoptic wild scenes in VIPOSeg dataset. In crowded scenes, motion and occlusion are sometimes extremely complex. In addition, numerous objects are challenging to the efficiency of VOS models. Objects on various scales are also difficult to deal with, especially small objects. As for object classes, VIPOSeg contains seen/unseen classes and thing/stuff classes. VOS models need to not only generalize from seen to unseen classes, but also learn to process both thing and stuff.

- Considering the challenges in panoptic VOS, we propose a strong baseline PAOT, which consists of the decoupled identity banks for thing and stuff, and a pyramid architecture with efficient long-short term transformers.
- Experimental results show that VIPOSeg is more challenging than previous VOS benchmarks, while our PAOT models show superior performance on the new VIPOSeg benchmark as well as previous benchmarks.

2 Related Work

Semi-supervised video object segmentation. As an early branch, online VOS methods [Caelles *et al.*, 2017; Yang *et al.*, 2018; Meinhardt and Leal-Taixé, 2020] fine-tune a segmentation model on given masks for each video. Another promising branch is matching-based VOS methods [Shin Yoon *et al.*, 2017; Voigtlaender *et al.*, 2019; Yang *et al.*, 2020], which constructs the embedding space to measure the distance between a pixel and the given object. STM [Oh *et al.*, 2019] introduces the memory networks to video object segmentation and models the matching as space-time memory reading. Later works [Seong *et al.*, 2020; Cheng *et al.*, 2021c] improve STM by better memory reading strategies. A multi-object identification mechanism is proposed in AOT [Yang *et al.*, 2021a; Yang *et al.*, 2021c; Yang and Yang, 2022] to process all the target objects simultaneously. This strategy is adopted in our framework to model the relationship between multiple objects, and we further propose solutions for other challenges in panoptic scenes.

Multi-scale architectures for VOS. CFBI+ [Yang *et al.*, 2021b] proposes a multi-scale foreground and background integration structure, and a hierarchical multi-scale architecture is proposed in HMMN [Seong *et al.*, 2021]. In this work, we also propose a multi-scale architecture, while the matching is performed sequentially in our pyramid architecture but not individually (CFBI+) or with guidance (HMMN). The design of our method is inspired by general transformer backbones [Wang *et al.*, 2021b; Liu *et al.*, 2021] but ours is for feature matching across multiple frames on both spatial and temporal dimensions but not feature extraction on static images.

Dataset	Task	Video	T/s	Class	Unseen	Stuff	Obj./Video
DAVIS [Pont-Tuset <i>et al.</i> , 2017]	VOS	150	2.9	-	✗	✗	2.51
YouTube-VOS [Xu <i>et al.</i> , 2018]	VOS	4453	4.5	94	✓	✗	1.64
UVO [Wang <i>et al.</i> , 2021a]	OWOS	1200	3.0	open	✓	✗	12.29
OVIS [Qi <i>et al.</i> , 2022]	VOS/VIS	901	12.8	25	✗	✗	5.80
VIPSeg [Miao <i>et al.</i> , 2022]	VPS	3536	4.8	124	✗	✓	13.26*
VIPOSeg	VOS	3149	4.3	125	✓	✓	13.26

Table 1: Detailed comparison of related datasets. Obj./Video stands for the average object number per video. T is the average video duration time. *For VIPSeg, test set is not included when calculating average object number because it is not public.

Video panoptic segmentation. Among the tasks for video segmentation [Zhou *et al.*, 2022; Li *et al.*, 2023], video panoptic segmentation (VPS) [Kim *et al.*, 2020] is also related to our panoptic VOS. VPS methods [Woo *et al.*, 2021; Li *et al.*, 2022; Kim *et al.*, 2022] manage to predict object classes and instances for all pixels in each frame of a video, while in panoptic VOS all objects are defined by reference masks when they first appear. Although they both consider thing and stuff, panoptic VOS is class agnostic and can generalize to arbitrary classes. In addition, most VPS datasets like Cityscapes-VPS [Cordts *et al.*, 2016] and KITTI-STEP [Weber *et al.*, 2021] only cover street scenes with limited object categories. VIPSeg [Miao *et al.*, 2022] is the first large-scale VPS dataset in the wild.

Related datasets. Detailed comparison of related datasets can be found in Table 1, which also covers some datasets beyond VOS. DAVIS [Pont-Tuset *et al.*, 2017] is a small VOS dataset containing 150 videos with sparse object annotations. YouTube-VOS [Xu *et al.*, 2018] is a large-scale VOS dataset containing 4453 video clips and 94 object categories. OVIS dataset [Qi *et al.*, 2022] focuses on heavy occlusion problems in video segmentation, in which the 901 video clips mainly include multiple occluded instances. UVO [Wang *et al.*, 2021a] is for open world object segmentation and has much denser annotations than YouTube-VOS. VIPSeg [Miao *et al.*, 2022] is a large-scale dataset for video panoptic segmentation in the wild. We build our VIPOSeg dataset based on VIPSeg and details are in the following section.

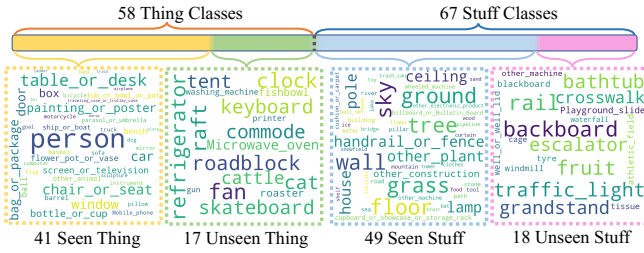


Figure 2: Object classes and class subsets in VIPOSeg.

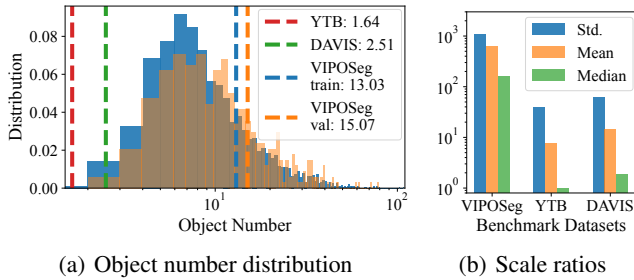


Figure 3: Comparison among VOS benchmark datasets including VIPOSeg, YouTube-VOS (YTB) and DAVIS. Figure (a) shows the object number distribution in VIPOSeg, as well as mean object numbers of other datasets. Figure (b) compares the distribution statistics of scale ratios in different datasets.

3 Benchmark

3.1 Producing VIPOSeg

Exhaustively annotating objects in images is extremely consuming, let alone in video frames. Fortunately, recently VIPSeg dataset [Miao *et al.*, 2022] provides 3536 videos annotated in a panoptic manner. It includes 124 classes consisting of 58 thing classes and 66 stuff classes. We adapt this dataset and build our VIPOSeg dataset based on it.

Splitting dataset and classes. In terms of VIPSeg, the 3536 videos are split into 2,806/343/387 for training, validation and test. We only use training and validation sets in our VIPOSeg (3149 videos in total) because the annotations for test set are private. In order to add unseen classes to validation set, we re-split the videos into new training and validation set. We first sort 58 thing classes and 66 stuff classes respectively by frequency of occurrence. Next, we choose 17 thing classes and 17 stuff classes from the tail as unseen classes. We also split ‘other machine’ into two classes, one for seen and another for unseen (detailed explanation is in supplementary material). Then, videos for validation set are selected by ensuring that enough objects in unseen classes should be included. Last but not least, we remove the annotations of unseen classes in training set. In summary, there are four subsets of 125 classes including 41/17 seen/unseen thing classes and 49/18 seen/unseen stuff classes (Figure 2).

Creating and correcting annotations. In order to generate reference masks for VOS, we convert the panoptic annotations into object index format and then select the masks that appear the first time in each video as reference masks. To dis-

tinguish thing/stuff and seen/unseen classes, we also record the class mapping from object index to class index for each video. The class mapping enables us to calculate the evaluation metrics on seen/unseen and thing/stuff classes. Another problem is that the mask annotations in original VIPSeg are noisy, especially in the edges of objects. To ensure the correctness of evaluation, we manually recognize low-quality annotations and cleaned their noises in validation set.

Settings of panoptic VOS. The panoptic VOS task comes along with the rich and dense annotations. In panoptic VOS, models are trained with panoptic data. Besides, extra annotations indicating whether an object is thing or stuff are available in both training and test. Previous classic VOS, where only spatially sparse annotated data is used for training and test, can be regarded as a simplified version of panoptic VOS. Our method PAOT provides solutions for both panoptic/classic settings (Section 4).

3.2 Significance of VIPOSeg

As a new benchmark dataset, VIPOSeg not only complements the deficiency of previous datasets but also surpasses them by a large margin in class diversity and object density. VIPOSeg has 4× denser annotations, 20× more videos than DAVIS, and 6× denser annotations than YouTube-VOS (Table 1 and Figure 3(a)). Denser annotations of panoptic scenes also includes objects on more diversified scales, with almost 30× larger mean and 6400× larger variance of scale ratios than YouTube-VOS (Figure 3(b)). More importantly, VIPOSeg contains stuff classes which never appear in previous VOS datasets (Table 1).

3.3 Challenges in VIPOSeg

With much denser object annotation and more diverse classes, challenges also emerge in the VIPOSeg dataset (Figure 1).

Motion and occlusion. Although previous datasets also include objects with motion and occlusion, they are not as challenging as VIPOSeg. The number of objects in VIPOSeg can be so large that the occlusion can be intractable.

Numerous objects. Another challenge that comes with the large number of objects is efficiency. Figure 1 column two shows scenes with numerous objects and Figure 3(a) shows the distribution of object number in VIPOSeg. VOS models will need more memory and be slower when the number of objects becomes larger. According to our experimental results in Table 3, CFBI+ [Yang *et al.*, 2021b] runs at 2 FPS and consumes over 30 GB memory when evaluated on VIPOSeg.

Various scales. Since the scenes are exhaustively annotated, objects on all scales are included. Figure 3(b) shows the mean, median and standard deviation of the scale ratios in VOS benchmarks. The scale ratio is defined as the ratio of the pixel numbers of the largest and the smallest objects in a frame. The scale ratios of frames in VIPOSeg have much larger mean value and variance than previous benchmarks.

Unseen classes. We deliberately wipe out the annotations of some classes in training set to make them unseen in validation set. Generalizing from seen to unseen is a common problem for most deep models. It is not easy to narrow the performance gap between seen and unseen.

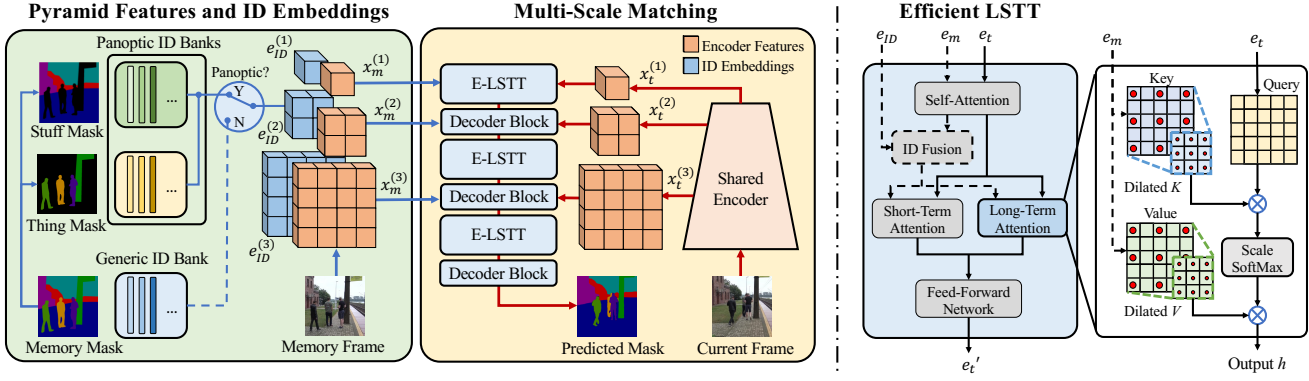


Figure 4: Left part illustrates the generation of generic/panoptic ID embeddings and the pyramid architecture for multi-scale object matching. Right part shows the detailed structure of efficient long short-term transformer blocks and dilated long-short term attention in it.

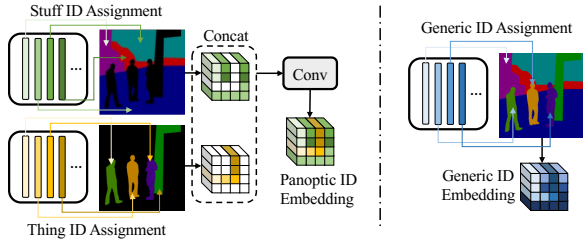


Figure 5: Detailed illustration of panoptic ID embedding generation (left) and generic ID embedding generation (right).

Stuff classes. Previous VOS datasets never contain stuff classes while VIPOSeg does. One may be curious about whether a VOS model can track the mask of flowing water like ‘sea wave’ (Figure 1 column three) and ‘waterfall’ (Figure 1 column four). The answer can be found in VIPOSeg.

4 Method

In the face of above challenges, we develop a method Panoptic Object Association with Transformers (PAOT), which is not only designed for panoptic VOS but also compatible with classic VOS. PAOT consists of following designs, 1) For the motion and occlusion problem, we employ multi-object association transformers (AOT) [Yang *et al.*, 2021a] as the base framework. 2) For objects on various scales, a pyramid architecture is proposed to incorporate multi-scale features into the matching procedure. 3) For the thing/stuff objects in panoptic scenes, we decouple a single ID bank into two separate ID banks for thing and stuff to generate panoptic ID embeddings. 4) For the efficiency problem caused by numerous objects, an efficient version of long-short term transformers (E-LSTT) is proposed to balance performance and efficiency.

4.1 Pyramid Architecture

A pyramid architecture (Figure 4) is proposed in PAOT to perform matching on different scales. The scales are determined by the features $x^{(i)}$ from the encoder. For memory/reference frames who have masks, the mask information is encoded in ID embeddings $e_{ID}^{(i)}$ by assigning ID vectors in the ID bank.

Each ID vector is corresponding to an object so the ID embedding contains information of all objects. The ID assignment can be regarded as a function which maps a one-hot label of multiple objects to a high dimensional embedding. Each scale i has an individual ID bank to generate the ID embedding to maintain rich target information. The ID embedding is fused with the memory frame embedding $e_m^{(i)}$ as key and value, waiting for the query of later frames.

For a current frame without a mask, the E-LSTT module is responsible for performing matching between the embeddings of the current frame $e_t^{(i)}$ and memory/reference frames $e_m^{(i)}$. Next, the decoder block is able to decode the matching information and incorporate the features on the larger scale $x_t^{(i+1)}$. The matching and decoding process is in a recursive manner from the current scale to the next scale,

$$e_t^{(i)'} = T_E^{(i)}(e_t^{(i)}, e_m^{(i)}, e_{ID}^{(i)}),$$

$$e_t^{(i+1)} = R^{(i)}(s^+(e_t^{(i)'}) + x_t^{(i+1)}),$$

where $T_E^{(i)}(\cdot)$ is the E-LSTT module, $s^+(\cdot)$ is the up-sampling function and $R^{(i)}(\cdot)$ is the decoder block (implemented as residual convolutional blocks [He *et al.*, 2016]).

4.2 Generation of Panoptic ID Embeddings

For panoptic VOS, we generate panoptic ID embedding from thing and stuff mask on each scale (Figure 4, 5). Previous VOS datasets and methods only consider the countable thing objects but omit stuff objects. Although thing objects and stuff objects can be treated equally in a unified manner in classic VOS methods, the difference between stuff and thing should not be ignored. Considering this, we decouple the ID bank into two separate ID banks for thing and stuff objects respectively. We aim to obtain more discriminative ID embeddings for thing objects while more generic ID embeddings for stuff objects, especially unseen stuff objects.

The label of a frame y is first decomposed into thing label y_{th} and stuff label y_{st} . The thing objects are assigned with ID vectors from thing ID bank and stuff objects are assigned with ID vectors from stuff ID bank. Last, the thing and stuff

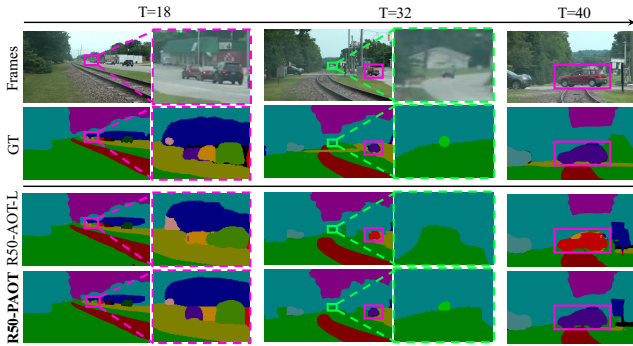


Figure 6: Comparison between AOT and PAOT models on a video sequence with objects on various scales. Small objects in boxes are enlarged for better viewing.

ID embeddings are concatenated and fed into the aggregation module (implemented as convolutional layers) to obtain the panoptic ID embedding on scale i ,

$$e_{ID}^{(i)} = \text{Conv}(\text{Cat}(ID_{th}^{(i)}(y_{th}), ID_{st}^{(i)}(y_{st}))).$$

4.3 Efficient Long Short-Term Transformers

Long-short term transformers (LSTT) are proposed in AOT [Yang *et al.*, 2021a] for object matching. Directly using LSTT in the pyramid structure causes efficiency problem due to the larger scales, and the problem will become more serious in panoptic scenes due to numerous objects.

The long-term attention dominates the computational cost of LSTT because the attention may involve multiple memory frames. In order to cut down the computational cost, we use single-head rather than multi-head attention for the long-term memory. Inspired by [Wang *et al.*, 2021b], we further apply the dilated attention where the key and value are down-sampled in the long-term attention on large scales (Figure 4). More details can be found in supplementary material.

5 Experiment

5.1 Implementation Details

Model settings. The encoder backbones of PAOT models are chosen in ResNet-50 [He *et al.*, 2016] and Swin Transformer-Base [Liu *et al.*, 2021]. As for multi-scale object matching, we set E-LSTT in four scales $16\times$, $16\times$, $8\times$, $4\times$ to be 2,1,1,0 layers respectively (4 layers in total). It should be noted that we do not use the $4\times$ scale feature maps for object matching but only for decoding considering the computational burden, and instead duplicate the $16\times$ features twice.

Training procedure. The training procedure consists of two steps: (1) pre-training on the synthetic video sequences generated by static image datasets [Everingham *et al.*, 2010; Lin *et al.*, 2014; Cheng *et al.*, 2014; Shi *et al.*, 2015; Hariharan *et al.*, 2011] by randomly applying multiple image augmentations [Oh *et al.*, 2018]. (2) main training on the real video sequences by randomly applying video augmentations [Yang *et al.*, 2020]. The datasets for training include DAVIS 2017 (D) [Pont-Tuset *et al.*, 2017], YouTube-VOS 2019 (Y) [Xu *et al.*, 2018] and our VIPOSeg (V). Models pre-trained

with BL-30K [Cheng *et al.*, 2021b] are marked with * (for STCN [Cheng *et al.*, 2021c]).

During training, we use 4 Nvidia Tesla A100 GPUs, and the batch size is 16. For pre-training, we use an initial learning rate of 4×10^{-4} for 100,000 steps. For main training, the initial learning rate is 2×10^{-4} , and the training steps are 100,000. The learning rate gradually decays to 1×10^{-5} in a polynomial manner [Yang *et al.*, 2020].

Task settings. For panoptic setting, V is used for training and evaluation. PAOT models with panoptic ID are marked with Pano-ID, otherwise generic ID is used. Note that PAOT with generic ID is compatible with classic VOS. For classic setting, Y+D are used for training and evaluation. Training with Y+D+V is mainly for classic setting and V is regarded as auxiliary data.

5.2 Evaluation Results on VIPOSeg

Evaluation metrics. For a new benchmark, it is crucial to choose proper metrics to evaluate the performance. We set eight separate metrics including **four mask IoUs** for seen/unseen thing/stuff ($M_s^{th}/M_u^{th}/M_s^{sf}/M_u^{sf}$), and **four boundary IoUs** [Cheng *et al.*, 2021a] for seen/unseen thing/stuff ($B_s^{th}/B_u^{th}/B_s^{sf}/B_u^{sf}$) respectively. The overall performance \mathcal{G} is the average of these eight metrics. Moreover, four average metrics are calculated to indicate the average performance on thing/stuff ($\mathcal{G}^{th}/\mathcal{G}^{sf}$) and seen/unseen ($\mathcal{G}_s/\mathcal{G}_u$). The results with these metrics can be found in Table 2. Except for these standard metrics, there is also a special metric on VIPOSeg, the decay constant λ . It is in charge of evaluating the robustness of models in crowded scenes. More details can be found in later **Crowd decay** section.

Panoptic setting. We train AOT [Yang *et al.*, 2021a] and PAOT models with VIPOSeg (V) as in panoptic VOS. The evaluation results are in middle of Table 2. Both the pyramid architecture and panoptic IDs in PAOT are beneficial to panoptic scenes. First, our PAOT model with generic IDs surpasses AOT by 1.1% with the same R-50 backbone, which shows the improvement of the pyramid architecture. Second, the PAOT models with panoptic IDs have higher overall performance than PAOT models with generic IDs. Their difference is mainly on the metrics of unseen and stuff. R50-PAOT (Pano-ID) have around **2% higher mask IoU M_u^{sf} and 1.5% higher boundary IoU B_u^{sf} on unseen stuff** objects than generic R50-PAOT. Therefore, decoupling the ID bank into thing and stuff is beneficial to learn more robust stuff ID vectors which generalize better on unseen objects.

Classic setting. We test several representative methods including CFBI+ [Yang *et al.*, 2021b], STCN [Cheng *et al.*, 2021c], AOT [Yang *et al.*, 2021a] and our PAOT on VIPOSeg validation set. The evaluation results are in top of Table 2. These models are trained with Y+D. First, the overall IoU scores of previous methods are around 73.0. Compared with them, our PAOT models are above 75.0, which **surpass previous methods by over 2%**. Qualitative results of these methods are in Figure 7. Second, previous methods like CFBI+ and STCN perform poorly on thing IoU \mathcal{G}^{th} . By contrast, multi-object association based methods like AOT

Methods	Training	Average IoU					Mask IoU				Boundary IoU				λ
		\mathcal{G}	\mathcal{G}_s	\mathcal{G}_u	\mathcal{G}^{th}	\mathcal{G}^{sf}	Seen/Unseen	Thing/Stuff	Thing	Stuff	Thing	Stuff	Thing	Stuff	
CFBI+ [Yang <i>et al.</i> , 2021b]	Y+D	72.1	73.0	71.3	68.3	76.0	69.6	69.4	80.4	77.3	67.7	66.7	74.4	71.7	1.42
STCN [Cheng <i>et al.</i> , 2021c]	Y+D	72.4	73.8	71.1	68.4	76.5	70.9	68.3	80.8	78.3	69.0	65.5	74.5	72.3	1.03
STCN* [Cheng <i>et al.</i> , 2021c]	Y+D	72.5	73.6	71.4	69.0	76.0	71.2	69.1	80.1	78.2	69.4	66.4	73.9	72.0	1.08
R50-AOT-L [Yang <i>et al.</i> , 2021a]	Y+D	73.7	74.8	72.6	72.1	75.4	73.9	72.6	79.7	77.2	72.1	69.8	73.6	70.9	0.88
SwinB-AOT-L [Yang <i>et al.</i> , 2021a]	Y+D	73.3	74.5	72.2	72.0	74.6	74.4	71.9	78.8	76.9	72.5	69.3	72.2	70.6	0.92
R50-PAOT	Y+D	75.4	76.5	74.3	74.2	76.5	76.0	74.7	80.7	78.3	74.2	72.0	74.9	72.2	0.84
SwinB-PAOT	Y+D	75.3	76.3	74.4	74.7	76.0	76.4	75.0	80.0	77.9	74.7	72.5	74.1	72.1	0.87
R50-AOT-L [Yang <i>et al.</i> , 2021a]	V	76.4	78.0	74.8	74.2	78.6	76.8	73.8	82.9	80.0	75.0	71.2	77.3	74.1	0.78
R50-PAOT	V	77.5	79.1	75.8	75.9	79.0	78.2	75.7	83.6	79.9	76.5	73.2	78.2	74.4	0.77
R50-PAOT (Pano-ID)	V	77.9	79.0	76.8	76.0	79.8	78.1	76.0	83.3	81.8	76.4	73.3	77.9	75.9	0.76
SwinB-PAOT	V	78.0	79.5	76.5	76.4	79.6	78.8	76.0	83.7	80.8	77.2	73.7	78.3	75.5	0.70
SwinB-PAOT (Pano-ID)	V	78.2	79.5	76.9	76.3	80.1	78.6	75.9	83.9	81.7	76.9	73.7	78.5	76.2	0.70
R50-AOT-L [Yang <i>et al.</i> , 2021a]	Y+D+V	76.5	77.9	75.0	74.3	78.6	76.7	74.1	82.8	80.2	74.9	71.6	77.2	74.2	0.80
R50-PAOT	Y+D+V	77.4	78.4	76.4	75.9	78.8	77.5	76.6	82.9	80.3	75.8	73.9	77.5	74.7	0.79
SwinB-PAOT	Y+D+V	77.9	79.3	76.5	76.3	79.5	78.8	75.8	83.3	81.2	77.2	73.5	77.8	75.7	0.73

Table 2: Evaluation results on VIPOSeg validation set. Training datasets include YouTube-VOS (Y), DAVIS (D) and VIPOSeg (V). * denotes that models are pre-trained with BL-30K. λ is the decay constant.

and PAOT improve thing IoU a lot because the simultaneous multi-object propagation with ID mechanism is capable of modeling multi-object relationship such as occlusion.

Boosting performance by panoptic training. The overall IoU of AOT or PAOT rises around 3% after replacing the training data from Y+D to V. There is a huge gap between models trained with and without VIPOSeg. The VIPOSeg training set enables the models to learn panoptic object association and to generalize in more complex scenes and classes. Besides, panoptic training data also benefits VOS models on previous classic VOS benchmarks, as shown in Table 4.

Crowd decay. Dense annotations in VIPOSeg enable us to evaluate the performance of models under scenes with different amounts of objects. Here we present the crowd decay evaluation. We model the problem as exponential decay, $\mathcal{G}(n) = e^{-\lambda n/s}$ where $s = 100$ is a scaling factor and λ is the decay constant that reflects how fast the performance \mathcal{G} drops when object number n increases. The IoU for each object number n is collected to estimate λ by least square. We show the decay constants for different methods and models in Table 2 and plot the decay curves in Figure 8. The results show that multi-object association methods (AOT, PAOT are around 0.8) have lower decay constants than other methods (CFBI+, STCN are above 1.0). SwinB-PAOT trained with VIPOSeg achieves the lowest decay constant 0.70, which means it deals with crowded scenes better than other models.

Speed and memory. For all methods evaluated on VIPOSeg, we record the FPS and maximal memory space they consume during evaluation, which can be found in Table 3. The measure of FPS and memory is on Nvidia Tesla A100 GPU. CFBI+ runs at 2 FPS while STCN and AOT are at around 10 FPS. This fact shows VIPOSeg benchmark is very challenging in model efficiency. STCN runs faster with

Methods	IDs	\mathcal{G}	Total FPS	Memory/GB
CFBI+ [Yang <i>et al.</i> , 2021b]	-	72.1	2.01	33.13
STCN [Cheng <i>et al.</i> , 2021c]	-	72.5	11.60	14.17
R50-AOT-L [Yang <i>et al.</i> , 2021a]	10	73.7	11.30	12.35
SwinB-AOT-L [Yang <i>et al.</i> , 2021a]	10	73.3	9.13	12.21
R50-PAOT[‡]	10	77.5	10.45	11.04
SwinB-PAOT[‡]	10	78.0	8.35	11.18
R50-PAOT[‡] (Pano-ID)	10+5	77.9	11.23	10.58
SwinB-PAOT[‡] (Pano-ID)	10+5	78.2	8.48	10.72
R50-PAOT [‡]	15	77.4	12.60	9.67
R50-PAOT [‡]	20	77.4	13.32	8.34
R50-PAOT [‡]	30	76.5	14.16	7.96

Table 3: Speed and memory consumption of different methods on VIPOSeg validation set. [‡] denotes models trained with V rather than Y+D. Memory is the maximal GPU memory used by the method.

more memory while AOT and PAOT run slightly slower with less memory. However, all of these models demand over 11 GB memory, which leaves a large space for further improvement. A larger ID capacity and better memory strategy may help with the efficiency problem.

5.3 Results on YouTube-VOS and DAVIS

The evaluation results on YouTube-VOS [Xu *et al.*, 2018] and DAVIS [Perazzi *et al.*, 2016; Pont-Tuset *et al.*, 2017] are listed in Table 4. More detailed tables can be found in supplementary material. For models trained with Y+D, our PAOT model with Swin Transformer-Base backbone achieves SOTA performance on all benchmarks. Adding VIPOSeg to training can further boost performance.

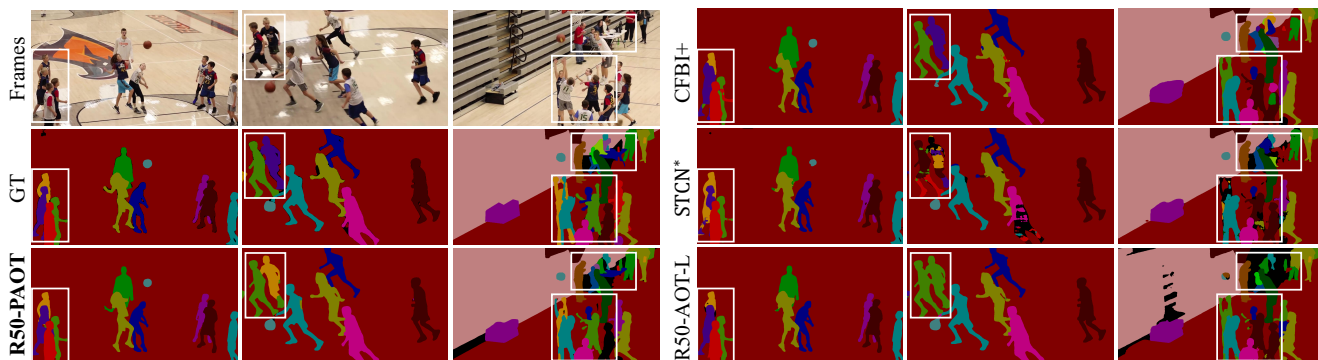


Figure 7: Qualitative results of different methods evaluated on VIPOSeg validation set. The scene is a basketball contest and includes multiple players moving fast and occluding each other. Difficult areas are marked with boxes.

Methods	Training	Y19	D17	D17-T	D16
CFBI+ [Yang <i>et al.</i> , 2021b]	Y+D	82.6	82.9	74.8	89.9
HMMN [Seong <i>et al.</i> , 2021]	Y+D	82.5	84.7	78.6	90.8
STCN [Cheng <i>et al.</i> , 2021c]	Y+D	82.7*	85.4	76.1	91.6
STCN* [Cheng <i>et al.</i> , 2021c]	Y+D	84.2*	85.3	79.9	91.7
RPCM [Xu <i>et al.</i> , 2022]	Y+D	83.9	83.7	79.2	91.5
R50-AOT-L [Yang <i>et al.</i> , 2021a]	Y+D	85.3*	84.9	79.6	91.1
SwinB-AOT-L [Yang <i>et al.</i> , 2021a]	Y+D	85.3*	85.4	81.2	92.0
R50-PAOT	Y+D	85.9*	85.3	81.0	92.2
R50-PAOT	Y+D+V	86.1*	86.0	82.1	92.5
SwinB-PAOT	Y+D	86.4*	86.2	84.0	93.8
SwinB-PAOT	Y+D+V	86.9*	87.0	83.6	93.3

Table 4: Evaluation results on YouTube-VOS 2019 validation (Y19), DAVIS 2016/2017 validation (D16/D17) and 2017 test (D17-T). * for Y19 denotes testing using all frames. *denotes models pre-trained with BL-30K.

6 Ablation Study and Discussion

Capacity of ID banks. The capacity of ID banks is a trade-off between efficiency and performance. The results are in Table 3. When the ID number increases, the performance drops while the speed rises and memory consumption decreases. Training more IDs results in less training data for each ID on average, and IDs with poorer generalization ability may affect the performance. For the classic setting, the best ID capacity is 10. For the panoptic setting, the best ID capacity is 10 for thing and 5 for stuff. For both R50-PAOT and SwinB-PAOT, panoptic ID strategy achieves better performance by decoupled ID banks with larger ID capacity.

Pyramid architecture. The pyramid architecture in PAOT is proposed to improve the original architecture of AOT. Here we compare two architectures and extend the LSTT in AOT-L from three to four layers (AOT-L4) for fair comparison. The results of SwinB backbone AOT and PAOT models on YouTube-VOS 2019 and VIPOSeg are in Table 5. Our pyramid architecture performs consistently better than AOT-L or AOT-L4 on different benchmarks.

Efficient LSTT. E-LSTT helps the PAOT models to better balance performance and efficiency. In Table 6, we compare the R50-PAOT models with and without E-LSTT. Models are

Models	Y19 VIPOSeg		Y19		VIPOSeg	
	Pyramid	\mathcal{G}^* \mathcal{G}	E-LSTT	\mathcal{G}^* \mathcal{G}	FPS	Mem./GB
SwinB-AOT-L	✗	85.3 73.3	✗	86.1 77.6	6.22	22.00
SwinB-AOT-L4	✗	85.4 74.2	✓	86.1 77.4	10.45	11.04
SwinB-PAOT	✓	86.5 75.3				

Table 5: Comparison between AOT substituting E-LSTT for original LSTT. \mathcal{G}^* denotes results of all-frame test. \mathcal{G} denotes results of all-frame test.

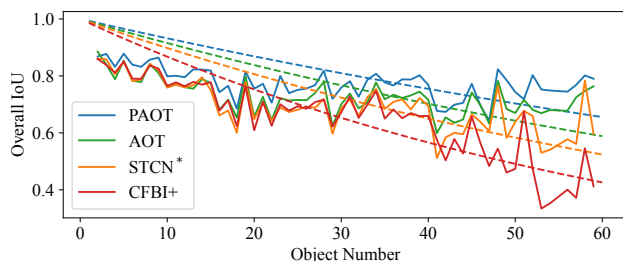


Figure 8: Crowd decay of different methods on VIPOSeg.

trained with Y+D+V and evaluated on YouTube-VOS 2019 and VIPOSeg. It can be found in the table that E-LSTT causes a little performance drop, but boosts the FPS from 6 to 10, and cuts down the memory consumption from 22 GB to 11 GB.

7 Conclusion

In this paper, we explore video object segmentation in panoptic scenes and present a benchmark dataset (VIPOSeg) for it. Our VIPOSeg dataset contains exhaustive annotations, and covers a variety of real-world object categories, which are carefully divided into thing/stuff and seen/unseen subsets. Training with VIPOSeg can boost the performance of VOS methods. In addition, the benchmark is capable of evaluating VOS models comprehensively. As a strong baseline method for panoptic VOS, PAOT tackles the challenges in VIPOSeg effectively by the pyramid architecture with efficient transformer and panoptic ID for panoptic object association. We hope our benchmark and baseline method can help the community for further research in related fields.

Acknowledgements

This work is supported by Major Program of National Natural Science Foundation of China (62293554) and the Fundamental Research Funds for the Central Universities (No. 226-2022-00051).

References

- [Caelles *et al.*, 2017] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 221–230, 2017.
- [Cheng *et al.*, 2014] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):569–582, 2014.
- [Cheng *et al.*, 2021a] Bowen Cheng, Ross Girshick, Piotr Dollár, Alexander C Berg, and Alexander Kirillov. Boundary iou: Improving object-centric image segmentation evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15334–15342, 2021.
- [Cheng *et al.*, 2021b] Ho Kei Cheng, Yu-Wing Tai, and Chi-Keung Tang. Modular interactive video object segmentation: Interaction-to-mask, propagation and difference-aware fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5559–5568, 2021.
- [Cheng *et al.*, 2021c] 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.
- [Cordts *et al.*, 2016] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.
- [Everingham *et al.*, 2010] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- [Hariharan *et al.*, 2011] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pages 991–998. IEEE, 2011.
- [He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [Kim *et al.*, 2020] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9859–9868, 2020.
- [Kim *et al.*, 2022] Dahun Kim, Jun Xie, Huiyu Wang, Siyuan Qiao, Qihang Yu, Hong-Seok Kim, Hartwig Adam, In So Kweon, and Liang-Chieh Chen. Tubeformer-deeplab: Video mask transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13914–13924, 2022.
- [Kristan *et al.*, 2023] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Hyung Jin Chang, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, et al. The tenth visual object tracking vot2022 challenge results. In *Computer Vision—ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VIII*, pages 431–460. Springer, 2023.
- [Li *et al.*, 2022] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Video k-net: A simple, strong, and unified baseline for video segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18847–18857, 2022.
- [Li *et al.*, 2023] Xiangtai Li, Henghui Ding, Wenwei Zhang, Haobo Yuan, Jiangmiao Pang, Guangliang Cheng, Kai Chen, Ziwei Liu, and Chen Change Loy. Transformer-based visual segmentation: A survey. *arXiv preprint arXiv:2304.09854*, 2023.
- [Lin *et al.*, 2014] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.
- [Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.
- [Meinhardt and Leal-Taixé, 2020] Tim Meinhardt and Laura Leal-Taixé. Make one-shot video object segmentation efficient again. *Advances in Neural Information Processing Systems*, 33:10607–10619, 2020.
- [Miao *et al.*, 2022] Jiaxu Miao, Xiaohan Wang, Yu Wu, Wei Li, Xu Zhang, Yunchao Wei, and Yi Yang. Large-scale video panoptic segmentation in the wild: A benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21033–21043, 2022.
- [Oh *et al.*, 2018] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. Fast video object segmentation by reference-guided mask propagation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7376–7385, 2018.

- [Oh *et al.*, 2019] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9226–9235, 2019.
- [Perazzi *et al.*, 2016] 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.
- [Pont-Tuset *et al.*, 2017] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.
- [Qi *et al.*, 2022] 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.
- [Seong *et al.*, 2020] Hongje Seong, Junhyuk Hyun, and Euntai Kim. Kernelized memory network for video object segmentation. In *European Conference on Computer Vision*, pages 629–645. Springer, 2020.
- [Seong *et al.*, 2021] Hongje Seong, Seoung Wug Oh, Joon-Young Lee, Seongwon Lee, Suhyeon Lee, and Euntai Kim. Hierarchical memory matching network for video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12889–12898, 2021.
- [Shi *et al.*, 2015] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE transactions on pattern analysis and machine intelligence*, 38(4):717–729, 2015.
- [Shin Yoon *et al.*, 2017] Jae Shin Yoon, Francois Rameau, Junsik Kim, Seokju Lee, Seunghak Shin, and In So Kweon. Pixel-level matching for video object segmentation using convolutional neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2167–2176, 2017.
- [Voigtlaender *et al.*, 2019] Paul Voigtlaender, Yuning Chai, Florian Schroff, Hartwig Adam, Bastian Leibe, and Liang-Chieh Chen. Feelvos: Fast end-to-end embedding learning for video object segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9481–9490, 2019.
- [Wang *et al.*, 2021a] Weiyao Wang, Matt Feiszli, Heng Wang, and Du Tran. Unidentified video objects: A benchmark for dense, open-world segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10776–10785, 2021.
- [Wang *et al.*, 2021b] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 568–578, 2021.
- [Weber *et al.*, 2021] Mark Weber, Jun Xie, Maxwell Collins, Yukun Zhu, Paul Voigtlaender, Hartwig Adam, Bradley Green, Andreas Geiger, Bastian Leibe, Daniel Cremers, et al. Step: Segmenting and tracking every pixel. *arXiv preprint arXiv:2102.11859*, 2021.
- [Woo *et al.*, 2021] Sanghyun Woo, Dahun Kim, Joon-Young Lee, and In So Kweon. Learning to associate every segment for video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2705–2714, 2021.
- [Xu *et al.*, 2018] 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.
- [Xu *et al.*, 2022] Xiaohao Xu, Jinglu Wang, Xiao Li, and Yan Lu. Reliable propagation-correction modulation for video object segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2946–2954, 2022.
- [Yang and Yang, 2022] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. *Advances in Neural Information Processing Systems*, 34, 2022.
- [Yang *et al.*, 2018] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. Efficient video object segmentation via network modulation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6499–6507, 2018.
- [Yang *et al.*, 2020] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by foreground-background integration. In *European Conference on Computer Vision*, pages 332–348. Springer, 2020.
- [Yang *et al.*, 2021a] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. *Advances in Neural Information Processing Systems*, 34, 2021.
- [Yang *et al.*, 2021b] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by multi-scale foreground-background integration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [Yang *et al.*, 2021c] Zongxin Yang, Jian Zhang, Wenhao Wang, Wenhua Han, Yue Yu, Yingying Li, Jian Wang, Yunchao Wei, Yifan Sun, and Yi Yang. Towards multi-object association from foreground-background integration. In *CVPR Workshops*, volume 2, 2021.
- [Zhou *et al.*, 2022] Tianfei Zhou, Fatih Porikli, David J Crandall, Luc Van Gool, and Wenguan Wang. A survey on deep learning technique for video segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.