Large-scale Video Panoptic Segmentation in the Wild: A Benchmark

Jiaxu Miao[1,2] Xiaohan Wang[1] Yu Wu[2] Wei Li[1] Xu Zhang[1] Yunchao Wei[3] Yi Yang[1†]
[1]CCAI, Zhejiang University [2] Baidu Research [3] Beijing Jiaotong University
jiaxumiao@zju.edu.cn, yangyics@zju.edu.cn

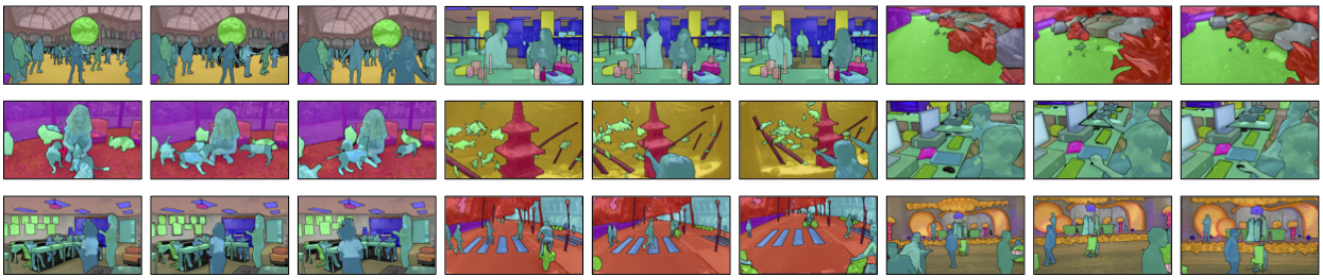

Figure 1. Examples of our large-scale VIdeo Panoptic Segmentation in the Wild (VIPSeg) dataset.

Abstract

*In this paper, we present a new large-scale dataset for the video panoptic segmentation task, which aims to assign semantic classes and track identities to all pixels in a video. As the ground truth for this task is difficult to annotate, previous datasets for video panoptic segmentation are limited by either small scales or the number of scenes. In contrast, our large-scale **VI**deo **P**anoptic **Seg**mentation in the Wild (VIPSeg) dataset provides 3,536 videos and 84,750 frames with pixel-level panoptic annotations, covering a wide range of real-world scenarios and categories. To the best of our knowledge, our VIPSeg is the first attempt to tackle the challenging video panoptic segmentation task in the wild by considering diverse scenarios. Based on VIPSeg, we evaluate existing video panoptic segmentation approaches and propose an efficient and effective clip-based baseline method to analyze our VIPSeg dataset. Our dataset is available at https://github.com/VIPSeg-Dataset/VIPSeg-Dataset/.*

1. Introduction

Panoptic segmentation unifies semantic and instance segmentation tasks by assigning a semantic label and an instance ID to every pixel in an image, which is a fundamental research topic in computer vision and has many practical applications such as detailed action understanding, video editing, autonomous driving, and augmented reality. Recently, plenty of approaches [15, 16, 26, 27, 29, 31, 34, 51, 63, 68, 74] have been proposed for panoptic segmentation and achieved remarkable progress.

Although the image panoptic segmentation task has been well explored, video panoptic segmentation [26] (VPS) is still a challenging problem. VPS model should not only provide unique and consistent semantic predictions within a video, but also associate instance IDs for the same object across frames. Recently some approaches and datasets [26, 48, 60, 61] have been proposed for video panoptic segmentation. However, there are many limitations in current VPS benchmarks. First, existing VPS datasets [26, 60] are small-scaled due to the exhausting labeling cost. For example, Cityscapes-VPS [26] only contains 500 videos with six annotated frames per video. KITTI-STEP [60] and MOTChallenge-STEP [60] only contain 50 and 4 video sequences, respectively. The video panoptic segmentation task is constrained by existing datasets due to their insufficient videos [60] and short video length [26]. Second, the diversity of existing VPS datasets is restricted, *i.e.*, only the street view scene is considered in previous datasets. Thus, the categories of things with pixel-level annotation are limited and biased. Some previous datasets [60] only focus on people and vehicles. The diversity issue prevents these datasets from being general in real-world applications (*e.g.* video editing, augmented reality), which contains many scenes and hundreds of things in our daily life.

To advance the research on video panoptic segmentation, we present a new dataset in this work, targeting large-scale VIdeo Panoptic Segmentation in the wild (VIPSeg).

†Corresponding author.
‡Part of this work was done when Jiaxu Miao was an intern at Baidu Research.

The dataset contains a wide range of real-world scenarios (*e.g.*, 232 scenes) and categories (*e.g.* 124 classes). Totally we annotated 3,536 videos and 84,750 frames with pixel-level panoptic annotations, including both semantic categories for background stuff (*e.g.*, sky,ground) and track identities for foreground things (*e.g.*, person, cats, cars). To the best of our knowledge, our VIPSeg is the first attempt to tackle the challenging video panoptic segmentation task in the wild by considering diverse scenarios. Since semantic IDs and instance IDs of all pixels are annotated, our VIPSeg can also applied to other video tasks including Video Object Segmentation, Video Semantic Segmentation,Video Instance Segmentation, *etc*..

Annotating such a large-scale video panoptic segmentation is difficult and expensive, since semantic classes and tracking ids of every pixel are required. To overcome exhausting human efforts, we propose a Sparse-to-Dense Interactive Annotation strategy to efficiently annotate panoptic masks by the collaboration of humans and computers. Concretely, we first propose to annotate instances for each frame at a sparse frame rate (1 fps) and associate instances using a tracking model [71] and manual correction. After that, we adopt a video object segmentation model AOT [71] to extend the frame rate from 1 fps to 5 fps and manually refine instance masks to improve segmentation quality.

We conduct extensive experiments on VIPSeg to evaluate existing video panoptic segmentation models. Most of existing works [26, 48] on VPS inference predictions iteratively, where they generate the next frame prediction by taking the previous results as the reference. However, real-world videos would last long, and the iterative inference would be less efficient in applications. Thus, we propose an clip-based model extended from PanopticFCN [34] to divide a video into non-overlapping clips and individually generate predictions for each clip. The clip-based method could process video panoptic segmentation in parallel and be more efficient in real applications. We adopt the clip-based model to evaluate and analyze our VIPSeg dataset.

2. Related Work

Panoptic Segmentation. Panoptic segmentation [27] is a comprehensive computer vision task that combines the semantic segmentation and instance segmentation tasks. Recently, the panoptic segmentation task has become more and more popular, and many methods [15, 16, 26, 27, 29, 31, 34, 51, 63, 68, 74] have been proposed to address this unified task. A simple baseline introduced in [27] is to train two sub-tasks separately and fuse the results heuristically. After that, some methods present an end-to-end model but still utilize two branches to tackle the panoptic segmentation task separately. For example, Xiong *et al*. [63] propose UPSNet which leverages a two-stage detection module for instance segmentation and a pixel-wise classification module for semantic segmentation. Cheng *et al*. [15] design a two-branch pipeline by predicting instance centers and pixel offsets for instance segmentation and pixel-wise classification for semantic segmentation. Differently, Li *et al*. [32] suggest to represent and predict things and stuff in a unified fully convolutional pipeline. However, this method still treats things and stuff with different strategies. Most recently, transformer-based methods [16, 74] consider things and stuff uniformly by initialized queries or kernels.

Video Semantic Segmentation. Compared to image semantic segmentation [9–11, 14, 21, 25, 30, 35, 46, 52, 57, 58, 62, 66, 72, 73, 75, 76], video semantic segmentation (VSS) requires assigning a class label to every pixel in all frames of a video sequence. Early works for video semantic segmentation only leverage adjacent RGB frames without annotations to improve the segmentation accuracy [18, 24, 38, 39, 44, 45] or accelerate inference speed by feature reusing [6, 20, 23, 33, 40, 49, 69, 79]. Since early datasets for video semantic segmentation [3, 17, 50] are limited by small scales and sparse annotations, temporal evaluation for VSS is not conducted. Recently, a large-scale dataset [43] with dense temporal annotation is introduced, which provides a suitable benchmark for the VSS task. A temporal context fusion method [43] is proposed to promote both the segmentation quality and temporal consistency. Video semantic segmentation is different from our setting because it does not require discriminating different instances and instance tracking.

Video Object Segmentation. Video Ojbect Segmentation (VOS) [4, 13, 41, 42, 44, 47, 53–56, 64, 70, 77, 78] aims to segment objects in a video sequence given only the object masks on the first frame, which is class-agnostic. VOS approaches can be roughly divided into two types. Finetuning-based methods [4, 42] train a network for foreground-background segmentation, and fine-tune the model using first-frame ground truth when testing. Propagation-based methods [53, 70, 71] take results of previous frames as input to generate the current frame mask. Our dataset can also be applied to the VOS task.

Video Instance Segmentation. Video Instance Segmentation (VIS) [2, 7, 8, 12, 19, 32, 65] combines instance segmentation and video object tracking, which aims to segment and track instance masks across video frames. Early works [5, 65] tackle the two sub-tasks separately, leveraging frame-by-frame instance segmentation and an additional tracking head to solve the VIS problem. Recently proposed methods [37, 67] consider temporal information to improve the segmentation and tracking performance. Clip-based methods [1, 22, 59] propose to leverage a clip of frames simultaneously for higher segmentation and tracking accuracy. For instance, IFC [22] proposes memory tokens to exchange information across frames efficiently and improves segmentation performance. Our dataset can also

be applied to the VIS task.

Video Panoptic Segmentation. Kim *et al*. [26] first introduce the video panoptic segmentation (VPS) task, which aims to simultaneously predict object classes, bounding boxes, masks, instance id associations, and semantic segmentation in video frames. VPSNet [26] is the first work for VPS, which is based on UPSNet [63]. Pixel-level fusion and object-level tracking are added to adjust the image panoptic segmentation method UPSNet to the VPS task. Woo *et al*. [61] further extend VPSNet by learning the temporal correspondence across frame pairs. VIP-Deeplab [48] extends Panoptic-Deeplab [15] using center offset regression from two frame pixels to one frame centers. Most of these methods [26, 48] inference video panoptic segmentation results iteratively. In this paper, we propose a clip-based method to improve temporal stability and efficiency.

3. VIPSeg: A Large-scale VIdeo Panoptic Segmentation Dataset

In this section, we detailly introduce our dataset, VIPSeg, and compare our dataset with existing VPS datasets and analyze VIPSeg using its statistics information. In addition, we describe the annotation pipeline of our VIPSeg dataset.

3.1. Dataset Summary

There are a total of 3,536 videos with 84,750 pixel-wise annotated frames in VIPSeg. Each video lasts from 3 seconds to 10 seconds. We sample frames with a frame rate of 5 fps. Different from existing VPS datasets [26, 60] that only focus on the street view scene, VIPSeg covers 232 scenarios with 124 categories, including 58 things' classes and 66 stuff's classes, making our dataset more challenging and practical. We decided a category as thing or stuff considering if it is easy to split into individual instances. The instance IDs of identical objects in a video are carefully associated across frames. For example, as shown in Fig. 3, for every moving dog, we provide the segmentation masks and associated IDs. We totally annotate 926,213 instance masks in VIPSeg.

3.2. Comparison with Existing Datasets

The comparisons between our dataset and existing related datasets are shown in Table 1. We mainly compare our VIPSeg dataset with existing real-world video panoptic segmentation datasets, Cityscapes-VPS [26], KITTI-STEP [60] and MOTChallenge-STEP [60]. A synthetic dataset (VIPER) [26] from GTA-5 for the VPS task will not be discussed in this paper. Compared with existing VPS datasets [26,60], our VIPSeg contains more than 3,000 videos, which is about six times larger than Cityscapes-VPS and about 60 times larger than KITTI-STEP and MOTChallenge-STEP.

Moreover, our dataset is comprised of much more diverse scenes, including 232 indoor and outdoor scenes, while previous datasets only focus on the street view scene. Our dataset contains 124 categories with 58 thing classes and 66 stuff classes, which is about six times larger than Cityscapes-VPS and KITTI-STEP, and 18 times larger than MOTChallenge-STEP, making our VIPSeg more challenging and practical for real-world applications. Since KITTI-STEP and MOTChallenge-STEP are extended from tracking datasets, only "person" and "vehicles" are annotated with tracking IDs. Differently, VIPSeg has more diverse thing classes including "person", "cars", "cats", "horses" and so on. Besides, the average sequence length of our VIPSeg is 24, which is much larger than Cityscapes-VPS (6 frames per video). A longer track length of instances will introduce more occlusions and appearance change, which is more complex and challenging.

3.3. Dataset Statistics

We organize the categories with a two-level hierarchical taxonomy. Fig. 2 shows the histogram of instance masks for parent classes and their sub-classes. There are a total of 25 parent classes and 124 sub-classes. In each parent class, the distribution of the sub-class frequencies is long-tailed, which is typically found when a dataset is naturally collected without manual balancing. For thing-classes, "person", "chair or seat" and "car" contain the most object masks, which are common objects in the real world. For stuff-classes, "tree" and "sky" have the most object masks.

Fig. 4 (a) demonstrates the distribution of ranked object frequencies for different scenes. The object frequencies also show a long-tail distribution. Scenes with "person","chair or seats" usually contain more instances, such as "computer room" or "crosswalk". In contrast, natural sceneries such as "grotto" or "forest broadleaf" contain fewer instances. Fig. 4 (b) shows the distribution of instance numbers for the tracking length. Most instances exist across 15 frames (3 seconds).

The distributions of mean object area of different thing-classes and stuff-classes are shown in Fig. 4(c)(d). The object area of stuff is much larger than things on average. Both two distributions are long-tailed, and the discrepancy of object area for things is larger than stuff, indicating that it is more challenging to recognize small thing objects.

3.4. Annotation Pipeline

We extend a video semantic segmentation dataset, VSPW [43], to our video panoptic segmentation dataset. Although semantic labels are provided, annotating such a large video panoptic segmentation dataset is still time-consuming and expensive. It is a burdensome project to annotate and associate 926,213 instances for all frames from 58 categories. The major difficulty is how to associate in-

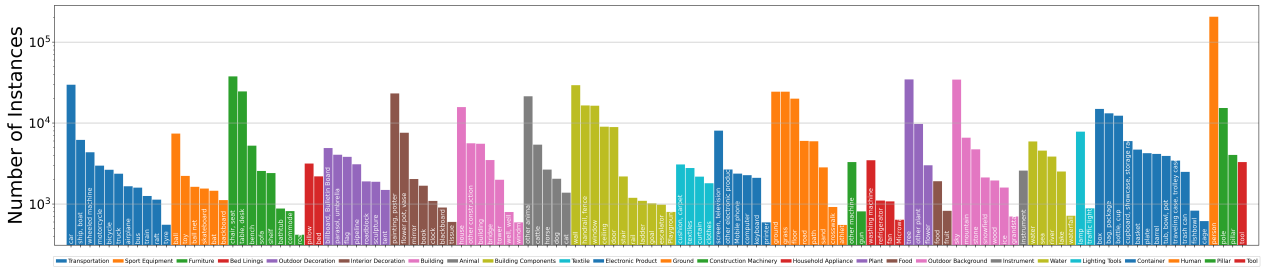

Figure 2. The histogram of the ranked instance masks for parent classes and sub-classes.

Table 1. Comparison of video panoptic segmentation datasets.

Dataset	#Scene	#Videos	#Frames	#Thing Classes	#Stuff Classes	#Annotated Masks	#Frames per Video
Cityscapes-VPS	1	500	3,000	8	11	72,171	6
KITTI-STEP	1	50	18,181	2	17	126,529	381
MOTChallenge-STEP	1	4	2,075	1	6	17,232	562
VIPSeg	232	3,536	84,750	58	66	926,213	24

Figure 3. Example of associated instance annotations.

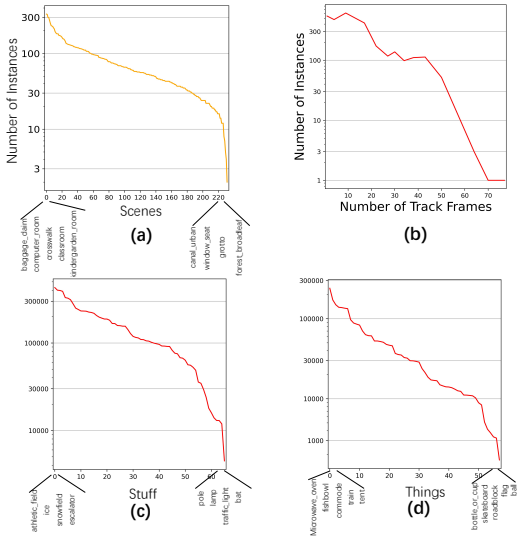

Figure 4. **(a)** The distribution of ranked object frequencies for different scenes. **(b)** The histogram of instance masks for parent classes and sub-classes.

stances across frames correctly with a dense frame rate. To save time and human labor, we design a Sparse-to-Dense Interactive Annotation pipeline, which provides an efficient way to annotate and associate instances across frames. Concretely, we first adopt a Sparse Annotation and Tracking Loop for instance labeling and tracking with a sparse frame rate of 1 fps. Then we use a Dense Pixel-label Propagation Loop to propagate annotated instance masks from 1 fps to 5 fps and manually refine annotations to improve quality. To guarantee the annotation quality of the interactive annotation pipeline, we employed four expert annotators to double-check the errors generated by machines. The annotation pipeline is shown in Fig. 5.

3.4.1 Sparse Annotation and Tracking Loop

Annotating a video with a high frame rate is often time-consuming and wastes human labor. Thus, we first ask human annotators to sparsely annotate instance-level segmentation masks with a frame rate of 1 fps. It is difficult to keep instance IDs consistent when annotating instance masks. Thus, we divide the procedure into two steps. First, annotators only need to label instance masks for each frame and overlook video-level instance consistency. It took about 1,200 hours for instance annotation and human review. Second, we adopt a multiple-object tracking model [71] to associate the annotated instances. Instance masks with their IDs are propagated from the first frame. We compute Intersection over Union (IoU) between instances of two frames and use Hungarian algorithm [28] to assign instance IDs to the next frame. Some hard cases, such as occlusion or motion blur, usually lead to tracking failure. Thus, human annotators correct the mis-associated instances in one frame, and the corrected instances are as inputs of the tracker to further improve the tracking results. This loop is conducted

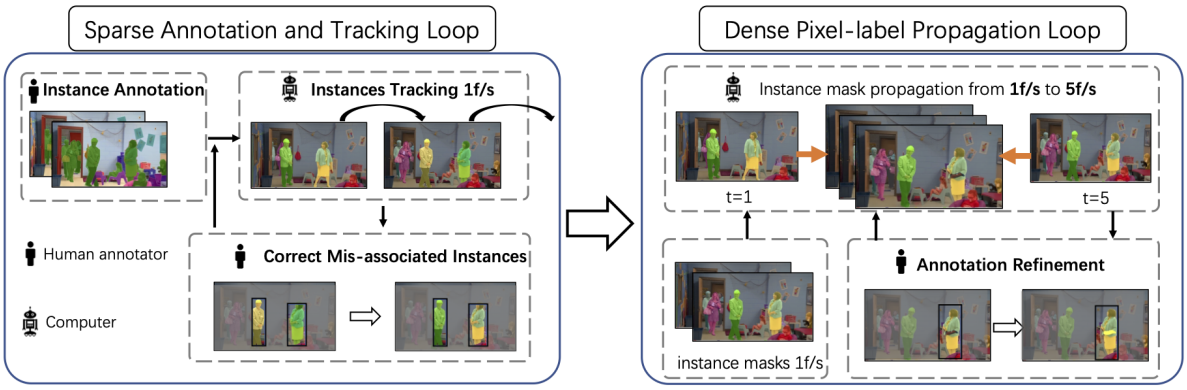

Figure 5. The "Sparse-to-Dense Interactive Annotation" pipeline, including two phases. **(a) Sparse Annotation and Tracking Loop.** First we annotate instances by 1 fps manually. The annotated instances in a video are associated by a tracking model. **(b) Dense Pixel-label Propagation Loop.** We use a video object segmentation model to propagate the annotated masks from 1fps to 5fps. The generated masks are further checked and refined by human annotators.

until all instances are associated. Instances association by computers costed about 20 hours.

3.4.2 Dense Pixel-label Propagation Loop

In this loop, we extend the associated instance masks and IDs from 1 fps to 5 fps. We adopt a state-of-the-art video object segmentation method, *i.e.*, AOT [71], to propagate the instance masks and ids from the annotated frames to their adjacent unlabelled frames and generate masks at 5 fps. Instance propagation by computers costed about 20 hours. Owing to the scene complexity of our VIPSeg, there exist defects in some propagated masks. After generating instance masks of unlabelled frames, annotators are asked to check the segmentation quality and refine the instance masks artificially. The propagation method usually failed when meeting complex videos with many instances in a scene (more than 20). In VIPSeg, around 28% of videos are complex and exist failure cases, which need further refinement. The time of the refinement for human annotators depends on the complexity of each video. Videos with less than ten instances required ten minutes or less. Videos with 10-30 instances required 20-30 minutes and complex videos with more than 30 instances usually took 40-60 minutes. The model propagation and artificial refinement are operated repeatedly until the results are satisfactory.

4. Method

Existing works [26, 48] on VPS are usually built on iterative frameworks, which take an adjacent frame as the reference to correlate temporal information within videos. To preserve the unique tracking ids within a video, they have to generate the next frame prediction based on previous results. However, real-world videos would last long, and the iterative inference would be less efficient in applications. It motivates us to develop a VPS baseline to study and analyse our dataset in a non-iterative way.

In this paper, we propose a clip-based VPS model, ClipPanoFCN, extended from an image-based method PanopticFCN [34]. For long video input, we divide it into several *non-overlapping* clips and thus individually predict the panoptic results, including tracking ids for thing objects. After that, we perform clip-level association and tracking to make predictions unique and consistent within the whole long video sequence. It contains two stages, *i.e.*, frame-level modelling and clip-level aggregation, as shown in Fig. 6.

4.1. Frame-level Modelling

For an input video $V = (I_1, I_2, ..., I_T)$, I_i is the i-th frame with a spatial size of $H \times W$ and the total frame number in the video is T. We first process each video frame I_i to obtain frame-level kernels and a high-resolution feature map $\mathbf{F}_i$ based on the image-level method PanopticFCN [34]. The generated kernels represent things or stuff in the frame, while the high-resolution feature map maintains spatial information of this frame. This module mainly contains three components, *i.e.*, FPN (Feature Pyramid Network) [36] backbone, kernel generator, feature encoder. Following PanopticFCN, for each frame I_i, we first extract the features using FPN. Then we use convolutional layers to predict centers for each individual object and regions for each stuff category, and another convolutional layers to generate kernel weights for each thing and stuff prediction. Thus, we can obtain kernels by selecting the kernel weight at the location of centers of things or by averaging the kernel weights in the categories' regions.

The feature encoder is comprised of three convolutional layers and takes features from the FPN backbone to generate the high-resolution feature map $\mathbf{F}_i \in \mathbb{R}^{C \times \frac{1}{4}H \times \frac{1}{4}W}$ for

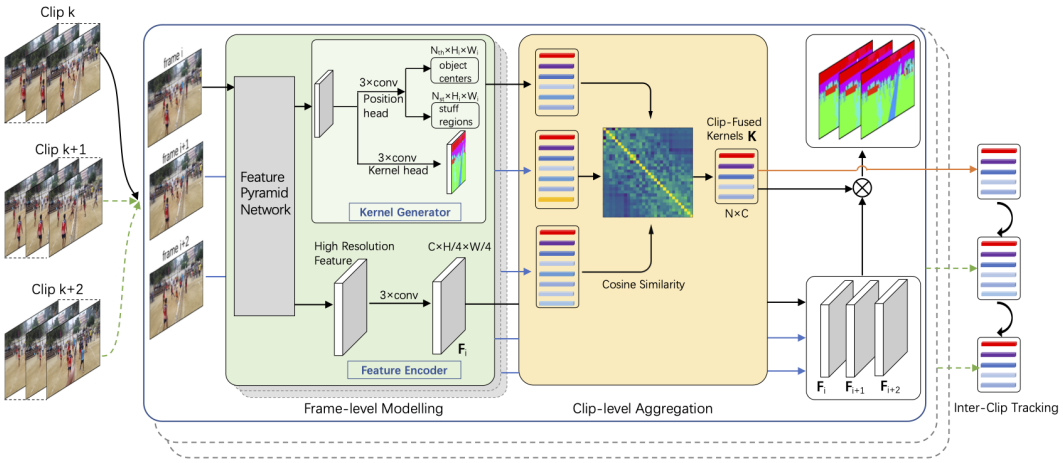

Figure 6. Our clip-based VPS model includes two stages, frame-level modelling and clip-level aggregation. Frame-level modelling is extended from PanopticFCN [34] to generate kernels for things and stuff per frame. The Clip-level aggregation module fuses kernels to generate an individual kernel for every instance in the clip.

frame I_i. Given a kernel of things or stuff, using $\mathbf{F}_i$ we can generate the corresponding object mask by convolutional operation.

4.2. Clip-level Aggregation

To efficiently adopt the image-level panoptic segmentation model to the video level, we propose the Clip-level Aggregation (CA) in a non-iterative way. As shown in Fig. 6, After generating kernels for each frame, we can aggregate these frame-level kernels within a clip size of c. Then we use the convolutional operation to generate instance masks with feature maps and clip-fused kernels. Finally, we associate identity tracks among clips to make unique and consistent predictions for the whole video.

Clip-fused Kernel. Kernel fusion is designed to remove duplicate predictions and merge kernel weights with the same identity. We use the average-clustering operation to aggregate kernel weights which have the same prediction identities. The intuition is that pixels belonging to the same things/stuff would not have dramatic appearance change within a short clip. For things, we calculate the cosine similarity of all generated kernels per category within the clip, and then merge them if their similarity is higher than a pre-defined threshold. For stuff, we simply calculate the average pooling of kernels from all frames in the clip. In this way, the fused kernel can be treated as an embedding for an individual object for thing classes, or a semantic category for those stuff classes in the clip. Assume the total number of things and stuff in the clip is N, we obtain clip-fused kernels $\mathbf{K} \in \mathbb{R}^{N \times C}$.

Panoptic Mask Prediction. The generated things and stuff kernels $\mathbf{K}$ in the clip are applied on the high-resolution features $\{ \mathbf{F}_i, \mathbf{F}_{i+1}, ..., \mathbf{F}_{i+c} \}$ by the convolutional operation to generate object masks in each frame. Since one instance has an individual kernel to generate the instance masks in the clip, the intra-clip association is guaranteed.

Inter-clip Tracking. Based on the clip-level results, we then associate and merge the tracking IDs of things among clips. Since content may change a lot in long videos, instead of kernel fusion, we fuse and merge clip-level predictions in a post-process way. To do so, we calculate the similarity between every two adjacent clip-fused kernels. We merge predictions if their clip-fused kernels are similar (similarity higher than a threshold). Otherwise, we will assign those dissimilar tracks new identities.

The association is performed clip by clip to generate video-level panoptic predictions. In this way, we are able to inference each clip in parallel and then post-process clip predictions with a very lightweight cost on the master node.

5. Experiments

5.1. Dataset Splits

The train set, validation set and test set of VIPSeg contain $2,806/343/387$ videos with $66,767/8,255/9,728$ frames, respectively. Considering the limitation of the computation source, we resize all the frames in VIPSeg into 720P (the size of the short side is resized to 720) for training and testing.

5.2. Evaluation Metrics

There are two commonly used evaluation metrics for video panoptic segmentation, VPQ [26] and STQ [60].

Video Panoptic Quality (VPQ) [26] for video panoptic segmentation is based on PQ (Panoptic Quality) [27] and computes the average quality by using tube IoU matching across a small span of frames. Formally, the VPQ score

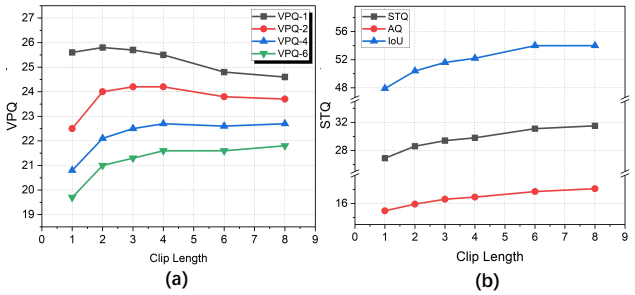

Figure 7. Impact of the clip length on VPQs and STQ.

across k frames is

$$\text{VPQ}^k = \frac{1}{\text{N}_{classes}} \sum_c \frac{\sum_{p,g \in \text{TP}_c} \text{IoU}(p,g)}{|\text{TP}|_c^k + \frac{1}{2}|\text{FP}|_c^k + \frac{1}{2}|\text{FN}|_c^k}, \quad (1)$$

where True Positive (TP) matches is defined as $\text{TP} = (p,g) : \text{IoU}(p,g) > 0.5$ while False Positives (FP) and False Negatives (FN) are defined accordingly. When $k = 1$, the VPQ metric is equivalent to the image PQ metric. Since VPQ is used to evaluate the sparsely annotated Cityscapes-VPS dataset with six frames per video, the spans of VPQ are mostly set to $k = 4$. When using more than four frames, the difficulty of the 3D IoU matching increases significantly. However, the video length of our VIPSeg is much longer. Thus, in this paper, we use VPQ with the spans at $k = 1, 2, 4, 6$. Longer spans will cost much time (It will cost about an hour when $k = 8$).

Segmentation and Tracking Quality (STQ) [60] is proposed to measure the segmentation quality and long tracking quality simultaneously. STQ is comprised of two factors, Association Quality (AQ) and Segmentation Quality (SQ). AQ is designed to measure pixel-level association across an entire video. SQ is used to measure semantic segmentation quality by class-level IoU. STQ is a balance of AQ and SQ by $\text{STQ} = (\text{AQ} \times \text{SQ})^{\frac{1}{2}}$. STQ can measure the association and semantic segmentation quality of predictions in the entire video. However, STQ evaluates segmentation quality using semantic segmentation and ignores IoU per instance.

5.3. Segmentation Results and Analysis

We evaluate existing VPS methods on our VIPSeg dataset, including VPSNet-FuseTrack [26], VPSNet-SiamTrack [61], and VIP-DeepLab [48]. Moreover, we evaluate and analyze VIPSeg using our Clip-PanoFCN.

5.3.1 Ablation Study and Analysis

We conduct ablation studies and result analysis based on our Clip-PanoFCN baseline.

Impact of Clip Length. Clip-based method for video panoptic segmentation can inference in parallel for efficiency and the clip length is an important hyper-parameter. Fig. 7(a) shows how the clip length C impact VPQs and STQ. We set C from 1 to 8. $C = 1$ means we predict panoptic masks by single frame and use tracking by kernels. For VPQ^1 score, which means only single frame segmentation quality is considered, with the clip length C grows, the performance decreases correspondingly because the clip kernel fusion module with long clip length introduces more noises and affects the single frame segmentation quality.

When $k > 1$, VPQ^k scores indicate the panoptic segmentation quality and instance association quality. With C growing from 1 to 3, VPQ increases accordingly while VPQ slightly decreases when C becomes larger, demonstrating that there exists a trade-off between the segmentation quality and association quality when using the clip-based model. A longer clip brings more stable association results but introduces more noises that harm the segmentation quality.

STQ score indicates the semantic segmentation and pixel-level association quality of entire videos. Fig. 7(b) shows how the clip length affect STQ. A natural conclusion is that for a longer clip the STQ score is higher. This is because STQ focuses on the quality of entire videos and longer clip brings better association results when the video length is large.

Impact of the Tracking Strategies. Table 2 shows that how tracking strategies affect the segmentation performances. PanopticFCN [34] is our image-based baseline. "+Track" denotes we use the kernel tracking strategy to associate instances. "+Clip" denotes using the clip-based model for object association. "+Clip and Track" denotes we use the clip-based model for intra-clip instance association and the kernel tracking for inter-clip instance association. Results show that both the clip-based model and kernel tracking improve $\text{VPQ}^2 - \text{VPQ}^6$. VPQ^1 is not improved because this metric equals to image PQ metric that is not sensitive to video-level instance association.

Table 2. Ablation on the Tracking Strategies.

Method	VPQ^1	VPQ^2	VPQ^4	VPQ^6	VPQ
PanopticFCN [34]	25.7	21.2	19.6	18.5	21.2
+Track	25.6	22.6	20.4	19.7	22.0
+Clip	25.7	23.8	21.6	20.0	22.7
+Clip and Track	25.7	24.2	22.5	21.2	23.4

Result Analysis of VIPSeg. Fig. 8 (a) presents how the number of instances per video affects the segmentation and association performance. The number of instances per video is negatively correlated to the STQ score of the video, indicating the videos with more instances perform lower STQ score. This is reasonable since more instances mean more complicated scenes and introduce more occlu-

Table 3. Comparison on the validation set and the test set.

(a) Results on the validation set.

Method	Backbone	VPQ^1	VPQ^2	VPQ^4	VPQ^6	VPQ	STQ
VIP-DeepLab [48]	ResNet-50	18.4|15.6|20.9	16.9|13.9|19.9	14.8|10.8|18.9	13.7|9.2|18.2	16.0|12.3|18.2	22.0
VPSNet-FuseTrack [26]	ResNet-50	19.9|20.9|19.2	18.1|18.5|17.8	15.8|15.2|16.4	14.5|13.6|15.5	17.0|17.0|17.2	20.8
VPSNet-SiamTrack [61]	ResNet-50	20.0|20.9|19.3	18.3|18.8|17.9	16.0|15.5|16.5	14.7|14.0|15.5	17.2|17.3|17.3	21.1
Clip-PanoFCN	ResNet-50	24.3|27.1|21.5	23.5|25.8|21.2	22.4|24.2|20.6	21.6|23.2|20.0	**22.9**|**25.0**|**20.8**	**31.5**

(b) Results on the test set.

Method	Backbone	VPQ^1	VPQ^2	VPQ^4	VPQ^6	VPQ	STQ
VIP-DeepLab [48]	ResNet-50	16.9|15.3|18.4	15.0|11.8|18.1	13.6|9.7|17.5	12.5|8.2|16.9	14.5|11.3|17.7	20.2
VPSNet-FuseTrack [26]	ResNet-50	18.2|18.4|18.0	17.0|16.5|17.5	14.8|13.2|16.2	13.6|11.7|15.5	15.9|15.0|16.8	19.0
VPSNet-SiamTrack [61]	ResNet-50	18.5|18.6|18.4	17.2|16.7|17.7	15.1|13.2|17.0	14.0|12.0|16.0	16.2|15.1|17.2	19.1
Clip-PanoFCN	ResNet-50	23.8|25.9|21.9	22.8|24.3|21.5	21.5|22.1|21.0	20.3|20.1|20.5	**22.0**|**23.1**|**21.2**	**28.7**

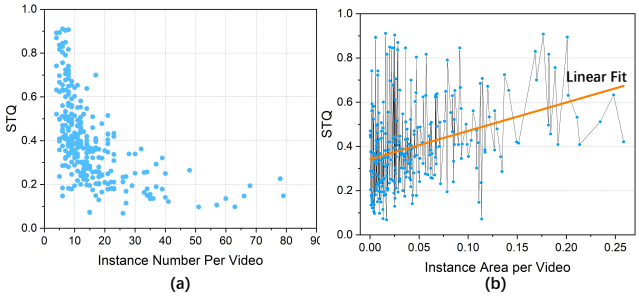

Figure 8. **(a)** Impact of the number of instance per video on STQ. **(b)** Impact of the average instance area on STQ.

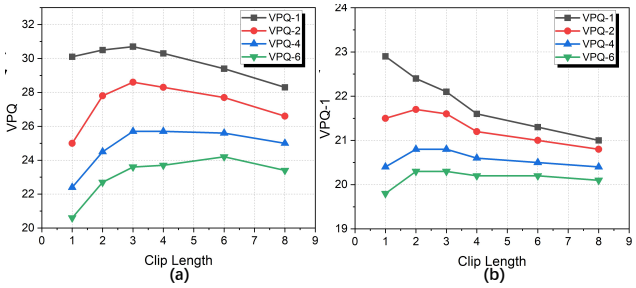

Figure 9. **(a)** Impact of the clip length on the segmentation quality of thing-classes. **(b)** Impact of the clip length on the segmentation quality of stuff-classes.

sions, making it harder to segment and track objects.

Fig. 8 (b) shows how the average area of instances per video affects the STQ score. Although there exists a tendency that the video with smaller instances performs lower STQ, the area of instances is not a critical factor to affect the segmentation and tracking quality.

Fig. 9 illustrates the comparison between results on thing-classes and stuff-classes for different clip lengths. For thing-classes, VPQ^k (k>1) is heavily affected by the clip length. In contrast, for stuff-classes VPQ^k is similar with the clip length growing, indicating that clip kernel fusion only takes effects on the instances association but cannot improve stuff segmentation quality.

5.3.2 Results Comparison

We report quantitative results on the baselines, including VPSNet-FuseTrack [26], VPSNet-SiamTrack [61], VIP-DeepLab [48] and our Clip-PanoFCN. We use the clip length of 8 here. Table 3 shows the results of the baselines on VPQ and STQ. Clip-PanoFCN outperforms VPSNet [26, 61] and VIP-DeepLab [48] on both VPQ and STQ. However, since the base models are different, the results cannot show the superiority of Clip-PanoFCN. The main advantage of Clip-PanoFCN is the parallel processing.

6. Limitations

The proposed dataset contains various categories in the real world. Although our VIPSeg is much larger than existing VPS datasets, the number of training videos is still insufficient to support the precise panoptic segmentation on such various classes and scenes.

Besides, the distribution of instances for categories is long-tailed in VIPSeg. The proposed Clip-PanoFCN is not robust to the few-shot categories while there exist a large number of tail classes in the proposed dataset and real-world applications.

7. Conclusion

In this paper, we introduce a large-scale dataset for video panoptic segmentation, VIPSeg. Different from the existing datasets that are either small-scales or with a limited number of scenes, our large-scale Video Panoptic Segmentation in the Wild (VIPSeg) dataset provides large-scale pixel-level panoptic annotations, covering a wide range of real-world scenarios and categories. Besides, we evaluate the existing video panoptic segmentation approaches and further propose an effective clip-based baseline method. Elaborate analysis and experiments demonstrate the significance of the proposed video panoptic segmentation benchmark.

References

[1] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9739–9748, 2020. 2

[2] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: real-time instance segmentation. In *IEEE ICCV*, pages 9157–9166, 2019. 2

[3] Gabriel J. Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, pages 44–57, 2008. 2

[4] Sergi Caelles, Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. One-shot video object segmentation. In *IEEE CVPR*, pages 221–230, 2017. 2

[5] Jiale Cao, Rao Muhammad Anwer, Hisham Cholakkal, Fahad Shahbaz Khan, Yanwei Pang, and Ling Shao. Sipmask: Spatial information preservation for fast image and video instance segmentation. In *ECCV*, pages 1–18. Springer, 2020. 2

[6] Joao Carreira, Viorica Patraucean, Laurent Mazare, Andrew Zisserman, and Simon Osindero. Massively parallel video networks. In *ECCV*, pages 649–666, 2018. 2

[7] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang, and Youliang Yan. Blendmask: Top-down meets bottom-up for instance segmentation. In *IEEE ICCV*, 2020. 2

[8] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *IEEE CVPR*, pages 4013–4022, 2018. 2

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE TPAMI*, 40(4):834–848, 2017. 2

[10] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 2

[11] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018. 2

[12] Xinlei Chen, Ross Girshick, Kaiming He, and Piotr Dollár. Tensormask: A foundation for dense object segmentation. In *IEEE ICCV*, pages 2061–2069, 2019. 2

[13] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *IEEE CVPR*, pages 1189–1198, 2018. 2

[14] Bowen Cheng, Liang-Chieh Chen, Yunchao Wei, Yukun Zhu, Zilong Huang, Jinjun Xiong, Thomas S Huang, Wen-Mei Hwu, and Honghui Shi. Spgnet: Semantic prediction guidance for scene parsing. In *IEEE ICCV*, pages 5218–5228, 2019. 2

[15] Bowen Cheng, Maxwell D Collins, Yukun Zhu, Ting Liu, Thomas S Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation. In *IEEE CVPR*, 2020. 1, 2, 3

[16] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021. 1, 2

[17] 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 *IEEE CVPR*, pages 3213–3223, 2016. 2

[18] Raghudeep Gadde, Varun Jampani, and Peter V Gehler. Semantic video cnns through representation warping. In *ICCV*, pages 4453–4462, 2017. 2

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE ICCV*, pages 2980–2988, 2017. 2

[20] Ping Hu, Fabian Caba, Oliver Wang, Zhe Lin, Stan Sclaroff, and Federico Perazzi. Temporally distributed networks for fast video semantic segmentation. In *CVPR*, pages 8818–8827, 2020. 2

[21] Zilong Huang, Xinggang Wang, Yunchao Wei, Lichao Huang, Humphrey Shi, Wenyu Liu, and Thomas S. Huang. Ccnet: Criss-cross attention for semantic segmentation. *IEEE TPAMI*, 2020. 2

[22] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. *arXiv preprint arXiv:2106.03299*, 2021. 2

[23] Samvit Jain, Xin Wang, and Joseph E Gonzalez. Accel: A corrective fusion network for efficient semantic segmentation on video. In *CVPR*, pages 8866–8875, 2019. 2

[24] Xiaojie Jin, Xin Li, Huaxin Xiao, Xiaohui Shen, Zhe Lin, Jimei Yang, Yunpeng Chen, Jian Dong, Luoqi Liu, Zequn Jie, et al. Video scene parsing with predictive feature learning. In *ICCV*, pages 5580–5588, 2017. 2

[25] Tsung-Wei Ke, Jyh-Jing Hwang, Ziwei Liu, and Stella X Yu. Adaptive affinity field for semantic segmentation. *ECCV*, 2018. 2

[26] 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. 1, 2, 3, 5, 6, 7, 8

[27] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *IEEE CVPR*, pages 6399–6408, 2019. 1, 2, 6

[28] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 4

[29] Jie Li, Allan Raventos, Arjun Bhargava, Takaaki Tagawa, and Adrien Gaidon. Learning to fuse things and stuff. *arXiv preprint arXiv:1812.01192*, 2018. 1, 2

[30] Xia Li, Zhisheng Zhong, Jianlong Wu, Yibo Yang, Zhouchen Lin, and Hong Liu. Expectation-maximization attention networks for semantic segmentation. In *IEEE ICCV*, 2019. 2

[31] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *IEEE CVPR*, pages 7026–7035, 2019. 1, 2

[32] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. In *IEEE CVPR*, pages 2359–2367, 2017. 2

[33] Yule Li, Jianping Shi, and Dahua Lin. Low-latency video semantic segmentation. In *CVPR*, pages 5997–6005, 2018. 2

[34] Yanwei Li, Hengshuang Zhao, Xiaojuan Qi, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Fully convolutional networks for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 214–223, 2021. 1, 2, 5, 6, 7

[35] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian D Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *IEEE CVPR*, 2017. 2

[36] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 5

[37] Dongfang Liu, Yiming Cui, Wenbo Tan, and Yingjie Chen. Sg-net: Spatial granularity network for one-stage video instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9816–9825, 2021. 2

[38] Si Liu, Changhu Wang, Ruihe Qian, Han Yu, Renda Bao, and Yao Sun. Surveillance video parsing with single frame supervision. In *CVPR*, pages 1013–1021. IEEE Computer Society, 2017. 2

[39] Si Liu, Changhu Wang, Ruihe Qian, Han Yu, Renda Bao, and Yao Sun. Surveillance video parsing with single frame supervision. In *CVPR*, pages 413–421, 2017. 2

[40] Yifan Liu, Chunhua Shen, Changqian Yu, and Jingdong Wang. Efficient semantic video segmentation with per-frame inference. In *ECCV*, 2020. 2

[41] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer, 2020. 2

[42] K-K Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. Video object segmentation without temporal information. *IEEE transactions on pattern analysis and machine intelligence*, 41(6):1515–1530, 2018. 2

[43] Jiaxu Miao, Yunchao Wei, Yu Wu, Chen Liang, Guangrui Li, and Yi Yang. Vspw: A large-scale dataset for video scene parsing in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2, 3

[44] Jiaxu Miao, Yunchao Wei, and Yi Yang. Memory aggregation networks for efficient interactive video object segmentation. In *IEEE CVPR*, 2020. 2

[45] David Nilsson and Cristian Sminchisescu. Semantic video segmentation by gated recurrent flow propagation. In *CVPR*, pages 6819–6828, 2018. 2

[46] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters—improve semantic segmentation by global convolutional network. In *IEEE CVPR*, pages 1743–1751, 2017. 2

[47] 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 *IEEE CVPR*, pages 724–732, 2016. 2

[48] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3997–4008, 2021. 1, 2, 3, 5, 7, 8

[49] Evan Shelhamer, Kate Rakelly, Judy Hoffman, and Trevor Darrell. Clockwork convnets for video semantic segmentation. In *ECCV*, pages 852–868. Springer, 2016. 2

[50] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. In *ECCV*, pages 746–760. Springer, 2012. 2

[51] Konstantin Sofiiuk, Olga Barinova, and Anton Konushin. Adaptis: Adaptive instance selection network. In *IEEE ICCV*, pages 7355–7363, 2019. 1, 2

[52] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE CVPR*, pages 5693–5703, 2019. 2

[53] 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 *IEEE CVPR*, pages 9481–9490, 2019. 2

[54] Paul Voigtlaender and Bastian Leibe. Online adaptation of convolutional neural networks for the 2017 davis challenge on video object segmentation. In *CVPR Workshop*, volume 5, 2017. 2

[55] Wenguan Wang, Jianbing Shen, Fatih Porikli, and Ruigang Yang. Semi-supervised video object segmentation with super-trajectories. *IEEE transactions on pattern analysis and machine intelligence*, 41(4):985–998, 2018. 2

[56] Wenguan Wang, Jianbing Shen, Ruigang Yang, and Fatih Porikli. Saliency-aware video object segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 40(1):20–33, 2017. 2

[57] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. *arXiv preprint arXiv:2101.11939*, 2021. 2

[58] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE CVPR*, pages 7794–7803, 2018. 2

[59] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end

video instance segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8741–8750, 2021. 2

[60] 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. 1, 3, 6, 7

[61] 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. 1, 3, 7, 8

[62] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *ECCV*, 2018. 2

[63] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *IEEE CVPR*, pages 8818–8826, 2019. 1, 2, 3

[64] Ning Xu, Linjie Yang, Yuchen Fan, Dingcheng Yue, Yuchen Liang, Jianchao Yang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. In *ECCV*, 2018. 2

[65] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, pages 5188–5197, 2019. 2

[66] Maoke Yang, Kun Yu, Chi Zhang, Zhiwei Li, and Kuiyuan Yang. Denseaspp for semantic segmentation in street scenes. In *IEEE CVPR*, pages 3684–3692, 2018. 2

[67] Shusheng Yang, Yuxin Fang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Crossover learning for fast online video instance segmentation. *arXiv preprint arXiv:2104.05970*, 2021. 2

[68] Tien-Ju Yang, Maxwell D Collins, Yukun Zhu, Jyh-Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang-Chieh Chen. Deeperlab: Single-shot image parser. *arXiv preprint arXiv:1902.05093*, 2019. 1, 2

[69] Yi Yang, Yueting Zhuang, and Yunhe Pan. Multiple knowledge representation for big data artificial intelligence: framework, applications, and case studies. *Frontiers of Information Technology & Electronic Engineering*, 22(12):1551–1558, 2021. 2

[70] Zongxin Yang, Yunchao Wei, and Yi Yang. Collaborative video object segmentation by foreground-background integration. In *ECCV*, pages 332–348. Springer, 2020. 2

[71] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *NeurIPS*, 2021. 2, 4, 5

[72] Yuhui Yuan and Jingdong Wang. Ocnet: Object context network for scene parsing. *arXiv preprint arXiv:1809.00916*, 2018. 2

[73] Rui Zhang, Sheng Tang, Yongdong Zhang, Jintao Li, and Shuicheng Yan. Scale-adaptive convolutions for scene parsing. In *IEEE ICCV*, pages 2031–2039, 2017. 2

[74] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. In *NeurIPS*, 2021. 1, 2

[75] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE CVPR*, 2017. 2

[76] Hengshuang Zhao, Yi Zhang, Shu Liu, Jianping Shi, Chen Change Loy, Dahua Lin, and Jiaya Jia. Psanet: Point-wise spatial attention network for scene parsing. In *ECCV*, pages 270–286, 2018. 2

[77] Tianfei Zhou, Jianwu Li, Shunzhou Wang, Ran Tao, and Jianbing Shen. Matnet: Motion-attentive transition network for zero-shot video object segmentation. *IEEE Transactions on Image Processing*, 29:8326–8338, 2020. 2

[78] Tianfei Zhou, Shunzhou Wang, Yi Zhou, Yazhou Yao, Jianwu Li, and Ling Shao. Motion-attentive transition for zero-shot video object segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13066–13073, 2020. 2

[79] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *CVPR*, pages 2349–2358, 2017. 2