Video Panoptic Segmentation

Dahun Kim[1,†] Sanghyun Woo[1,†] Joon-Young Lee[2] In So Kweon[1]

[1]KAIST [2]Adobe Research

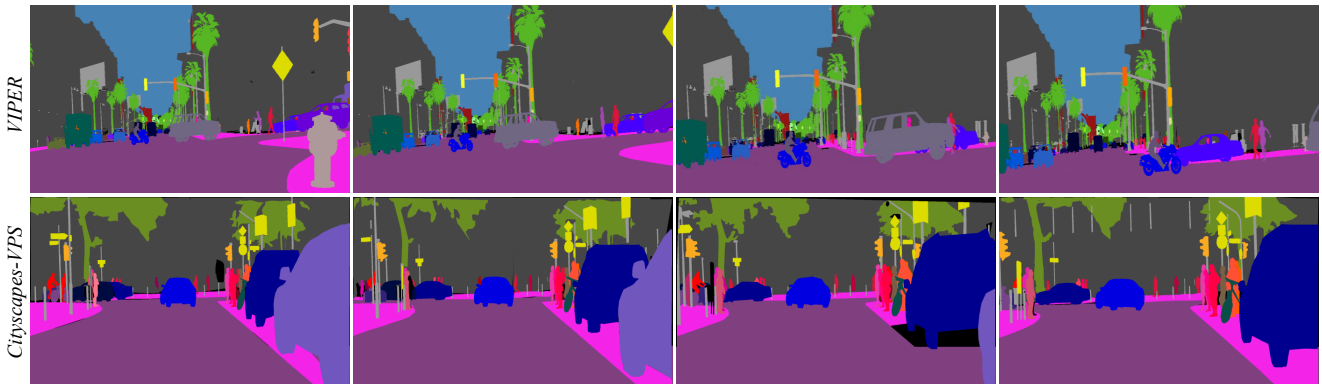

Figure 1: Example video sequences of reformatted VIPER and newly created Cityscapes-VPS annotations for video panoptic segmentation.

Abstract

Panoptic segmentation has become a new standard of visual recognition task by unifying previous semantic segmentation and instance segmentation tasks in concert. In this paper, we propose and explore a new video extension of this task, called video panoptic segmentation. The task requires generating consistent panoptic segmentation as well as an association of instance ids across video frames. To invigorate research on this new task, we present two types of video panoptic datasets. The first is a re-organization of the synthetic VIPER dataset into the video panoptic format to exploit its large-scale pixel annotations. The second is a temporal extension on the Cityscapes val. set, by providing new video panoptic annotations (Cityscapes-VPS). Moreover, we propose a novel video panoptic segmentation network (VPSNet) which jointly predicts object classes, bounding boxes, masks, instance id tracking, and semantic segmentation in video frames. To provide appropriate metrics for this task, we propose a video panoptic quality (VPQ) metric and evaluate our method and several other baselines. Experimental results demonstrate the effectiveness of the presented two datasets. We achieve state-of-the-art results in image PQ on Cityscapes and also in VPQ on Cityscapes-VPS and VIPER datasets. The datasets and code will be released.

[†] This work was done during an internship at Adobe Research.

1. Introduction

As an effort to unify existing recognition tasks, object classification, detection, and segmentation and to leverage the possible complementariness of these tasks into a single complete task, Kirillov *et al*. [16] proposed a holistic segmentation of all foreground instances and background regions in a scene and named the task *panoptic segmentation*. Since then, a large number of works [7, 8, 10, 15, 17–20, 24, 31, 37, 40] have proposed learning-based approaches to this new benchmark task, confirming its importance to the field.

In this paper, we extend the panoptic segmentation in the image domain to the video domain. Different from image panoptic segmentation, the new problem aims at a simultaneous prediction of object classes, bounding boxes, masks, instance id associations, and semantic segmentation, while assigning unique answers to each pixel in a video. Figure 1 illustrates sample video sequences of ground truth annotations for this problem. Naturally, we name the new task *video panoptic segmentation* (VPS). The new task opens up possibilities for applications that require a holistic and global view of video segmentation such as autonomous driving, augmented reality, and video editing. In particular, temporally dense panoptic segmentation of a video can work as intermediate-level representations for even higher-level video understanding tasks such as temporal reasoning or action-actor recognition which anticipates the behaviors

of objects and humans. To best of our knowledge, this is the first work to address video panoptic segmentation problem.

Thanks to the existence of panoptic segmentation benchmarks such as COCO [23], Cityscapes [5], and Mapillary [25], the panoptic *image* segmentation has successfully driven active participation of the community. However, the direction towards the video domain has not yet explored, probably due to the lack of appropriate datasets and evaluation metrics. While video object/instance segmentation datasets are available these days, no dataset permits direct training of video *panoptic* segmentation (VPS). This is not surprising when considering its extremely high cost of collecting such data. To improve the situation, we make an important first step in the direction of panoptic *video* segmentation, by presenting two types of datasets. First, we adapt the synthetic VIPER [32] dataset into the video panoptic format and create corresponding metadata. Second, we collect a new video panoptic segmentation dataset, named *Cityscapes-VPS*, that extends the public Cityscapes to a video level by providing every five video frames with pixel-level panoptic labels that are temporally associated with respect to the public image-level annotations.

In addition, we propose a video panoptic segmentation network (VPSNet) to provide a baseline method for this new task. On top of UPSNet [37], which is a state-of-the-art method for image panoptic segmentation, we design our VPSNet to take an additional frame as the reference to correlate time information at two levels: pixel-level fusion and object-level tracking. To pick up the complementary feature points in the reference frame, we propose a flow-based feature map alignment module along with an asymmetric attention block that computes similarities between the target and reference features to fuse them into *one-frame* shape. Moreover, to associate object instances across time, we add an object track head [38] which learns the correspondence between the instances in the target and reference frames based on their RoI feature similarity. It establishes a baseline for the VPS task and gives us insights into the main algorithmic challenges it presents.

We adapt the standard image panoptic quality (PQ) measure to fit the video panoptic quality (VPQ) format. Specifically, the metric is obtained from a span of several frames, where the sequence of each panoptic segment within the span is considered a single 3D tube prediction to produce an IoU with the ground truth tube. The longer the time-span, the more challenging it is to obtain IoU over a threshold and to be counted as a true-positive for the final VPQ score. We evaluate our proposed method with several other naive baselines using the VPQ metric.

Experimental results demonstrate the effectiveness of the two presented datasets. Our VPSNet achieves state-of-the-art image PQ on Cityscapes and VIPER. More importantly, in terms of VPQ, it outperforms the strong baseline [38] and other simple candidate methods, while still presenting algorithmic challenges of the VPS task.

We summarize the contribution of this paper as follows.

1. To our best knowledge, it is the first time that video panoptic segmentation (VPS) is formally defined and explored.
2. We present the first VPS datasets by re-formatting the virtual VIPER dataset and creating new video panoptic labels based on the Cityscapes benchmark. Both datasets are complementary in constructing an accurate VPS model.
3. We propose a novel VPSNet which achieves state-of-the-art image panoptic quality (PQ) on Cityscapes and VIPER, and compare it with several baselines on our new datasets.
4. We propose a video panoptic quality (VPQ) metric to measure the spatial-temporal consistency of predicted and ground truth panoptic segmentation masks. The effectiveness of our proposed datasets and methods is demonstrated by the VPQ evaluation.

2. Related Work

Panoptic Segmentation: The joint task of thing and stuff segmentation is reinvented by Kirillov *et al*. [16] in the form of combining the semantic segmentation and instance segmentation tasks and is named panoptic segmentation. Since then, much research [7, 8, 10, 15, 17–20, 24, 31, 37, 40] has been actively gathered to propose new approaches to this unified task, which is now a *de facto* standard of visual recognition task. A naive baseline introduced in [16] is to train the two sub-tasks separately then fuse the results by heuristic rules. More advanced approaches to this problem present a unified, end-to-end model. Li *et al*. [20] propose AUNet which leverages mask level attention to transfer knowledge from the instance head to the semantic head. Li *et al*. [18] suggest a new objective function to enforce consistency between things and stuff pixels when merging them into a single segmentation result. Liu *et al*. [24] design a spatial ranking module to address the occlusion between the predicted instances. Xiong *et al*. [37] introduce a non-parametric panoptic head to predict instance id and resolve the conflicts between things and stuff segmentation.

Video Semantic Segmentation: As a direct extension of semantic segmentation to videos, all pixels in a video are predicted as different semantic classes. However, the research in this field has not gained much attention and not currently popular compared to its counterpart in the image domain. One possible reason is the lack of available training data with temporally dense annotation, as research progress depends greatly on the existence of datasets. Despite the absence of a dataset for Video Semantic Segmen-

tation (VSS), several approaches have been proposed in the literature [14, 21, 26, 33, 43]. Temporal information is utilized via optical flow to improve the accuracy or efficiency of the scene labeling performance. Different from our setting, VSS does not require either discriminating object instances or explicit tracking of the objects across frames. Our new *Cityscapes-VPS* is a super-set of a VSS dataset and thus is able to benefit this independent field as well.

Video Instance Segmentation: Even more recently, Yang *et al*. [38] proposed a Video Instance Segmentation (VIS) problem to extend image instance segmentation to videos. It combines several existing tasks: video object segmentation [1, 3, 4, 27, 30, 35, 36, 39] and video object detection [9, 42, 43], and aims at simultaneous detection, segmentation, and tracking of instances in videos. They propose Mask-Track R-CNN which has a tracking branch added to Mask R-CNN [11] to jointly learn these multiple tasks. The object association is trained based on object feature similarity learning, and the learned features are used together with other cues such as spatial correlation and detection confidence to track the objects at inference. The first difference to our setting is that VIS only deals with foreground *thing* objects but not background *stuff* regions. Moreover, the problem permits overlaps between predicted object masks and even multiple predictions for a single instance, while our task requires algorithms to assign a single label to all things and stuff pixels. Last but not least, their dataset contains a small number of objects ($\sim$ 5) per frame, whereas we deal with a much larger number of objects ($>$ 20 on average), which makes our task even more challenging.

3. Problem Definition

Task Format: For a video sequence with T frames, we set a temporal window that spans k consecutive frames. Given a k-span snippet $I^{t:t+k} = \{I^t, I^{t+1}, ..., I^{t+k}\}$, we define a *tube* prediction as a track of its frame-level segments as $\hat{u}_{(c_i,z_i)} = \{\hat{s}^t, ..., \hat{s}^{t+k}\}_{(c_i,z_i)}$, for semantic class c and instance id z of the tube. Note that instance id z_i for a *thing* class can be larger than 0, *e.g.*, *car*-0, *car*-1, ... , whereas it is always 0 for a *stuff* class, *e.g.*, *sky*. All pixels in the video are grouped by such tuple prediction, and they will result in a set of *stuff* and *things* video tubes that are mutually exclusive to each other. The ground truth tube is defined similarly, with a slight adjustment concerning the annotation frequency as described below. The goal of video panoptic segmentation is to accurately localize all the semantic and instance boundaries throughout a video and assign correct labels to those segmented video tubes.

Evaluation Metric: By the construction of the VPS problem, no overlaps are possible among video tubes. Thus, AP metric used in object detection or segmentation cannot be used to evaluate the VPS task. Instead, we borrow the

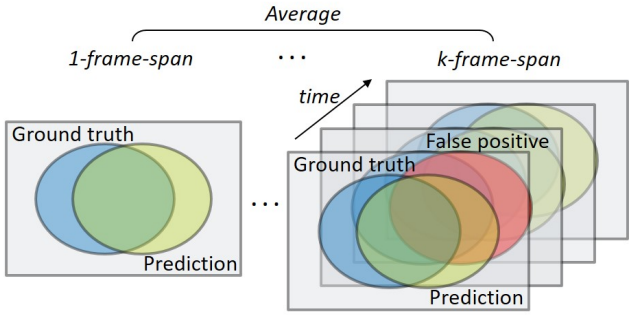

Figure 2: **Tube matching and video panoptic quality (VPQ) metric.** An IoU is obtained by matching predicted and ground truth *tubes*. A frame-level false positive segment penalizes the whole predicted tube to get a low IoU. Each VPQ^k is computed by sliding the window through a video, and averaged by the number of frames. k indicate the temporal window size. VPQ^k is then averaged over different k values, to get a final VPQ score.

panoptic quality (PQ) metric in image panoptic segmentation with modifications adapted to our new task.

Given a snippet $I^{t:t+k}$, we denote a ***set*** of the ground truth and predicted tubes as $\mathcal{U}^{t:t+k}$ and $\hat{\mathcal{U}}^{t:t+k}$. A set of True Positive matches is defined as TP = $\{(u, \hat{u}) \in \mathcal{U} \times \hat{\mathcal{U}}$: IoU $(u, \hat{u}) > 0.5\}$. False Positives (FP) and False Negatives (FN) are defined accordingly. When the annotation is given every λ frames, the matching only considers the annotated frame indices $t : t + k : \lambda$ $(start : end : stride)$ in a snippet, *e.g.*, when k = 10 and λ = 5, frame t, t+5 and t+10 are considered. We slide the k-span window with a stride λ throughout a video, starting from frame 0 to the end, *i.e.*, t goes by $0 : T - k : \lambda$ (We assume frame 0 is annotated). Each stride constructs a new snippet, where we compute the IoUs, TP, FP and FN as above.

At a dataset level, the snippet-level IoU, |TP|, |FP| and |FN| values are collected *across all predicted videos*. Then, the *dataset-level* VPQ metric is computed per each class c, and averaged across all classes as,

$$VPQ^k = \frac{1}{N_{classes}} \sum_c \frac{\sum_{(u,\hat{u})\in TP_c} IoU(u, \hat{u})}{|TP_c| + \frac{1}{2}|FP_c| + \frac{1}{2}|FN_c|}, \quad (1)$$

where $\frac{1}{2}|FP| + \frac{1}{2}|FN|$ in the denominator is to penalize unmatched tubes, as suggested in the image PQ metric.

By definition, k = 0 will make the metric equivalent to the image PQ metric, and k = T-1 will construct a set of whole video-long tubes. Any cross-frame inconsistency of semantic or instance label prediction will result in a low tube IoU, and may drop the match out of the TP set, as illustrated in Figure 2. Therefore, the larger window size we have, the more challenging it is to get a high VPQ score. In practice, we include different window sizes $k \in \{0, 5, 10, 15\}$ to provide a more comprehensive evaluation. The final VPQ is computed by averaging over K = 4 as, $VPQ = \frac{1}{K}\sum_k VPQ^k$.

Having different k values enables a smooth transition from the existing image PQ evaluation to videos, encouraging the image-to-video transition of further technical developments for this pioneering field to leap forward.

Hyper-parameter: We set k as a user-defined parameter. Having such a fixed temporal window size regularizes the difficulty of IoU matching across video samples of different lengths. On the other hand, the difficulty of matching whole T-long tubes, extremely varies with the video length, *e.g.*, when $T = 10$ and $T = 1000$.

We empirically observed that, in our Cityscapes-VPS dataset ($\lambda = 5$), many object associations are disconnected by significant scene changes when $k > 15$. Given a new annotation frequency ($1/\lambda$), the k shall be reset, which will accordingly set a level of difficulty for the dataset.

4. Dataset Collection

Existing Image-level Benchmarks: There are several public datasets which have dense panoptic segmentation annotations: Cityscapes [5], ADE20k [41], Mapillary [25], and COCO [23]. However, none of these datasets matches the requirement for our video panoptic segmentation task. Thus, we need to prepare a suitable dataset for the development and evaluation of video panoptic segmentation methods. We pursue several directions when collecting VPS datasets. First, both the quality and quantity of the annotation should be high, of which the former is a common problem in some of the existing polygon-based segmentation datasets and the latter is limited by the extreme cost of panoptic annotations. More importantly, it should be easily adaptable to and extensible from the existing image-based panoptic datasets, so that it can promote the research community to seamlessly transfer the knowledge between the image and video domains. With the above directions in mind, we present two VPS datasets by 1) reformatting the VIPER dataset and 2) creating new video panoptic annotations based on the Cityscapes dataset.

Revisiting VIPER dataset: To maximize both the quality and quantity of the available annotations for the VPS task, we take advantage of the synthetic VIPER dataset [32] extracted from the GTA-V game engine. It includes pixel-wise annotations of semantic and instance segmentations for 10 *thing* and 13 *stuff* classes on 254K frames of ego-centric driving scenes at 1080×1920 resolution. As shown in Figure 1-(top row), we tailor their annotations into our VPS format and create metadata in a popular COCO style, so that it can be seamlessly plugged into recent recognition models such as Mask-RCNN [11].

Cityscapes-VPS: Instead of building our dataset from scratch in isolation, we build our benchmark on top of the public Cityscapes dataset [5], which is the most popular dataset for panoptic segmentation, together with COCO.

	YT-VIS	City	re-VIPER	City-VPS
Videos	2540	3475	124	500
Frames	108k	3475	184k	3000
Things	40	8	10	8
Stuff	x	11	13	11
Instances	4297	60 K	31 K	10 K
Masks	115 K	60 K	2.8 M	56 K
Temporal	✓	x	✓	✓
Dense (Panoptic)	x	✓	✓	✓

Table 1: High-level statistics of our reformatted VIPER and new Cityscapes-VPS with previous video instance / semantic segmentation datasets. YT-VIS and City stands for YouTube-VIS and Cityscapes respectively. We count only *trainval* data with *labels*.

It consists of image-level annotated frames of ego-centric driving scenarios, where each labeled frame is the 20th frame in a 30 frame video snippet. There are 2965, 500, and 1525 such sampled images paired with dense panoptic annotations for 8 *thing* and 11 *stuff* classes for training, validation, and testing, respectively. Specifically, we select the validation set to build our own video-level extended dataset. We sample every five frames from each of the 500 videos, and then ask human annotators to carefully label each pixel with all 19 classes, and assign temporally consistent instance ids to the *thing* objects, as shown in Figure 1-(bottom row). Our resulting dataset provides dense panoptic annotations for 3000 frames at 1024×2048 resolution with instance id association across frames within each video. The new benchmark is referred to as *Cityscapes-VPS*.

Our new dataset *Cityscapes-VPS* is not only the first benchmark for video panoptic segmentation but also a useful benchmark for other vision tasks such as video instance segmentation and video semantic segmentation; the latter has also been suffering lack of well-established video benchmark. We show some high-level statistics of the reformatted VIPER and new Cityscapes-VPS, and related datasets in Table. 1.

5. Proposed Method

Unlike static images, videos have rich temporal and motion context, and a VPS model should faithfully use this information to capture the *panoptic* movement of all *things* and *stuff* classes in a video. We propose a video panoptic segmentation network (VPSNet). Given an input video sequence, VPSNet performs object detection, mask prediction, tracking, and semantic segmentation all simultaneously. This section describes our network architecture and its implementation in detail.

5.1. Network Design

Overview: By the nature of the VPS task, temporal inconsistency in any of the class label and instance id will result in low video quality of these panoptic segmentation sequences. More strict requirements are therefore in place for

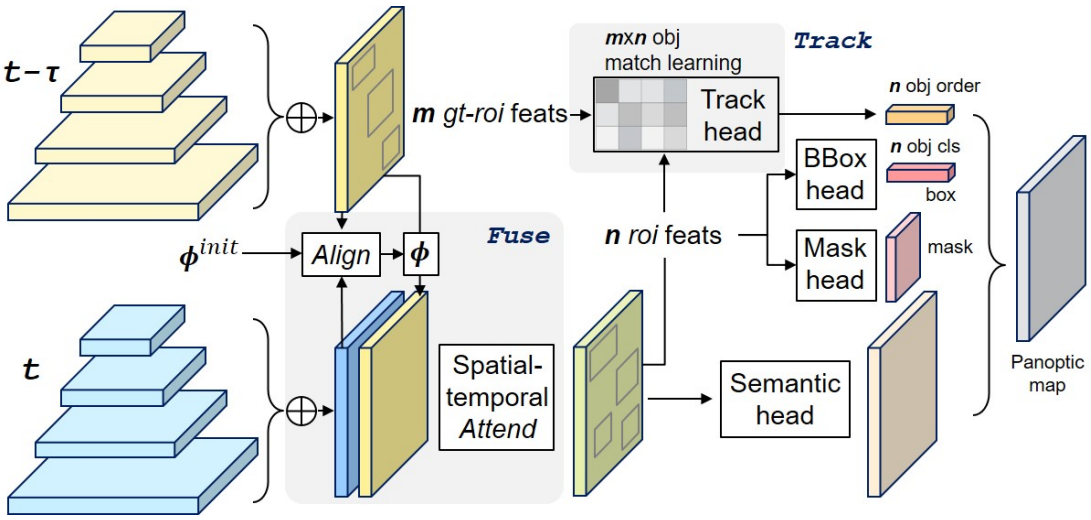

Figure 3: **Overall architecture of our VPSNet.**

the *thing* classes. With this consideration in mind, we design our VPSNet to use video context in two levels: pixel level and object level. The first is to leverage neighboring frame features for the downstream multi-task branches and the second is to explicitly model cross-frame instance association specifically for tracking. Each module for feature fusion and object tracking is not totally new in isolation, but they both are jointly used for the first time for the task of video panoptic segmentation. We call each of them *Fuse* and *Track* module throughout the paper. The overall model architecture is shown in Figure 3.

Baseline: We build upon an image-level panoptic segmentation network. While not being sensitive to any specific design of a baseline network, we choose the state-of-the-art method, UPSNet [37], which adopts Mask R-CNN [11] and deformable convolutions [6] for instance and semantic segmentation branches respectively with a panoptic head that combines these two branches. One of the several modifications is that we do not use *unknown* class predictions for the simplicity of the algorithm. Also, we have an extra non-parametric neck layer, which is inspired by Pang *et al*. [28]. They use *balanced semantic features* to enhance the pyramidal neck representations. Different from theirs, our main design purpose is to have a representative feature map itself at a single resolution level. For this reason, our extra neck consists of only the *gather* and *redistribute* steps with no additional parameters. First, at the *gather* step, the input feature pyramid network (FPN) [22] features $\{p^2, p^3, p^4, p^5\}$ are resized to the highest resolution *i.e.*, the same size as p^2, and element-wise summed over multiple levels, to produce f. Then, this representative feature is *redistributed* to the original features by a residual addition.

Fuse at Pixel Level: The main idea is to leverage video context to improve the per-frame feature by temporal feature fusion. At each time step t, the feature extractor is given a target frame I_t and one (or more) reference frame(s) $I_{t-\tau}$, then produces FPN features $\{p^2, p^3, p^4, p^5\}_t$ and $\{p^2, p^3, p^4, p^5\}_{t-\tau}$. We sample the reference frame with $\tau \in \{t-5 : t+5\}$

We propose an *align-and-attend* pipeline at in between the *gather* and *redistribute* steps. Given the gathered features f_t and $f_{t-\tau}$, our *align* module learns flow warping to align the reference feature $f_{t-\tau}$ onto the target feature f_t. The *align* module receives an initial optical flow $\phi^{init}_{t \to t-\tau}$ computed by FlowNet2 [13], and refine it for more accurate deep feature flow. After concatenating these aligned features, our *attend* module learns spatial-temporal attention to reweight the features and fuse the time dimension into one to get g_t, which is then redistributed to $\{p^2, p^3, p^4, p^5\}_t$ which are then fed forward to the downstream instance and semantic branches.

Track at Object Level: Here, the goal is to track all object instances in I_t with respect to those in $I_{t-\tau}$. Along with the multi-task heads for panoptic segmentation, we add the MaskTrack head [38] which is used in a state-of-the-art video instance segmentation method. It learns a $m \times n$ feature affinity matrix A between generated n RoI proposals $\{r_1, r_2 ... r_n\}_t$ from I_t and m RoI features $\{r_1, r_2 ... r_m\}_{t-\tau}$ from $I_{t-\tau}$. For each pair $\{r_{i,t}, r_{j,t-\tau}\}$, a Siamese fully-connected layer embeds them into single vectors $\{e_{i,t}, e_{j,t-\tau}\}$, then the cosine similarity is measured by $A_{ij} = cosine(e_{i,t}, e_{j,t-\tau})$.

MaskTrack is designed for still images and only utilizes appearance features, and does not use any video features during training. To handle this problem, we couple the tracking branch with the temporal fusion module. Specifically, every RoI features $\{r_1, r_2 ... r_n\}_t$ are first enhanced by the above temporal fused feature, g_t, from multiple frames, and thus become more discriminative before being fed into the tracking branch. Therefore, from a standpoint of the in-

stance tracking, our VPSNet synchronizes it on both pixel-level and object-level. The pixel-level module aligns local feature of the instance to transfer it between the reference and target frames, and the object-level module focuses more on distinguishing the target instance from other reference objects by the similarity function on the temporally augmented RoI features. During training, the tracking head in our VPSNet is the same as [38]. During the inference stage, we add an additional cue from the panoptic head: the IoU of *things* logits. The IoU of instance logits can be viewed as a deformation factor or spatial correlation between frames and our experiments show that it improves the video panoptic quality for *things* classes.

5.2. Implementation Details

We follow most of the settings and hyper-parameters of Mask R-CNN and other panoptic segmentation models such as UPSNet [37]. Hereafter, we only explain those which are different. Throughout the experiments, we use ResNet-50 FPN [12, 22] as the feature extractor.

Training: We implement our models in PyTorch [29] with MMDetection [2] toolbox. We use the distributed training framework with 8 GPUs. Each mini-batch has 1 image per GPU. We use the ground truth box of a reference frame to train the track head. We crop random 800×1600 pixels out of 1024×2048 Cityscapes and 1080×1920 VIPER images after randomly scaling each frame by 0.8 to 1.25 $\times$. Due to the high resolution of images, we downsample the logits for semantic head and panoptic head to 200×400 pixels. Besides the RPN losses, our VPSNet contains 6 task-related loss functions in total: bbox head (classification and bounding-box), mask head, semantic head, panoptic head, and track head. We set all loss weights to 1.0 to make their scales to be roughly on the same order of magnitude.

We set the learning rate and weight decay as 0.005 and 0.0001 for all datasets. For VIPER, we train for 12 epochs and apply lr decay at 8 and 11 epochs. For both Cityscapes and Cityscapes-VPS, we train for 144 epochs and apply lr decay at 96 and 128 epochs. For the pretrained models, we import COCO- or VIPER-pretrained *Base* model parameters and initialize the remaining layers, *e.g.*, Fuse (*align-and-attend*) and Track modules, by Kaiming initialization.

Inference: Given a new testing video, our method processes each frame sequentially in an online fashion. At each frame, our VPSNet first generates a set of instance hypotheses. As a mask pruning process, we perform the class-agnostic non-maximum suppression with the box IoU threshold as 0.5 to filter out some redundant boxes. Then the remaining boxes are sorted by the predicted class probabilities and kept if the probability is larger than 0.6. For the first frame of a video sequence, we assign instance ids according to the order of the probability. For all other frames, the remaining boxes after pruning are matched to identified instances from previous frames based on the learned affinity A, and are assigned instance id accordingly. After processing all frames, our method produces a sequence of panoptic segmentation, each pixel of which contains a unique category label and instance label throughout the sequence. For both IPQ and VPQ evaluation, we test all available models with single scale testing.

Our methods	feat. align	feat. attend	obj. match	PQ	PQ^{Th}	PQ^{St}
Base				52.1	47.2	56.2
Align	✓			52.3	47.3	56.4
Attend		✓		50.7	45.8	54.8
Fuse	✓	✓		53.0	48.3	57.0
Track			✓	53.0	47.9	57.2
FuseTrack	✓	✓	✓	**55.4**	**52.2**	**58.0**

Table 2: Image panoptic segmentation results on VIPER.

Method	Backbone	PQ	PQ^{Th}	PQ^{St}
AUNet [20]	ResNet-101	59.0	54.8	62.1
PanopticFPN [15]	ResNet-101	58.1	52.0	62.5
DeeperLab [40]	Xception-71	56.5	-	-
Seamless [31]	ResNet-50	59.8	54.6	63.6
AdaptIS [34]	ResNet-50	59.0	55.8	61.3
TASCNet [18]	ResNet-50	55.9	50.6	59.8
UPSNet [37]	ResNet-50	59.3	54.6	62.7
TASCNet+CO [18]	ResNet-50	59.2	56.0	61.5
UPSNet+CO [37]	ResNet-50	60.5	57.0	63.0
VPSNet-Base+CO	ResNet-50	60.6	57.0	63.2
VPSNet-Fuse+CO	ResNet-50	61.6	57.7	64.4
VPSNet-Fuse+VP	ResNet-50	**62.2**	**58.0**	**65.3**

Table 3: Image panoptic segmentation results on Cityscapes. '+CO' and '+VP' indicate the model is pretrained on COCO and VIPER, respectively.

6. Experimental Results

In this section, we present the experimental results on the two proposed video-level datasets, *VIPER* and *Cityscapes-VPS*, as well as the conventional image-level Cityscapes benchmark. In particular, we mainly investigate the results in two aspects: image-level prediction and cross-frame association, which will be reflected in the IPQ and VPQ, respectively. We demonstrate the contributions of each of the proposed pixel-level Fuse and object-level Track modules in the performance of video panoptic segmentation. Here is the information on the dataset splits used in experiments.

- **VIPER**: Based on its high quantity and quality of the panoptic video annotation, we mainly experiment with this benchmark. We follow the public train / val split. For evaluation, we choose 10 validation videos from *day* scenario, and use the first 60 frames of each videos: total 600 images.
- **Cityscapes**: We use the public train / val split, and evaluate our image-level model on the validation set.
- **Cityscapes-VPS**: The created video panoptic anno-

VPSNet variants on **VIPER**	Temporal window size				VPQ
	k = 1	k = 5	k = 10	k = 15	
Base	52.1 / 47.2 / 56.2	29.4 / 0.8 / 53.2	29.3 / 0.6 / 53.2	29.0 / 0.5 / 52.8	34.9 / 12.3 / 54.1
Fuse	53.0 / 48.3 / 57.0	30.0 / 0.8 / 54.4	29.8 / 0.8 / 54.0	29.6 / 0.6 / 53.8	35.6 / 12.6 / 54.8
Track	53.0 / 47.9 / 57.2	47.1 / 39.3 / 53.6	42.7 / 30.0 / 53.2	40.4 / 25.4 / 52.8	45.8 / 35.7 / 54.2
FuseTrack Cls-Sort	55.4 / 52.2 / 58.0	30.5 / 0.8 / 55.2	30.1 / 0.6 / 54.6	29.8 / 0.5 / 54.3	36.5 / 13.5 / 55.5
FuseTrack IoU-Match	55.4 / 52.2 / 58.0	45.0 / 32.8 / 55.2	40.1 / 22.8 / 54.6	37.9 / 18.2 / 54.3	44.6 / 31.5 / 55.5
FuseTrack Disjoined	55.4 / 52.2 / 58.0	52.0 / 48.3 / 55.2	48.6 / 40.4 / 54.6	46.9 / 37.5 / 54.3	50.7 / 44.6 / 55.5
FuseTrack (VPSNet)	**55.4** / 52.2 / 58.0	**53.6** / 51.7 / 55.2	**50.1** / 44.7 / 54.6	**48.4** / 41.4 / 54.3	**51.9** / 47.5 / 55.5

VPSNet variants on **Cityscapes-VPS**	Temporal window size				VPQ
	k = 1	k = 5	k = 10	k = 15	
Track	61.6 / 54.9 / 66.5	54.3 / 39.9 / 64.9	50.7 / 34.6 / 62.4	47.8 / 30.7 / 60.4	53.6 / 40.0 / 63.6
FuseTrack (VPSNet)	**62.7** / 56.9 / 66.8	**56.9** / 44.5 / 65.9	**53.3** / 40.4 / 62.7	**51.4** / 36.9 / 61.9	**56.1** / 44.7 / 64.3

Table 4: Video panoptic segmentation results on VIPER (top) and Cityscapes-VPS (bottom). All models are our VPSNet variants. Each cell contains VPQ / VPQ^{Th} / VPQ^{St} scores.

tations are given with the 500 validation videos of Cityscapes. We further split these videos into 400 training videos and 100 validation videos. Each video consists of 30 consecutive frames, with every 5 frames paired with the ground truth annotations. For each video, all 30 frames are predicted, and only the 6 frames with ground truth are evaluated.

Image Panoptic Quality: One thing we can expect from the VPS learning compared to its image-level counterpart is whether it improves per-frame PQ by properly utilizing spatial-temporal features. We evaluate our method with the existing panoptic quality (PQ), recognition quality (RQ), and segmentation quality (SQ). The results are presented in Table 2 and Table 3.

First, we study the importance of the proposed Fuse and Track modules to our image-level panoptic segmentation performance on the VIPER dataset as shown in Table 2. We find that both pixel-level and object-level modules have complementary contributions, each improving the baseline by +1% PQ. Without any of them, the PQ will drop by -3.4%. The best PQ was achieved when these two modules are used together.

We also experiment on the Cityscapes benchmark, to provide a comparison with the state-of-the-art panoptic segmentation methods. Our VPSNet with only the Fuse module can be trained in this setting, since it only requires a neighboring reference frame without any extra annotations. In Table 3, we find that our VPSNet with Fuse module outperforms the state-of-the-art baseline method [37] by +1.0% PQ, which implies that it effectively exploits spatial-temporal context to improve per-frame panoptic segmentation. The pretraining on the VIPER dataset shows its complementary effectiveness to either COCO or Cityscapes dataset by boosting the score by +1.6% PQ from our baseline, achieving 62.6% PQ.

Video Panoptic Quality: We evaluate the spatial-temporal consistency between the predicted and ground truth panoptic video segmentation. The quantitative results are shown in Table 4. Different from the image panoptic segmentation, our new task requires extra consistency in the instance ids across frames, which makes the problem much more challenging for *things* than stuff classes. Not surprisingly, the mean video panoptic quality of things classes (VPQ^{Th}) is generally lower than that of stuff classes (VPQ^{St}).

Since there is no prior work directly applicable to our new task, we present several baseline VPS methods to provide a reference level. Specifically, we enumerate over different methods by replacing only the tracking branch of our VPSNet. The alternative tracking methods are object sorting by classification logit values (Cls-Sort), and flow-guided object matching by mask IoU (IoU-Match). First, Cls-Sort relies on semantic consistency of the same object between frames. However, it fails to track objects possibly because there are a number of instances of the same class in a frame, *e.g.*, car, person, thus making the class logit information not enough for differentiating these instances. On the other hand, IoU-Match is a simple yet strong candidate method for our task by leveraging spatial correlation to determine the instance labels, improving the image-level baseline by +9.7% VPQ.

Our model with Track module further improves this by +1.2% VPQ, by using the learned RoI feature matching algorithm together with the semantic consistency and spatial correlation cues. Our full model with both Fuse and Track modules achieves the best performance by a great improvement of +6.1% VPQ over the variant with only-Track module, and +17.0% over the image-level base model. To show the contribution of the fused feature solely on the object matching performance, we experiment with a VPSNet variant where the fused feature is fed to all task branches except for the tracking branch (Disjoined). The result implies that the Fuse and Track modules share information, and synergize each other to learn more discriminative features for both segmentation and tracking. We observed the consis-

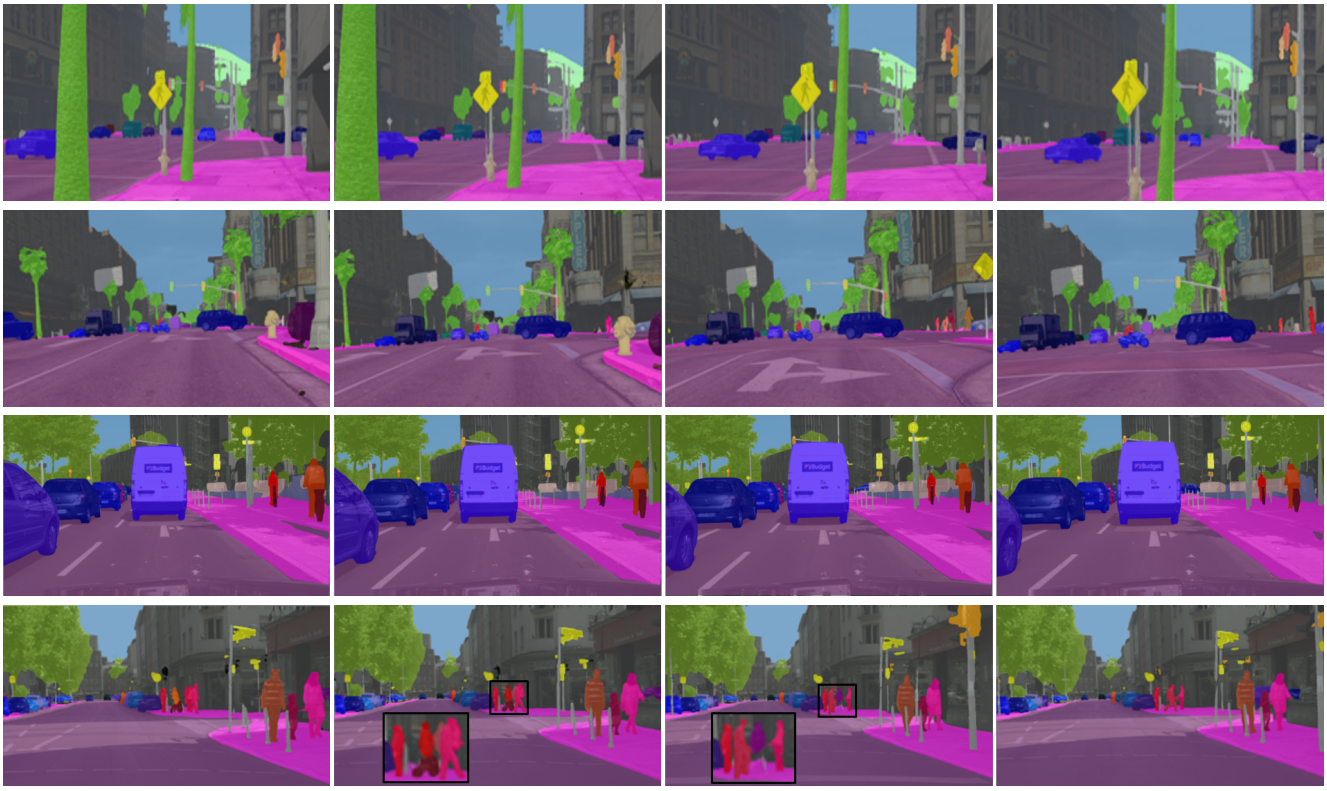

Figure 4: **Sample results of VPSNet on VIPER and Cityscapes-VPS.** Each row has four sampled frames from a video sequence of VIPER (top two rows) and Cityscapes-VPS (bottom two rows). The last row includes failure cases when the crowded objects are crossing each other. Objects with the same predicted identity have the same color.

tent tendency with our Cityscapes-VPS dataset, where our full VPSNet (FuseTrack) achieves +2.5% VPQ higher than the Track variant. Figure 4 shows the qualitative results of our VPSNet on VIPER and Cityscapes-VPS.

Discussion: We find several challenges still remaining for our new task. First, even the state-of-the-art video instance tracking algorithm [38] and our VPSNet suffer a considerable performance drop as the temporal length increases. In the context of video, possible improvements are expected to made on handling a large number of instances and resolving overlaps between these objects, *e.g.*, Figure 4-(4th row), by better modeling the temporal information [27, 43]. Second, our task is still challenging for *stuff* classes as well considering the fact that the window size of 15 frames represents only $0.5 \sim 1$ second in a video. The mutual exclusiveness between things and stuff class pixels could be further exploited to encourage both semantic segmentation and instance segmentation to regularize each other.

Another important future direction is to improve the efficiency of an algorithm as in several video segmentation approaches [21, 33] by sampling keyframes and propagate information in between to produce temporally dense panoptic segmentation results.

7. Conclusion

We present a new task named video panoptic segmentation with two types of associated datasets. The first is to adapt the synthetic VIPER dataset into our VPS format, which can provide maximal quantity and quality of panoptic annotations. The second is to create a new video panoptic segmentation benchmark, *Cityscapes-VPS* which extends the popular image-level Cityscapes dataset. We also propose a new method, VPSNet, by combining the temporal feature fusion module and object tracking branch with a single-frame panoptic segmentation network. Last but not least, we suggest a video panoptic quality measure for evaluation to provide early explorations towards this task. We hope the new task and new algorithm will drive the research directions to step forward towards video understanding in the real-world.

Acknowledgements This work was in part supported by the Institute for Information Communications Technology Promotion (2017-0-01772) grant funded by the Korea government. Dahun Kim was partially supported by Global Ph.D. Fellowship Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018H1A2A1062075).

References

[1] 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. 3

[2] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 6

[3] Yuhua Chen, Jordi Pont-Tuset, Alberto Montes, and Luc Van Gool. Blazingly fast video object segmentation with pixel-wise metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1189–1198, 2018. 3

[4] Jingchun Cheng, Yi-Hsuan Tsai, Shengjin Wang, and Ming-Hsuan Yang. Segflow: Joint learning for video object segmentation and optical flow. In *Proceedings of the IEEE international conference on computer vision*, pages 686–695, 2017. 3

[5] 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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 4

[6] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. pages 764–773, 2017. 5

[7] Daan de Geus, Panagiotis Meletis, and Gijs Dubbelman. Panoptic segmentation with a joint semantic and instance segmentation network. *arXiv preprint arXiv:1809.02110*, 2018. 1, 2

[8] Daan de Geus, Panagiotis Meletis, and Gijs Dubbelman. Single network panoptic segmentation for street scene understanding. *arXiv preprint arXiv:1902.02678*, 2019. 1, 2

[9] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Detect to track and track to detect. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3038–3046, 2017. 3

[10] Cheng-Yang Fu, Tamara L Berg, and Alexander C Berg. Imp: Instance mask projection for high accuracy semantic segmentation of things. 2019. 1, 2

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 3, 4, 5

[12] 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. 6

[13] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 5

[14] Samvit Jain, Xin Wang, and Joseph E Gonzalez. Accel: A corrective fusion network for efficient semantic segmentation on video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8866–8875, 2019. 3

[15] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 1, 2, 6

[16] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019. 1, 2

[17] Justin Lazarow, Kwonjoon Lee, and Zhuowen Tu. Learning instance occlusion for panoptic segmentation. *arXiv preprint arXiv:1906.05896*, 2019. 1, 2

[18] 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, 6

[19] Qizhu Li, Anurag Arnab, and Philip HS Torr. Weakly- and semi-supervised panoptic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 102–118, 2018. 1, 2

[20] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7026–7035, 2019. 1, 2, 6

[21] Yule Li, Jianping Shi, and Dahua Lin. Low-latency video semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5997–6005, 2018. 3, 8

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. pages 2117–2125, 2017. 5, 6

[23] 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. pages 740–755. Springer, 2014. 2, 4

[24] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An end-to-end network for panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6181, 2019. 1, 2

[25] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4990–4999, 2017. 2, 4

[26] David Nilsson and Cristian Sminchisescu. Semantic video segmentation by gated recurrent flow propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6819–6828, 2018. 3

[27] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. *arXiv preprint arXiv:1904.00607*, 2019. 3, 8

[28] Jiangmiao Pang, Kai Chen, Jianping Shi, Huajun Feng, Wanli Ouyang, and Dahua Lin. Libra r-cnn: Towards balanced learning for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 821–830, 2019. 5

[29] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017. 6

[30] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. Learning video object segmentation from static images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2663–2672, 2017. 3

[31] Lorenzo Porzi, Samuel Rota Bulo, Aleksander Colovic, and Peter Kontschieder. Seamless scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8277–8286, 2019. 1, 2, 6

[32] Stephan R Richter, Zeeshan Hayder, and Vladlen Koltun. Playing for benchmarks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2213–2222, 2017. 2, 4

[33] Evan Shelhamer, Kate Rakelly, Judy Hoffman, and Trevor Darrell. Clockwork convnets for video semantic segmentation. In *European Conference on Computer Vision*, pages 852–868. Springer, 2016. 3, 8

[34] Konstantin Sofiiuk, Olga Barinova, and Anton Konushin. Adaptis: Adaptive instance selection network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7355–7363, 2019. 6

[35] Pavel Tokmakov, Karteek Alahari, and Cordelia Schmid. Learning video object segmentation with visual memory. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4481–4490, 2017. 3

[36] 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. 3

[37] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019. 1, 2, 5, 6, 7

[38] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. *arXiv preprint arXiv:1905.04804*, 2019. 2, 3, 5, 6, 8

[39] 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. 3

[40] 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, 6

[41] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 4

[42] Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 408–417, 2017. 3

[43] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2349–2358, 2017. 3, 8