Mask2Former-VPS for WOD Challenge 2023 2D-VPS

Xiang Wu, Feng Xiong, Biye Jiang
Alibaba Group
pingxian.wx@alibaba-inc.com

1. Introduction

Mask2Former-Video [1] has achieved state-of-the-art performance on the task of **Video Instance Segmentation**, which treats a video sequence as a 3D spatial-temporal volume and designs shared queries across frames to segment and track object instances simultaneously. We make three key changes to adapt Mask2Former-Video to **Video Panoptic Segmentation** for the challenge of 2D Video Panoptic Segmentation: 1) we design instance query and stuff query to separately learn the instance and stuff categories in a tricky way; 2) we apply the masked attention to the temporal volume to track object instances across time in way of clips and then merge clips; 3) we adopt a stitching method to connect the tracking results of multiple cameras. Our method, named Mask2Former-VPS, achieves 27.57 wSTQ on WOD-2023 **Video Panoptic Segmentation** challenge with a single model.

2. Mask2Former-VPS

Our method is strongly based on the work Mask2Former for Video Instance Segmentation. We do not change the architecture and the loss, while having only a bit modification on data-set setting and a proper post-process method. So our model architecture is just the same as Mask2Former for Video Instance Segmentation introduced by FAIR.

Our data-set setting:

During training and inference, 5 cameras are treated the same, we just split one original sequence into 5 by camera name, and ”sequence” we mentioned below all refer to the single camera if not specified. In training period, we process a sequence by randomly choose 2 frames in it each time. In inferring period, we split 99 (some sequences may be shorter than this) frames into overlapped clips by timestamp order, precisely, 2 frames per clip and 1 frame is overlapped. After inference on clips, we use a post-process method to merge temporal clips and another post-process method to do tracking between multiple cameras.

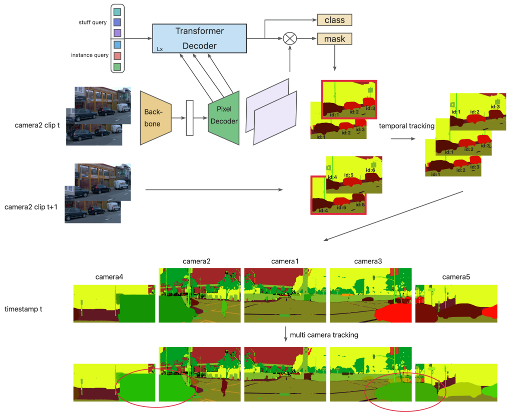

Figure 1. Mask2Former-VPS generalizes to panoptic task by trickily setting stuff id. The model performs temporal tracking within clips. Tracking between clips and between multi-cameras are done in the post-process stage

2.1. Define Panoptic Task with Tracking

Mask2Former-Video has a strong performance on video instance segmentation. When turning to video panoptic segmentation, this network need to be generalized to predict stuffs as well.

To make this simple, we set stuffs as a special instance in data-set. Single stuff class always shares same instance id in one clip, which is consistent with visual cognition. This is equivalent to setting a stuff query module besides a instance query module, and stuffs masks also participate in bipartite matching with things masks together.

We use small clips consists of 2 random frames from the same sequence during training because of GPU memory limitation. In inference period, each clip consists of 2 consecutive frames. By doing so, we perform local tracking first, instead of directly processing the whole video.

Other training pipeline settings are the same as Mask2Former-Video.

2.2. Temporal Tracking Strategy

When dealing with long videos which are about 99 frames in WOD validation set and testing set, we need to do tracking between clips after inferring on individual clips. We use 2 frames for a clip, and there is 1 overlapping frame for tracking between neighbor clips.

For clips in one sequence, our merging method is shown in algorithm 1.

After temporal tracking, we will get a complete video tracking result. Note that different cameras have already been split into different videos and have not been merged yet.

Algorithm 1 Merge Temporal Clips

```
INPUT $clips$
$clips \leftarrow$ SortByTime($clips$)
$reference\_clip \leftarrow clips[0]$
while $\exists\ reference\_clip.next\_clip$ do
    InheritIdByIoUClipWise($reference\_clip$,
    $reference\_clip.next\_clip$)
    $reference\_clip \leftarrow reference\_clip.next\_clip$
end while
$clips \leftarrow$ DropDuplicatedFrames($clips$)
OUTPUT $clips$
```

Algorithm 2 Merge Spatial Clips

```
INPUT $clips$
$clips \leftarrow$ SortByTime($clips$)
$reference\_clip \leftarrow clips[0]$
$reference\_frame \leftarrow reference\_clip[0]$
while $\exists\ reference\_frame.neighbor\_frame$ do
    InheritIdByIoUFrameWise($reference\_frame$,
    $reference\_frame.neighbor\_frame$)
end while
while $\exists\ reference\_clip.next\_clip$ do
    for $frame \in reference\_clip$ do
        InheritIdByIoUFrameWise($frame$,
        $frmae.next\_frame$)
    end for
    $reference\_clip \leftarrow reference\_clip.next\_clip$
    $reference\_frame \leftarrow reference\_clip[0]$
    while $\exists\ reference\_frame.neighbor\_frame$ do
        InheritIdByIoUFrameWise($reference\_frame$,
        $reference\_frame.neighbor\_frame$)
    end while
end while
OUTPUT $clips$
```

2.3. Spatial Tracking Strategy

We use a simple spatial tracking strategy, as we accept the hypothesis that nearby cameras share 20 percent joint area. The stitching method is, after clips are merged together to a complete sequence again (each sequence still includes only one camera), for every 5 sequences split from one original sequence, our stitching method is shown in algorithm 2. Before algorithm 2, we process 5 sequences into spatial clips, with each clip consisting of 5 frames ordered from spatial left to right and sharing same timestamp.

After spatial tracking, we will get a long video tracking result, in which multi cameras are also tracked. Now, we get the final result for submission.

3. Training and Inference Details

We use Detectron2 [2] and follow the IFC [3] settings for video panoptic segmentation.

We use swin-large as backbone, and the pixel decoder and transformer decoder are same as Mask2Former-Video.

More specifically, we use the AdamW [4] optimizer and the cosine annealing learning rate schedule. We use an initial learning rate of 0.0001 and a weight decay of 0.05 for all backbones. A learning rate multiplier of 0.1 is applied to the backbone and we decay the learning rate by step, to 1e-6 finally. We train our models for 40k iterations with a global batch size of 6. During training, each video clip is composed of T = 2 frames, with a general size 1024x1024 (just resize from raw image). Cropping is not used. Our model is initialized with COCO instance segmentation models from [5].

During inference, we do not use any tricks.

For more details, please refer to the following configuration file of Mask2Former-Video for ytvis2019 introduced by FAIR: https://github.com/facebookresearch/Mask2Former/blob/main/configs/youtubevis_2019/swin/video_maskformer2_swin_large_IN21k_384_bs16_8ep.yaml

4. Experiment Results

Model	wSTQ	wAQ	mIOU
VIP-DeepLab Baseline	0.1750	0.1062	0.2883
Mask2Former-IoU(**ours**)	0.2428	0.1039	**0.5672**
Mask2Former-VPS(**ours**)	**0.2757**	**0.1466**	0.5185

Table 1. Our models' performance on WOD 2023 testing set

Our Mask2Former-VPS model finally achieves 27.57 wSTQ on WOD-2023 **Video Panoptic Segmentation** challenge with a single model. Besides, our naive version of Mask2Former-IoU achieves a higher mIoU than Mask2Former-VPS, trained on single frame and doing tracking by IoU.

Mask2Former-VPS shows a much better result in tracking task than Mask2Former-IoU as shown in figure 2.

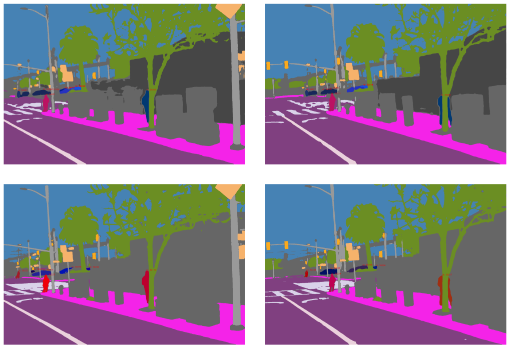

Figure 2. Mask2Former-VPS (first row) has a strong performance on local tracking, even if the object is small, mostly behind another object, and the ego-vehicle is turning sharply. As a contrast, Mask2Former-IoU (second row) performs badly in hard cases.

It seems that simply treating multiple frames as a 3D frame harms semantic segmentation performance in Mask2Former-VPS. We visualized all validation set results and it turns out that Mask2Former-VPS has a trend to predict neighbor cars as one instance as shown in figure 3, but as a contrast, Mask2Former-IoU does not suffer from this problem and has higher mIoU.

We guess this pain comes from 3D-frame, because when training Mask2Former-VPS, we try to teach model predicting multiple neighbor instances as one in temporal dimension, but it turns out the model is generalizing this ability to spatial dimension as well. One rescue may be adjusting loss function to reward temporal dimension's recall while punishing spatial dimension's low precision. We strongly believe that Mask2Former-VPS has a high chance to achieve a much better performance.

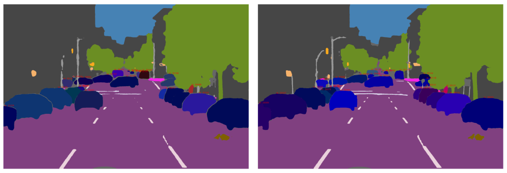

Figure 3. Mask2Former-VPS (left) tends to predict neighbor cars as one car (in same color). As a contrast, Mask2Former-IoU (right) performs well.

References

[1] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, Alexander G. Schwing, Mask2Former for Video Instance Segmentation. arXiv preprint arXiv:2112.10764, 2021.

[2] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[3] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. In NeurIPS, 2021.

[4] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In ICLR, 2019.

[5] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. arXiv preprint arXiv:2112.01527, 2021.