The Runner-up Solution for YouTube-VIS Long Video Challenge 2022

Junfeng Wu[1*]   Yi Jiang[2]   Qihao Liu[3*]   Xiang Bai[1]   Song Bai[2]

[1]Huazhong University of Science and Technology
[2]ByteDance Inc.   [3]Johns Hopkins University

Abstract

This technical report describes our 2nd-place solution for the ECCV 2022 YouTube-VIS Long Video Challenge. We adopt the previously proposed online video instance segmentation method IDOL for this challenge. In addition, we use pseudo labels to further help contrastive learning, so as to obtain more temporally consistent instance embedding to improve tracking performance between frames. The proposed method obtains 40.2 AP on the YouTube-VIS 2022 long video dataset and was ranked second place in this challenge. We hope our simple and effective method could benefit further research.

1. Introduction

Video instance segmentation aims at detecting, segmenting, and tracking object instances simultaneously in a given video. It has attracted considerable attention since first defined [18] in 2019 due to the huge challenge and the wide applications in video understanding, video editing, autonomous driving, augmented reality, etc. Current VIS methods can be categorized as online or offline methods. Online methods [3,5,8,9,17–19] take as input a video frame by frame, detecting and segmenting objects per frame while tracking instances and optimizing results across frames. Offline methods [1,2,7,10,15,16], in contrast, take the whole video as input and generate the instance sequence of the entire video with a single step.

In this challenge, the videos are longer with the lower sampling rate, which masks algorithm easily lose targets ID due to the large movement of the objects and accumulate errors during the longer tracking process. In addition, due to the extremely low sampling rate, the appearance similarity of objects between adjacent frames are smaller. All these characteristics degrade the performance of previous algorithms significantly and make this dataset very challenging.

To handle longer videos, as well as perform more robust tracking on low sample rate video frames, we use IDOL [17] as our baseline algorithm. IDOL is a online video instance segmentation method based on contrastive learning, which is able to ensure, in the embedding space, the similarity of the same instance across frames and the difference of different instances in all frames, even for instances that belong to the same category and have very similar appearances. It provides more discriminative instance embeddings with better temporal consistency, which guarantees more accurate association results. To improve the temporal consistency of instance embeddings between low sampling rate frames, we propose to pre-train the model on COCO [11] with pseudo-labels, which further improves the performance of IDOL.

Our method achieves the second place in the ECCV 2022 YouTube-VIS Long Video Challenge, with the score 53.6 AP on the public validation set, and 40.2 AP on the private test set. We believe the simplicity and effectiveness of our method shall benefit further research.

2. Method

2.1. Video Instance Segmentation

We take IDOL [17] as our baseline method. Following most previous online VIS models [18, 19] which utilize additional association head upon on instance segmentation models [6,14], IDOL takes DeformableDETR [20] with dynamic mask head [14] as its instance segmentation pipeline.

Given an input frame $x \in R^{3 \times H \times W}$ of a video, a CNN backbone extracts multi-scale feature maps. The Deformable DETR takes the feature maps with additional fixed positional encodings [4] and N learnable object queries as input. The object queries are first transformed into output embeddings $E \in R^{N \times C}$ by the transformer decoder. After that, they are decoded into box coordinates, class labels, and instance masks following SeqFormer [16] as shown in Fig 1. Then it calculate pair-wise matching cost which takes into account both the class prediction and the similarity of predicted and ground truth boxes.

An extra light-weighted FFN is employed as a con-
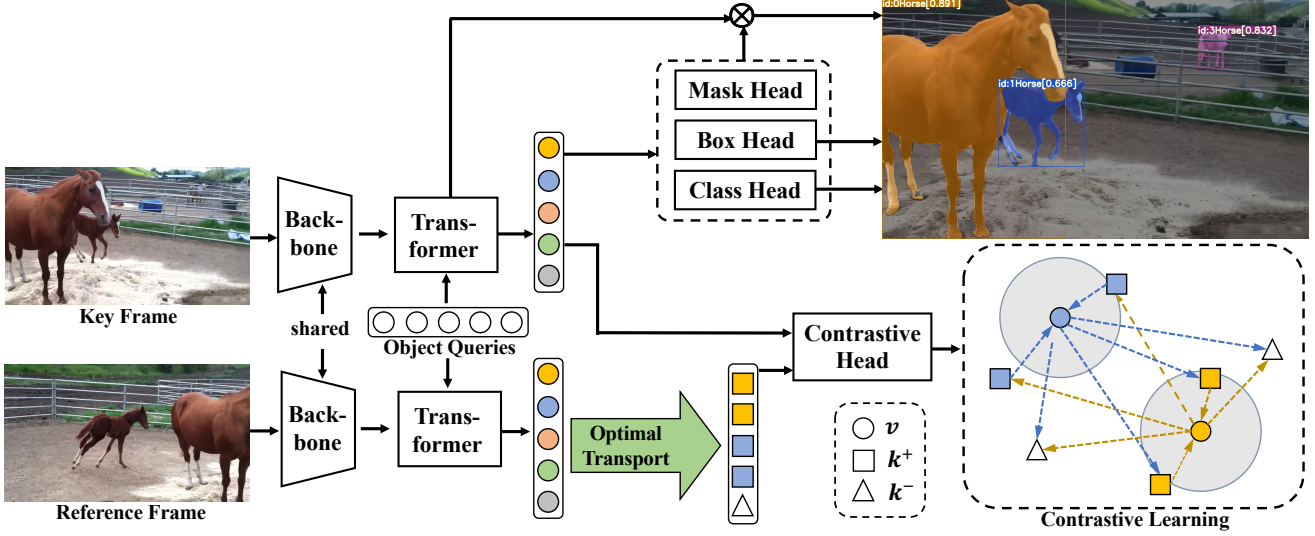
---

* Work done during an internship at ByteDance.

Figure 1. The training pipeline of IDOL. Given a key frame and a reference frame, the shared-weight backbone and transformer predict the instance embeddings on them respectively. The embeddings on the key frame are used to predict masks, boxes, and categories, while the embeddings on the reference frame are selected as positive and negative embeddings for contrastive learning.

trastive head to decode the contrastive embeddings from the output embeddings. Given a key frame for instance segmentation training, a reference frame from the temporal neighborhood is selected for contrastive learning. For each instance in the key frame, their output embedding with the lowest cost are decoded to the contrastive embedding $\mathbf{v}$. If the same instance appears on the reference frame, we select positive and negative samples for it according to the cost with predictions. The contrastive loss function for a positive pair of examples is defined as follows:

$$\mathcal{L}_{embed} = \log[1 + \sum_{\mathbf{k}^+} \sum_{\mathbf{k}^-} \exp(\mathbf{v} \cdot \mathbf{k}^- - \mathbf{v} \cdot \mathbf{k}^+)]. \quad (1)$$

where $\mathbf{k}^+$ and $\mathbf{k}^-$ are positive and negative feature embeddings from the reference frame, respectively. Finally, the whole model is optimized with a multi-task loss function

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_1 \mathcal{L}_{box} + \lambda_1 \mathcal{L}_{mask} + \lambda_2 \mathcal{L}_{embed}, \quad (2)$$

Given a test video, we initialize an empty memory bank for it and perform instance segmentation on each frame sequentially in an online scheme. Assume there are N instances predicted by the model with N contrastive embeddings, and M instances in the memory bank. We compute similarity score f between predicted instance i and memory instance j, and search for the best assignment for instance i by:

$$\hat{j} = \arg\max \mathbf{f}(i, j), \forall j \in \{1, 2, ..., M\}. \quad (3)$$

If $\mathbf{f}(i, \hat{j}) > 0.5$, we assign the instance i on current frame to the memory instance $\hat{j}$. For the prediction without an assignment but has a high class score, we start a new instance ID in the memory bank.
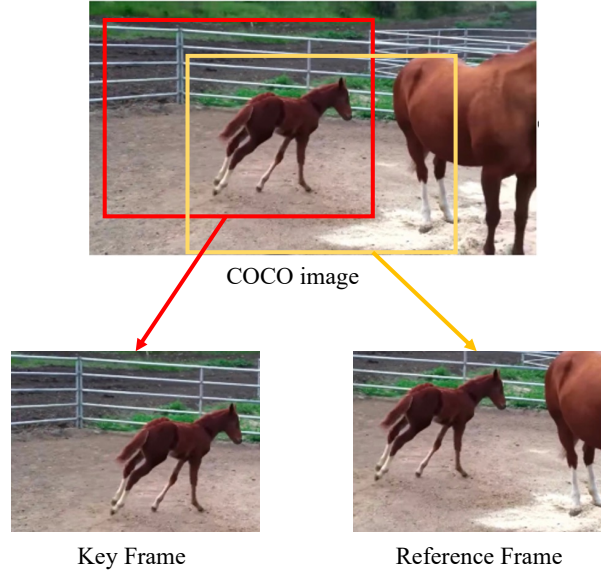

COCO image

Key Frame   Reference Frame

Figure 2. Form pseudo key-reference frame pair by random crop.

2.2. Pseudo Key-reference Frame Pair

Since the videos of this challenge have lower sampling rate, the gap in the appearance similarity of objects between the two frames is larger, which requires contrastive embeddings with higher temporal consistency. To achieve this, we randomly and independently crop the image from COCO twice to form a pseudo key-reference frame pair, which is used to pre-train the contrastive embedding of our models. As shown in Fig 2, if we randomly crop twice on a coco

image, we can get two sub-images on which the same object appears in different positions. They can be regarded as two adjacent key-reference frame pair with camera motion. Pre-training IDOL on these pseudo frames requires that the embeddings belonging to the same horse are as close as possible in the embedding space. This forces the model to learn a position-insensitive contrastive embedding that relies on appearance of the object rather than the spatial position.

3. Experiments

3.1. Implementation Details

We use the same setting for Deformable DETR and the dynamic mask head following IDOL [17]. The models are first pre-trained on the MS COCO 2017 [11] dataset for instance segmentation and then pre-trained on pseud key-reference frames from COCO. Finally, the models are trained on YouTube-VIS 2022 training set for 12000 iterations, the learning rate is decayed by a factor of 0.1 at the 6000 iterations. For data augmentation in training, we use multi-scale training scales at [320, 352, 392, 416, 448, 480, 512, 544, 576, 608, 640] for shortest side. During inference, the input frames are downscaled that the shortest side is at 480 pixels by default. For multi-scale testing, the shortest side is at [480, 640, 800]. The model is trained on 8 A100 GPUs, with 4 pairs of frames per GPU.

3.2. Main Results

Team	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}
IIG	**42.9**	60.7	**46.8**	**35.0**	51.4
Ours	40.2	**61.1**	41.7	32.6	**55.0**
kyriemelon	38.0	55.8	39.2	32.4	48.0
Starburst	38.0	59.9	37.8	33.5	47.7
Carl-Huang	36.3	56.8	37.1	32.2	45.9

Table 1. Comparison with other methods on the ECCV 2022 Youtube-VIS test set.

We evaluate the performance of our method by participating in the ECCV 2022 YouTube-VIS Long Video Challenge. As shown in Table 1, our method achieves 40.2 AP on the test set and achieve second place.

3.3. Ablation Study

In this section we study how we achieve the final results as show in Table 2. The baseline is with ResNet-50 backbone and single-scale testing. Integrated with the Swin Transformer backbone [12], our method achieves a much higher AP of 48.4. The pseudo key-reference frame pair pre-train improves the AP from 48.4 to 50.7. After that, we utilize multi-scale testing for further boosting performance.

Different from image instance segmentation, the IoU computation is carried out in both spatial domain and temporal domain. Multi-scale testing can further improve the result from 50.7 to 52.0. Finally, by ensembling Swin-L and ConvNext-L [13] in the same way, we achieve the 53.6 AP. All the results are evaluate on ECCV 2022 Youtube-VIS validation set.

Method	mAP	ΔmAP
ResNet-50	39.2	-
Swin-L	48.4	9.2
+pseudo frame	50.7	2.3
+multi-scale	52.0	1.3
+multi-model	53.6	1.6

Table 2. Ablation study on the ECCV 2022 Youtube-VIS validation set.

4. Conclusions

In this work, we adopt IDOL for the ECCV 2022 YouTube-VIS Long Video Challenge. In addition, we use pseudo labels to further imporve contrastive learning, so as to obtain more temporally consistent instance embedding to improve tracking. We believe the simplicity and effectiveness of our method shall benefit further research.

References

[1] Ali Athar, Sabarinath Mahadevan, Aljosa Osep, Laura Leal-Taixé, and Bastian Leibe. Stem-seg: Spatio-temporal embeddings for instance segmentation in videos. In *ECCV*, 2020. 1

[2] Gedas Bertasius and Lorenzo Torresani. Classifying, segmenting, and tracking object instances in video with mask propagation. In *CVPR*, 2020. 1

[3] 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*, 2020. 1

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1

[5] Yang Fu, Linjie Yang, Ding Liu, Thomas S Huang, and Humphrey Shi. Compfeat: Comprehensive feature aggregation for video instance segmentation. *arXiv preprint arXiv:2012.03400*, 2020. 1

[6] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1

[7] Sukjun Hwang, Miran Heo, Seoung Wug Oh, and Seon Joo Kim. Video instance segmentation using inter-frame communication transformers. In *NeurIPS*, 2021. 1

[8] Lei Ke, Xia Li, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Prototypical cross-attention networks

for multiple object tracking and segmentation. In *NeurIPS*, 2021. 1

[9] Minghan Li, Shuai Li, Lida Li, and Lei Zhang. Spatial feature calibration and temporal fusion for effective one-stage video instance segmentation. In *CVPR*, 2021. 1

[10] Huaijia Lin, Ruizheng Wu, Shu Liu, Jiangbo Lu, and Jiaya Jia. Video instance segmentation with a propose-reduce paradigm. In *ICCV*, 2021. 1

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1, 3

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 3

[13] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 3

[14] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *ECCV*, 2020. 1

[15] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-end video instance segmentation with transformers. In *CVPR*, 2021. 1

[16] Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *ECCV*, 2022. 1

[17] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. In *ECCV*, 2022. 1, 3

[18] Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *ICCV*, 2019. 1

[19] 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. In *ICCV*, 2021. 1

[20] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 1