2nd Place Solution for OVIS Challenge

Jixiang Sun[1]*, Yong Liu[12]*, Cong Wei[1]*, Yitong Wang[2], Yansong Tang[1], Yujiu Yang[1]
[1] Shenzhen International Graduate School, Tsinghua University
[2]ByteDance Inc.
{liu-yong20,weic22,sun-jx22}@mails.tsinghua.edu.cn,
wangjingshen@bytedance.com,
{tang.yansong,yang.yujiu}@sz.tsinghua.edu.cn,

Abstract

Video Instance Segmentation (VIS) is an important vision task that aims to simultaneously perform classification, tracking, and segmentation in videos. To solve VIS in heavily occluded scenes, we believe that the segmentation performance of single frame is of great importance. Besides, due to the great difference between each frame in difficult scenarios, offline methods may introduce more noise during extracting video features. Thus, we take online method as baseline to process these heavily occluded videos. In this report, we will describe how we can further improve the performance of the state-of-the-art online methods. With different model ensemble, the proposed method finally obtains 47.75 AP on the OVIS test set and was ranked second place in the OVIS challenge.

1. Introduction

Video instance segmentation (VIS) [28] is one of the most fundamental tasks of computer vision. It aims at simultaneously classifying, tracking, and segmenting objects in videos. Due to its wide applications in video editing, autonomous driving, and augmented reality, VIS has attracted great attention in recent years.

With the development of deep learning, there have been many excellent works focusing on video instance segmentation. Generally speaking, current methods can be divided into two categories: offline methods and online methods. Offline methods [1, 2, 8, 12, 15, 25, 26] take the whole video as input and output video instances simultaneously. In contrast, online methods [3, 5, 11, 13, 14, 27, 29] take each frame as input and perform classification and segmentation frame by frame. Finally, the instances of different frames are linked by the design of tracking head.

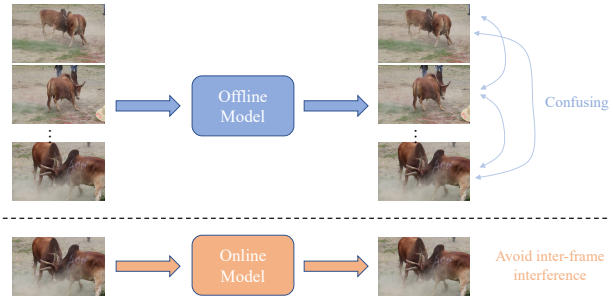

Figure 1. Illustration of offline methods and online methods. The offline methods take multiple frames as input and fuse inter-frame information during extracting features. For difficult scenarios, this may lead to introducing a lot of noise. On the contrary, the online methods process each frame independently.

Since taking the entire video as input, the offline method has the advantage of aggregating inter-frame information to assist in the segmentation of each frame. However, for difficult scenes such as heavy occlusion, the difference between frames is of great variation, which means that there is a lot of noise when aggregating inter-frame information. Fig. 1 illustrates this problem and the comparison between the online methods and offline methods. Thus, we take online methods to process the challenging videos in OVIS dataset [23]. Specifically, we adopt the IDOL [27] and MinVIS [11], the state-of-the-art online models, as our baseline. The core idea of IDOL and MinVIS is the same. They process each frame independently and do not introduce information from other frames when updating queries of each frame. This approach allows queries of each frame to contain sufficient intra-frame instances information and to avoid interference caused by the inter-frame differences. The main difference between IDOL and MinVIS is how to associate the queries of different frames during post-processing.

However, whatever the method of associating different frames, it drastically depends on the representation ability

*Equal Contribution.
This work was done during Yong Liu's internship at ByteDance Inc.

of each frame's query. In other words, the single frame segmentation performance is of great importance, especially for the difficult scenarios. To this end, we introduce some modules to improve the single frame processing part based on IDOL [27] and MinVIS [11].

First of all, to fully learn the information of the training sample and make the model adapt to the heavily occluded scenes, we take the idea of BatchFormer [9,10] that excavating the relationship between training samples by performing attention operation on the batch dimension. Besides, applying this idea also helps the model to achieve better performance with a smaller batch, which is beneficial for the models with huge backbone.

Secondly, it is important to correctly identify the object boundaries in occlusion scenes. For the occluded objects, the overall semantic information has been confused, which has a significant impact on the segmentation and classification. Therefore, focusing more on the object boundaries helps the model to better distinguish the nearby instances. In this case, we proposed Boundary Branch module, which takes the boundary of masks into account.

Thirdly, inspired by the [19], we also apply the quality assessment module only at the training stage. The quality assessment module can improve the robustness of the training process by making the model predict its own accuracy.

Thanks to the superior performance of online methods [11, 27] and the above improvements, we achieved the second place in the 2nd OVIS Challenge with the score of 47.75 AP on the final test set. Besides, on the validation set, we also get the excellent performance of 45.07 AP.

2. Method

2.1. Overview

2.1.1 IDOL

Here we briefly review the structure of IDOL [27], which currently holds the state-of-the-art performance on the OVIS task. It is an online video instance segmentation model, integrating DeformableDETR [30] with a contrastive learning framework and a cross-frame instance association strategy.

During training, IDOL takes a key frame and a reference frame as its input. They are passed into a share-weighted backbone which extracts their feature maps. Then, the feature maps are passed to the Deformable DETR module along with additional fixed positional encodings [4] and N learnable object queries. A transformer decoder transforms the object queries into output embeddings, which are then separately decoded into box coordinates and class labels by 3-layer feed-forward network (FFN). The feature maps goes through a FPN-like [17] mask branch and is transformed into feature map F_{mask}. The output embeddings are passed through another FFN and becomes parameters $!$ of mask head:

$$m_i = MaskHead(F_{mask}; !_i)$$

IDOL also introduced contrastive learning for aggregating the embeddings of the same object in different frames. A light-weighted FFN is used as contrastive head, which decodes the output embeddings in the Deformable DETR module. For each instance in the key frame, the output embedding with the lowest cost is sent to the contrastive head, and gets a contrastive embedding $\mathbf{v}$. If the same instance is in the reference frame, then the top $m1$ predictions with the lowest cost are selected as positives, and top $m2$ predictions with the highest cost are selected as negatives. The values of $m1$ and $m2$ are calculated by the optimal transport method [6, 7]. The postive and negative embeddings are sent to the contrastive head, and gets $\mathbf{k}^+$ and $\mathbf{k}$ respectively. The contrastive loss is calculated as below:

$$L_{embed} = \log[1 + \sum_{\mathbf{k}^+} \sum_{\mathbf{k}^-} \exp(\mathbf{v} \quad \mathbf{k} \quad \mathbf{v} \quad \mathbf{k}^+)]$$

Finally, the loss function is calculated as below, where L_{cls}, L_{box}, L_{mask} represents matching costs of class, box coordinates, and masks with their ground truth respectively.

$$L = L_{cls} + \quad _1 L_{box} + \quad _1 L_{mask} + \quad _2 L_{embed}$$

During inference, IDOL implements a temporally weighted softmax method in the instance association process. It is used to address the problem of unstable prediction of instances, which often appears in online VIS models. With N contrastive embeddings $\mathbf{d}_i \ 2 \ \mathrm{R}^C$ from N predicted instances, and M groups of mutiple temporal contrastive embeddings $f\mathbf{e}_j^t g_{t=1}^T; \mathbf{e}_j^t \ 2 \ \mathrm{R}^C$ in the memory bank, the bi-directional similarity f between predicted instance i and memory instance j is

$$f(i;j) = [\frac{\exp(\hat{\mathbf{e}}_j \quad \mathbf{d}_i) + \quad _j}{\sum_{k=1}^{M} \exp(\hat{\mathbf{e}}_k \quad \mathbf{d}_i) + \quad _k} + \frac{\exp(\hat{\mathbf{e}}_j \quad \mathbf{d}_i)}{\sum_{k=1}^{N} \exp(\hat{\mathbf{e}}_j \quad \mathbf{d}_k)}]=2$$

Where $_j$ is the existing time of instance j in the memory, and

$$\hat{\mathbf{e}}_j = \frac{\sum_{t=1}^{T} \mathbf{e}_j^t \quad (\quad + T=t)}{\sum_{t=1}^{T} \quad + T=t}$$

The best match for instance i in the memory bank is

$$\hat{j} = \mathrm{argmax} f(i;j); 8j \ 2 \ 1;2;:::;M:$$

For those $\hat{j}$ with $f(i;\hat{j}) > 0{:}5$, memory instance $\hat{j}$ is assigned to instance i. Otherwise, instance i is added into the memory bank.

2.1.2 BatchFormer

BatchFormerV2 [10], a module whose structure is similar to the traditional attention [24] layer $Z = softmax(\frac{QK^T}{\sqrt{C}})V$,

Figure 2. The structure of our proposed model. It could either represent IDOL or MinVIS combined with BatchformerV2. For IDOL, an extra contrastive module is added to the heads. For MinVIS, a multi-layer Pixel Decoder is added before the Transformer Blocks.

is introduced for exploring sample relationships during the training of scarce data. Instead of performing attention on the feature dimension, BatchFormer performs attention on batch dimension, and concats the attention results of each feature. Specifically, given inputs $Q, K, V \in R^{B \times N \times C}$, the output Z of BatchFormerV2 can be represented as:

$$Z_i = \mathrm{softmax}\left(\frac{Q_i K_i^T}{\sqrt{C}}\right) V_i, \quad Z = \mathrm{concat}(Z_1, \ldots, Z_N);$$

where $Q_i, K_i, V_i \in R^{B \times C}$ are inputs representing a specific feature channel. All inputs are fed into a shared Transformer block for efficiency of computation and simplicity of dense predictions.

2.1.3 Combining VIS Models with Batchformer

Since BatchFormerV2 improve the performance of image segmentation models such as Deformable DETR [30], it is reasonable to infer that BatchFormer has positive effects on video segmentation models as well. Therefore, we combined BatchFormerV2 module with both IDOL [27] and MinVIS [11]. Same as the implementation in the original paper, we inserted BatchFormerV2 blocks into the Transformer module of both models. Specifically, the original structure of MinVIS made sure that the number of Pixel Decoder layers is equal to Transformer Decoder layers. Therefore, when combining BatchFormerV2 with MinVIS, the added BatchFormerV2 block is designed to take the same input as the Transformer block before it.

2.1.4 Boundary Branch

In order for the model to focus more on the shape and boundary of an object, we introduced the Boundary Branch module which computes the boundary loss of an object. Similar to the Mask Head in [27], the proposed Boundary Head performs convolution on output embeddings and gets the predicted boundary. The ground truth of the boundary is obtained by applying Laplacian operator to the mask ground truth. Following [18], we used dice loss [22] combined with binary cross-entropy loss as the loss function of Boundary Head.

$$L_b = L_{\mathrm{Dice}} + L_{\mathrm{BCE}};$$
$$L_{\mathrm{Dice}}(p, q) = 1 - \frac{2 \sum_i p_i q_i}{\sum_i (p_i)^2 + \sum_i (q_i)^2 + \ldots}. \quad (1)$$

2.1.5 Quality Assessment

Following [19, 20], We proposed Quality Assessment, a module which calculates another type of loss for predicted masks, to further improve the accuracy of the predicted masks of a model. Specifically, Quality Assessment module performs a two-layer FFN on the given output embeddings to predict the quality scores S of a mask. Meanwhile, the maskIOU V between the predicted mask and mask ground truth is calculated. The final loss is obtained by calculating the MSE between S and V of all masks.

$$V_i = \mathrm{maskIoU}(M_i, GT_i);$$
$$L_q = \frac{1}{N} \sum_{i=1}^{N} (S_i - V_i)^2. \quad (2)$$

Table 1. Comparison with other methods on the OVIS test set.

Method	mAP	AP50	AP75	AR1	AR10
q	49.56	72.45	54.12	19.95	55.93
Ours	47.75	74.27	49.90	19.42	54.87
tiantian.tt	47.35	71.62	50.49	19.99	54.99
weic	47.29	74.23	49.13	19.18	54.57
Yong	47.24	74.10	49.09	19.18	54.50

Table 2. Ablation study of our applied modules on the OVIS test set.

Method	mAP	AP50	AP75
Baseline	43.68	65.19	47.24
+Quality assessment	44.54	68.38	46.31
+Boundary branch	45.17	67.55	48.57
+BatchFormer	46.26	68.88	49.85
+Multi-scale	46.84	73.64	48.81
+Model ensemble	47.75	74.27	49.90

2.2. Implementation Details.

We took the Swin Transformer-Large [21] as backbone for all models. For our model taking IDOL [27] as baseline, we selected parameters of BatchFormerV2 [10] according to the best parameters in the original paper, and the training setting is generally same as initial IDOL. We used AdamW optimizer with initial learning rate of 1e-4. Note that we did not perform pre-training on COCO dataset [16] but initialized the model by the pre-trained weights of IDOL directly. To train the proposed modules and netune the IDOL part, we randomly cropped the image from COCO twice to generate the pseudo training videos. Then, we train our model on the pseudo video set and the OVIS train set for 175000 and 40000 iterations with batch size of 8, respectively. For training data augmentation, we performed multi-scale training scales and resied the shortest side to [320, 352, 392, 416, 448, 480, 512, 544, 576, 608, 640]. As for the model taking MinVIS [11] as baseline, we adopted the two stage training strategy. Firstly, we applied our improvements on the MinVIS and trained it on the COCO dataset for 50 epochs with batch size of 16. After that, we performed the main training on OVIS training set for 20000 iterations with batch size of 8. All models are trained on 8 80GB A100 GPUs. During inference, the input videos are resized with the short size of 720 pixels in default.

3. Experiment

3.1. Comparison with Other Methods

In the 2nd OVIS Challenge, we rank the second place on the test set. The leaderboard is shown in Tab. 1. It can be seen that with the help of the introduced modules, our model achieved the 47.75 mAP (2nd) and 74.27 AP50 (1st). This demonstrates that the recognition and detection effect can be signi cantly improved by learning sample relationships, focusing on object boundaries, and improving the training robustness of the model.

3.2. Ablation Study

In this section we analyze the effectiveness of our applied modules on the OVIS test set and the results are shown in Tab. 2. The baseline is the IDOL with Swin-L backbone. Integrated with the quality assessment module, our method achieved the score of 44.54 mAP. After applying the boundary branch and corresponding loss, the performance is improved to 45.17 mAP. After that, we introduced the idea of BatchFormer [10] during training and the result is boosted to 46.26. Utilizing 720p and 1080p scale for inference can further improve the result from 46.84 mAP. Finally, by ensembling with the improved MinVIS that also is equipped with the above modules, the performance eventually reached 47.75 mAP, ranking second place in the OVIS Challenge.

4. Conclusion

We made several improvements on VIS models such as IDOL and MinVIS, and with the help of our proposed methods, we achieved the second place in the 2nd OVIS Challenge with the score of 47.75 AP on the nal test set. Besides, we also get the excellent performance of 45.07 AP on the validation set.

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 European conference on computer vision, pages 213–229. Springer, 2020. 2

[5] Yang Fu, Linjie Yang, Ding Liu, Thomas S. Huang, and Humphrey Shi. Compfeat: Comprehensive feature aggregation for video instance segmentation. In AAAI, 2021. 1

[6] Zheng Ge, Songtao Liu, Zeming Li, Osamu Yoshie, and Jian Sun. Ota: Optimal transport assignment for object detection.

In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 303–312, 2021. 2

[7] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430, 2021. 2

[8] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation via object token association. arXiv preprint arXiv:2206.04403, 2022. 1

[9] Zhi Hou, Baosheng Yu, and Dacheng Tao. Batchformer: Learning to explore sample relationships for robust representation learning. In CVPR, 2022. 2

[10] Zhi Hou, Baosheng Yu, Chaoyue Wang, Yibing Zhan, and Dacheng Tao. Batchformerv2: Exploring sample relationships for dense representation learning. arXiv preprint arXiv:2204.01254, 2022. 2, 4

[11] De-An Huang, Zhiding Yu, and Anima Anandkumar. Minvis: A minimal video instance segmentation framework without video-based training. arXiv preprint arXiv:2208.02245, 2022. 1, 2, 3, 4

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

[13] 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

[14] 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

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

[16] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In ECCV, 2014. 4

[17] Tsung-Yi Lin. Y, dollár p, girshick r, he k, hariharan b, belongie s. feature pyramid networks for object detection. In Proceedings-30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Volume 2017, pages 936–944, 2017. 2

[18] Yong Liu, Ran Yu, Jiahao Wang, Xinyuan Zhao, Yitong Wang, Yansong Tang, and Yujiu Yang. Global spectral filter memory network for video object segmentation. arXiv preprint arXiv:2210.05567, 2022. 3

[19] Yong Liu, Ran Yu, Fei Yin, Xinyuan Zhao, Wei Zhao, Weihao Xia, and Yujiu Yang. Learning quality-aware dynamic memory for video object segmentation. arXiv preprint arXiv:2207.07922, 2022. 2, 3

[20] Yong Liu, Ran Yu, Xinyuan Zhao, and Yujiu Yang. Quality-aware and selective prior enhancement memory network for video object segmentation. In CVPR Workshop, 2021. 3

[21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In ICCV, 2021. 4

[22] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In 2016 fourth international conference on 3D vision (3DV), pages 565–571. IEEE, 2016. 3

[23] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip Torr, and Song Bai. Occluded video instance segmentation: A benchmark. International Journal of Computer Vision, 2022. 1

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017. 2

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

[26] Junfeng Wu, Yi Jiang, Wenqing Zhang, Xiang Bai, and Song Bai. Seqformer: a frustratingly simple model for video instance segmentation. arXiv preprint arXiv:2112.08275, 2021. 1

[27] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In defense of online models for video instance segmentation. arXiv preprint arXiv:2207.10661, 2022. 1, 2, 3, 4

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

[29] 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

[30] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159, 2020. 2, 3