Mask-Free Video Instance Segmentation (Supplemental material)

Lei Ke[1,2] Martin Danelljan[1] Henghui Ding[1] Yu-Wing Tai[2] Chi-Keung Tang[2] Fisher Yu[1]
[1]ETH Zürich [2]HKUST

In this supplementary material, we first conduct additional experiment analysis of our Temporal KNN-patch Loss (TK-Loss) in Section 1. Then, we present visualization of temporal matching correspondence and compute its approximate accuracy in Section 2. We further show more qualitative VIS results analysis (including failure cases) in Section 3. Finally, we provide MaskFreeVIS algorithm pseudocode and more implementation details in Section 4. Please refer to our project page for extensive MaskFreeVIS video results.

1. Supplementary Experiments

Patch *vs*. Pixel in TK-Loss Extending Table 4 in the paper, in Table 1, we further compare the results of image patch *vs*. single pixels under different max K values during temporal matching. The one-to-K correspondence produces gains in both pixel and patch matching manners, while the improvement on patch matching is much more obvious.

Table 1. Patch *vs*. Pixel in one-to-**K** patch correspondence on YouTube-VIS 2019.

K	Pixel	Patch	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
1	✓		39.1	64.8	41.7	39.8	47.8
1		✓	40.8	65.8	44.1	40.3	48.9
3	✓		39.8	65.9	40.6	39.6	48.2
3		✓	41.9	66.9	45.1	**41.9**	50.3
5	✓		40.1	65.2	42.2	40.0	48.2
5		✓	**42.5**	66.8	**45.7**	41.2	**51.2**
7	✓		39.6	64.9	41.0	39.8	48.5
7		✓	42.3	**67.1**	44.6	40.6	50.7

Influence of Tube Length During model training, we sample a temporal tube from the video. We study the influence of the sampled tube lengths in Table 2, and observe that the performing of MaskFreeVIS saturates at temporal tube length 5. For even longer temporal tube, different from [5], the temporal correlation between the beginning frame and ending frame (two temporally most distant frame) is weak to find sufficient patch correspondence.

Additional Results on Various Amount of YTVIS Data For experiments in Figure 6 of the paper, we sample different portions (in percents) of YTVIS data by uniformly sampling frames per video. In Figure 1, we experiment with another video sampling strategy by directly sampling different numbers of videos from the YTVIS training set. Our MaskFreeVIS consistently attains large improvements (over 3.5 mask AP) over the baseline in both the COCO mask and box pretraining settings, with performance on par with the oracle Mask2Former in data-efficient settings.

Table 2. Results of varying **Tube Length** during training for TK-Loss on YouTube-VIS 2019. Tube length 1 denotes model training with **only** spatial losses in BoxInst [7].

Tube Length	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
1	38.3	65.4	38.5	38.0	47.4
3	42.1	66.4	44.9	41.0	50.8
5	**42.5**	66.8	**45.7**	41.2	**51.2**
7	42.5	**67.5**	45.2	**41.3**	51.1

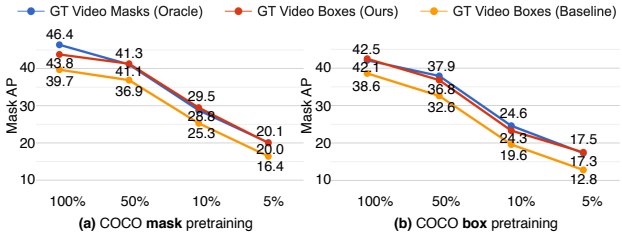

Figure 1. Results on YTVIS 2019 val with various percentages of the YTVIS training data, by *directly sampling different numbers of videos* from the YTVIS training set. Baseline denotes Mask2Former [1] trained with GT video boxes using BoxInst [7], while Oracle denotes the fully supervised Mask2Former trained with GT video masks.

Image-based Pretraining Results on COCO In Table 3, we report the performance on COCO of image-pretrained Mask2Former networks used as initial weights for our approach. The mask-free version employs the spatial losses of BoxInst [7]. We also show the corresponding VIS results on YTVIS 2019 by taking these image-pretrained models as initialization for our approach. Compared to the fully-supervised Mask2Former on COCO, the box-training process eliminates the image masks usage and obtaining a lower performance (over 10.0 AP) in image mask AP on

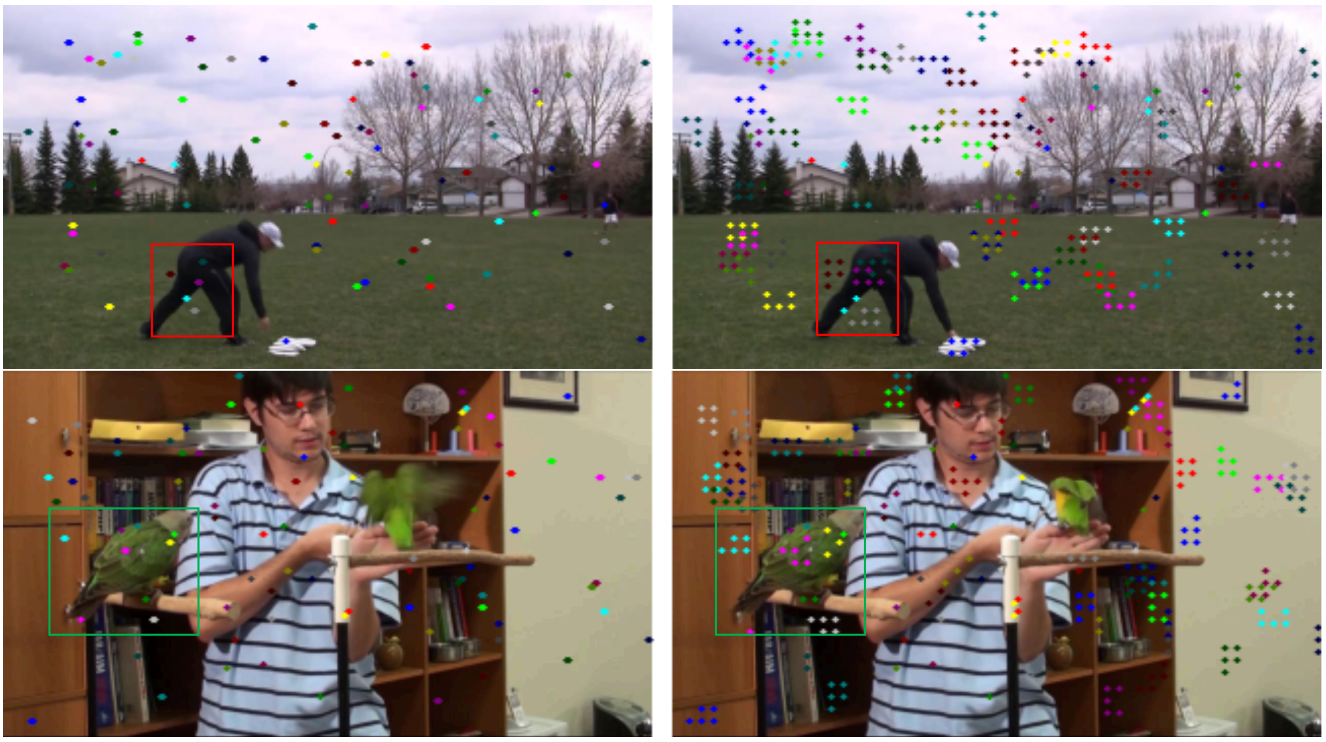

Figure 2. Visualization of the temporal correspondence in TK-Loss. We randomly sample 100 patch center points from Frame t, and draw its temporally matched patch center points in Frame t+1. Matches are shown in the same color, and should have consistent instance mask label. Taking patch center points near the left leg of the man (inside the red box, 1st row in Frame t) as an example, the matches in Frame t+1 consistently belong to the same foreground (leg) / background (grass) region. Best viewed in color.

COCO. However, even initialized from this low-performing image-pretrained models, our MaskFreeVIS using the proposed TK-Loss still greatly reduces the gap between fully-supervised and weakly-supervised VIS models as shown in the rightmost column of the Table 3.

Table 3. Results of image-based pretrained Mask2Former (M2F) [2] on COCO *val* and the corresponding video results on YTVIS 2019 by taking the image-pretrained one as initialization. M2F + BoxInst is mask-free, which is used to initialize MaskFreeVIS, while image-based M2F (Oracle) is to initialize video-based M2F (Oracle). Oracle denotes training with GT image or video masks.

Backbone	Image Method	Image AP	VIS Method	Video AP
R50	M2F + BoxInst	32.6	MaskFreeVIS	42.5
R50	M2F (Oracle)	43.7	M2F (Oracle)	46.4
R101	M2F + BoxInst	34.5	MaskFreeVIS	45.8
R101	M2F (Oracle)	44.2	M2F (Oracle)	49.2
SwinL	M2F + BoxInst	40.3	MaskFreeVIS	54.3
SwinL	M2F (Oracle)	50.1	M2F (Oracle)	60.4

Fully Mask-free Results on OVIS Extending from Table 12 of the paper, we further present the results of MaskFreeVIS on OVIS using COCO box pretraining as initialization in Table 4. Our MaskFreeVIS consistently improves the baseline from 10.3 to 13.5 mask AP without using any masks.

Table 4. Full results of our MaskFreeVIS on OVIS [6] using R50. I: using COCO mask pretrained model as initialization. V: using YTVIS video masks during training.

Method	Mask ann.	AP	AP_{50}	AP_{75}	AR_1	AR_{10}
Fully-supervised:						
VMT [4]	I+V	16.9	36.4	13.7	10.4	22.7
VITA [3]	I+V	19.6	41.2	17.4	11.7	26.0
Video Mask-free:						
VITA [3] + BoxInst [7]	I	12.1	28.3	10.2	8.8	17.9
VITA [3] + **MaskFreeVIS**	I	**15.7**$_{\uparrow 3.6}$	**35.1**	**13.1**	**10.1**	**20.4**
Mask-free:						
VITA [3] + BoxInst [7]	-	10.3	27.2	8.4	7.3	16.2
VITA [3] + **MaskFreeVIS**	-	**13.5**$_{\uparrow 3.2}$	**32.7**	**10.6**	**8.8**	**18.5**

2. More analysis on Temporal Correspondence

Visualization on Temporal Correspondence We visualize the dense temporal correspondence matching for TK-Loss computation in Figure 2. For better visualization, we randomly sample 100 patch center points from Frame t, and plots their respective patch correspondences in Frame t+1

Figure 3. Qualitative video instance segmentation results comparison between Mask2Former using Spatial Pairwise loss of BoxInst [7] (Baseline), our proposed TK-Loss (Ours), and Mask2Former (Oracle) trained with GT video and image masks.

using the same color. We observe robust one-to-K patch matching results, especially for the regions near the left leg of the man (inside the red box) and the white frisbee.

Correspondence Accuracy To further analyze the accuracy rate for the temporal correspondence, since there is no matching ground truth, we adopt the instance masks labels as an approximate measure. We randomly take 10% of the videos from the YTVIS 2019 train set, and split them to 5-frame tube. Following the cyclic connection manner, we compute whether two matched patch center points belonging to the same instance mask label. The average matching accuracy per image pair is 95.7%, where we observe the wrong matches are mainly due to the overlapping objects with similar local patch patterns.

3. More Qualitative Comparisons

In Figure 3, we provide more qualitative results comparison among Baseline (using spatial losses of BoxInst [7]), Ours (using the proposed TK-Loss), and Oracle Mask2Former (trained with GT video and image masks). Compared to the Baseline, the predicted masks by our approach is more temporally coherent and accurate, even out-

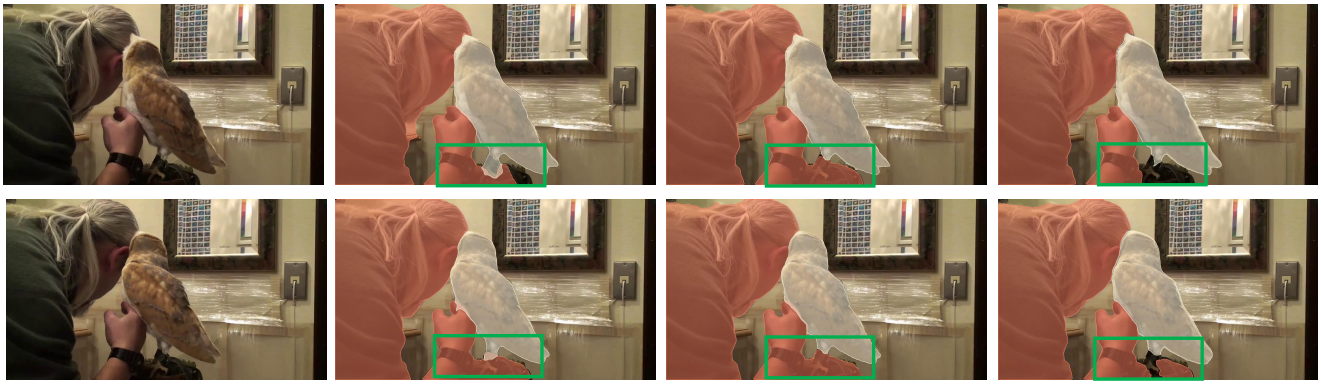

Figure 4. One typical failure case of our MaskFreeVIS. The neighboring hand watch and shelf belong to the same black color, and continuously closing to each other with no sufficient motion information for delineating these two objects.

performing the oracle results in some cases (such as the first row of Figure 3). We also identify one typical **failure case** of our MaskFreeVIS in Figure 4, where the neighboring hand watch and shelf are in almost the same black color, and continuously closing to each other with no sufficient motion information for distinction. We observe even the oracle model trained with GT video masks sometimes fail in correctly delineating these two objects (last row of Figure 4). Please refer to the attached video file on our project page for more qualitative results of our MaskFreeVIS.

4. More Implementation Details

Algorithm Pseudocode We outline the pseudocode for computing Temporal KNN-patch Loss in Algorithm 1, where the execution code does not exceed 15 lines. This further demonstrates the simplicity, beauty and lightness of our TK-Loss without any learnable model parameters.

More implementation details Before computing temporal image patch affinities, we first convert the input image from RGB color space to CIE Lab color space for better differentiating color differences. We set dilation rate to 3 when performing temporal patch searching. For the loss balance weights in Equation 7 and Equation 8 of the paper, we set λ_{pair} to 1.0 and λ_{temp} to 0.1. We follow the same training setting and schedule of the baseline methods when integrating our TK-Loss with Mask2Former [1], SeqFormer [8], VITA [3] and Unicorn [9] for video instance segmentation training. When performing mask-free pre-training on COCO with spatial losses of BoxInst, we keep the training details of the integrated method unchanged. When integrating with Mask2Former using ResNet-50 and batch size 16, the MaskFreeVIS training on YTVIS 2019 can be finished in around 4.0 hours with 8 Titan RTX. When jointly training with COCO labels, it needs around 2 days.

Algorithm 1 Temporal KNN-patch Loss.

Input: Tube length T, mask predictions M, frame width W, height H, radius R, patch distance threshold D.
Output: TK-Loss $\mathcal{L}_{\text{temp}}$

$\texttt{top}K$ denotes selecting top K patch candidates with the maximum patch similarities computed using L_2 distance $\texttt{Dis}(\cdot,\cdot)$
$L_{\texttt{cons}}$ denotes mask consistency loss (Equation 3 of the paper)
$\mathcal{L}_{\text{temp}} \leftarrow 0$.
for $t = 1, \ldots, T$ **do**
  $\hat{t} \leftarrow (t+1)\%T$
  $\mathcal{L}_f^{t\to\hat{t}} \leftarrow 0$.
  for $j = 1, \ldots, H \times W$ **do**
    # 1) Patch Candidate Extraction:
    $\mathcal{S}_{p_j}^{t\to\hat{t}} \leftarrow \{\hat{p}_i\}_i$, where $\|p_j - \hat{p}_i\| \le R$
    # 2) Temporal KNN-Matching:
    $\mathcal{S}_{p_j}^{t\to\hat{t}} \leftarrow \texttt{top}K(\mathcal{S}_{p_j}^{t\to\hat{t}})$, where $\texttt{Dis}(p_j, \hat{p}_i) \le D$
    # 3) Consistency Loss
    $\mathcal{L}_f^{t\to\hat{t}} \leftarrow \mathcal{L}_f^{t\to\hat{t}} + \sum_{\hat{p}_i \in \mathcal{S}_{p_j}^{t\to\hat{t}}} L_{\texttt{cons}}(M_{p_j}^t, M_{\hat{p}_i}^{\hat{t}})$
  end for
  # 4) Cyclic Connection
  $\mathcal{L}_{\text{temp}} \leftarrow \mathcal{L}_{\text{temp}} + \mathcal{L}_f^{t\to\hat{t}}/(H \times W)$
end for

References

[1] Bowen Cheng, Anwesa Choudhuri, Ishan Misra, Alexander Kirillov, Rohit Girdhar, and Alexander G Schwing. Mask2former for video instance segmentation. *arXiv preprint arXiv:2112.10764*, 2021. 1, 4

[2] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 2

[3] Miran Heo, Sukjun Hwang, Seoung Wug Oh, Joon-Young Lee, and Seon Joo Kim. Vita: Video instance segmentation

via object token association. In *NeurIPS*, 2022. 2, 4

[4] Lei Ke, Henghui Ding, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Video mask transfiner for high-quality video instance segmentation. In *ECCV*, 2022. 2

[5] 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

[6] 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. *IJCV*, 2021. 2

[7] Zhi Tian, Chunhua Shen, Xinlong Wang, and Hao Chen. Boxinst: High-performance instance segmentation with box annotations. In *CVPR*, 2021. 1, 2, 3

[8] Junfeng Wu, Yi Jiang, Wenqing Zhang, Xiang Bai, and Song Bai. Seqformer: a frustratingly simple model for video instance segmentation. In *ECCV*, 2022. 4

[9] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *ECCV*, 2022. 4