FlowVOS: Weakly-Supervised Visual Warping for Detail-Preserving and Temporally Consistent Single-Shot Video Object Segmentation

Supplementary Material

Julia Gong
jxgong@cs.stanford.edu
F. Christopher Holsinger
holsinger@stanford.edu
Serena Yeung
syyeung@stanford.edu

Stanford University
Stanford, California, USA

1 Additional Implementation Details

Training. We provide training details here as noted in Sec. 4.2 of our paper. Training and inference were both conducted on a TITAN RTX GPU. Following STM-cycle [6], we downsample the input resolution by half to 240×427 and upsample the output segmentation using nearest interpolation for efficiency. We train with a batch size of 4 videos, sample 3 frames per video, treat the first frame in temporal order as the annotated frame, and increase the maximum sampled temporal skip between frames every 5 epochs. We use data augmentation to train all models (see below for details). We set $\lambda = 1$ for both the Mask Flow Loss and segmentation network loss. To implement the Visual Flow Loss (Eq. 4), we first mask the previous frame, then warp for greater efficiency; this does not affect results. All three training losses are equally weighted. We tuned the teacher-forcing hyperparameter per validation set, with $p = 0.5$ for DAVIS17 and $p = 1$ for YouTubeVOS. We set hyperparameters $E_s = 240, E_a = 5$. We optimized all model parameters with the Adam optimizer [5] with learning rate $10^{-5}, \beta_1 = 0.9$, and $\beta_2 = 0.999$.

Data Augmentation. As mentioned above, we use data augmentation during model training. Augmentations are applied per video sequence; in a given batch, all sampled frames from the same video are augmented in the same way. Only geometric augmentations are applied to the corresponding ground-truth masks. Following STM-cycle [6] for fair comparison, our data augmentations include random affine transformations, scaling, horizontal flips, random added noise, and random contrast jittering. All augmented images are then

pre-processed via channel-wise normalization using the ImageNet [1] means and standard deviations. In inference, we only perform pre-processing, and not augmentation.

Training Epochs and Early Stopping. As discussed in Sec. 3.4 of our paper, we set the two-stage training hyperparameters to $E_s = 240, E_a = 5$. We train all models in all experiments for a total of 480 epochs, and we use the DAVIS17 [8] validation score as the stopping criterion for early stopping. Just as with reported results in the paper, we use the official DAVIS17 evaluation code [8] to calculate the stopping criterion validation scores. We tune the teacher-forcing hyperparameter per validation set. Due to YouTubeVOS [9] validation ground-truth only being available on the official challenge server, rather than using it as a stopping criterion, we still use the DAVIS17 validation set as the stopping criterion and evaluate the two highest-performing models on the YouTubeVOS validation evaluation server to select the best model. The final early-stopped model for DAVIS17 was trained for a total of 314 epochs; the final model for YouTubeVOS was trained for a total of 376 epochs.

2 End-to-End Segmentation Method Details

As noted in Sec. 3.3 of our paper, we now discuss additional details of the end-to-end segmentation method (in particular, contextualizing the segmentation network). To segment frame t, prior work STM's [4] query encoder on frame t regresses a key and value embedding. The key queries, via matrix multiplication, a memory bank of encoded previous frames and masks from time 1 to $t-1$; the decoder $\mathcal{D}$ uses the value embedding and memory read result to predict the final mask. STM-cycle [6] introduces cycle-consistency by sampling frames in temporal *and* reverse temporal order to reduce error propagation. While this approach improves performance, it still struggles with mask detail and temporal consistency. Therefore, our end-to-end method uses the flow module's output flow field to refine the final segmentation via concatenation in the decoder. Following optical flow works [2, 3], we use the Markov property that only the previous frame X_{t-1} and current frame X_t are needed to predict visual warping.

3 Analysis of Hyperparameter Robustness

As mentioned in Sec. 3.4 of our paper, we perform hyperparameter robustness analysis for the associated hyperparameters of the teacher-forcing and two-stage training mechanisms in Tables 1 and 2.

We analyze the effect of varying each of our main hyperparameters: (1) the teacher-forcing probability p, (2) the alternating freezing frequency E_a for two-stage training, and (3) weighting of the Mask Flow Loss, Visual Flow Loss, and segmentation loss. We conduct all of these experiments on the DAVIS17 [8] validation set using the standard Jaccard (IOU) mean $\mathcal{J}$ and combined $\mathcal{J}\&\mathcal{F}$ scores. We also analyze teacher-forcing on the YouTubeVOS [9] validation set using the standard seen and unseen $\mathcal{J}_S, \mathcal{J}_\mathcal{U}$ and global mean $\mathcal{G}$ scores. We only vary teacher-forcing for YouTubeVOS because the models for DAVIS17 and YouTubeVOS only differ in their teacher-forcing hyperparameter.*

*Among all of our submissions on the official YouTubeVOS evaluation server, only 2 submissions were used for model selection. All others were solely for the hyperparameter robustness analysis in Table 1(b).

Teacher-forcing probability. To analyze the sensitivity of our method to teacher-forcing, we vary p incrementally and keep the other hyperparameters constant, with $E_a = 5$ and equal weighting of all three losses. We show these results on DAVIS17 in Table 1(a) and on YouTubeVOS in Table 1(b).

First, we observe for both datasets that higher values of p tend to outperform lower values (and are less sensitive), which we hypothesize is because the flow module can learn accurate flow fields from warping ground-truth segmentations that can apply to the model's learned segmentations, while lower values of p may yield distorted flow fields when previous masks are not yet reasonable early on in training.

In Table 1(a), notice that the $\mathcal{J}\&\mathcal{F}$ score increases as p increases up until $p = 0.5$ as used in our work, after which performance has a slight decrease. This sensitivity is clearly stronger with p small than with p large, indicating the importance of having some degree of teacher-forcing. We intuitively see that with p close to 0, the flow module may encounter too much noise from the previous segmentation predictions (mostly early in training) to learn optimally. In Table 1(b), notice a similar trend where scores tend to increase with p; however, $p = 1$ has a slight advantage over other incrementally lower values, though not by a large margin. Especially with the small deltas among higher values of p, we believe some of the differences can be attributed to training variability. p also appears to impact the generalization to unseen classes more than those in seen classes, which we hypothesize stems from the same reason discussed above that higher values of p are less sensitive than lower values of p.

We also show results using annealed teacher-forcing, where p begins at 1 and decreases by 0.05 every 5 epochs until it plateaus at $p = 0$. We hypothesize that this does not do very well because the rate of decrease in p may not be in sync with the learning progress of the flow module; we leave investigating a learned modulating mechanism for p to future work.

With this discussion in mind, we underscore that teacher-forcing is only used during training (*not* inference), meaning that higher teacher-forcing values are training hyperparameters that do not impact the method at inference. Only the model's previous predicted masks are used in inference, and we find empirically that noisiness of predicted masks is not a big issue at this stage. Even in extreme cases like jump-cuts, predicted masks are less ideal, but still reasonable (see Figure 4).

Overall, we outperform the state-of-the-art STM-cycle [6] (whose $\mathcal{J} = 68.7, \mathcal{J}\&\mathcal{F} = 71.7$) across all values of p in Table 1(a), and we similarly do so for YouTubeVOS across reasonable values of p (compared to [6], whose $\mathcal{J}_\mathcal{S} = 71.7, \mathcal{J}_\mathcal{U} = 61.4, \mathcal{G} = 69.9$).

Alternating freezing frequency. To analyze the sensitivity of our method to the frequency of alternating freezing of the segmentation weights, we vary the freezing frequency E_a and keep the other hyperparameters constant, with $p = 0.5$ and equal weighting of all three losses. We show these results in Table 2(a).

Notice that the performance increases as E_a increases until 5, after which it decreases. We also show the two extremes of not freezing the segmentation weights at all ($E_a = 0$), and always freezing ($E_a = \infty$, the weights are never unfrozen). This trend shows that a reasonably small value of $E_a = 5$ as in our work balances allowing the flow module to learn from the segmentation weights independently, while also letting the segmentation model learn from the warped masks to refine the final prediction. Overall, the variation is not very large, suggesting that the model is not extremely sensitive to the freezing frequency.

Teacher-forcing probability p	$\mathcal{J}\%$	$\mathcal{J}\&\mathcal{F}\%$
0.0	69.4	72.0
0.25	69.8	72.2
0.5	**70.6**	**73.2**
0.75	69.8	72.7
1.0	70.2	73.0
anneal -0.05 every 5 epochs until $p = 0$	69.5	72.0

(a)
Varying teacher-forcing on the DAVIS17 validation set.

Teacher-forcing probability p	$\mathcal{J}_\mathcal{S}\%$	$\mathcal{J}_\mathcal{U}\%$	$\mathcal{G}\%$
0.0	71.1	59.6	68.7
0.25	71.5	62.0	69.8
0.5	**71.9**	62.7	70.5
0.75	71.4	63.6	70.9
1.0	71.7	**64.0**	**71.1**
anneal -0.05 every 5 epochs until $p = 0$	**71.9**	60.9	69.4

(b)
Varying teacher-forcing on the YouTubeVOS validation set.

Table 1: Analysis of the effect of teacher-forcing hyperparameter choice on our method. (a) shows varying of the teacher-forcing probability p on the DAVIS17 validation set while holding $E_a = 5$ and equal loss weighting constant. (b) shows the same for the YouTubeVOS validation set (all models were early-stopped using DAVIS17). In (a), $\mathcal{J}$ is the Jaccard (IOU) mean and $\mathcal{J}\&\mathcal{F}$ is the combined Jaccard and contour F-score; in (b), subscripts $\mathcal{S}$,$\mathcal{U}$ denote seen and unseen classes in training and $\mathcal{G}$ is the global mean.

Loss weighting. To analyze the effect of weighting the three losses used to train our method, we vary the loss weights while keeping the other hyperparameters constant, with $p = 0.5$ and $E_a = 5$. We show these results in Table 2(b). First, we write our combined loss function as

$$\mathcal{L} = \lambda_{MF}\mathcal{L}_{MF} + \lambda_{VF}\mathcal{L}_{VF} + \lambda_{seg}\mathcal{L}_{seg}, \tag{1}$$

where $\mathcal{L}_{MF}$ is the Mask Flow Loss (MFL), $\mathcal{L}_{VF}$ is the Visual Flow Loss (VFL), $\mathcal{L}_{seg}$ is the segmentation loss, and $\lambda_{MF}, \lambda_{VF}, \lambda_{seg}$ each denotes the weight for the corresponding subscripted loss. As noted in the paper, $\mathcal{L}_{MF}, \mathcal{L}_{seg}$ are each a combination of the cross-entropy and mask IOU losses. Following [6], which equally weights the cross-entropy and mask IOU components of the segmentation loss, we keep these weights equal (both 1) for $\mathcal{L}_{seg}$. However, we still vary the cross-entropy and mask IOU weights for $\mathcal{L}_{MF}$; we call these weights $\lambda_{MF,C}$ and $\lambda_{MF,I}$, respectively.

Overall, we find that the the optimal combination of weights is equally weighting all losses (first row). There does not appear to be extreme sensitivity to the different combinations of weights; however, it does appear that fully weighting the segmentation loss, as well as placing higher weights on the Visual Flow Loss and cross-entropy component of the Mask Flow Loss may have a slight advantage.

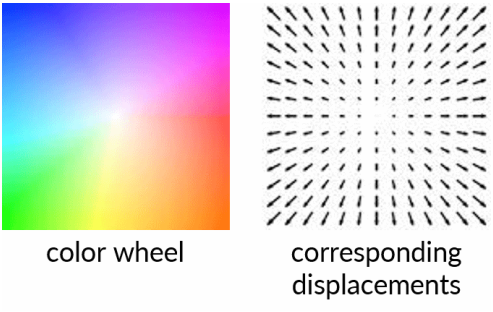

Figure 1: Color-coding scheme used to visualize flow fields in this paper. Pixel displacements colored on the left roughly correspond to the vectors in the illustration on the right. The center color (white) indicates no motion. We intensify the flow fields in our paper to better visualize the detailed displacements. Best viewed in color.

4 Additional Experiments

As mentioned in the paper, our method can be integrated into any state-of-the-art segmentation network. Thus, while we primarily integrate our method with STM-cycle [6] in this work, we further experiment with an additional backbone model, AGSS-VOS [7], to illustrate that our method complements other networks more generally. Since [7] already uses an optical flow module [3], we thus add our weakly-supervised losses to the training procedure. On DAVIS17 [8] validation, our method using [7] as the backbone model achieves 68.1% $\mathcal{J}\&\mathcal{F}$, 65.5% $\mathcal{J}$, and 70.7% $\mathcal{F}$, as compared to [7]'s 67.4% $\mathcal{J}\&\mathcal{F}$, 64.9% $\mathcal{J}$, and 69.9% $\mathcal{F}$. Note that we performed these experiments without any hyperparameter tuning.

5 Optical Flow Color-Coding

As discussed in Figure 4 of our work, following FlowNet 2.0 [3], we color-code our flow fields using polar coordinate displacements. The color-coding scheme is shown in Figure 1. See the caption for details.

6 Additional Qualitative Results

6.1 Qualitative Ablations: Visual Flow Loss

As mentioned in Sec. 3.2 of our paper, the Visual Flow Loss (VFL) in our foreground-targeted approach only penalizes differences between the *masked* warped previous frame and *masked* target frame, as opposed to their unmasked counterparts. Because of camera motion and other background activity, removing this masking can cause the model to optimize for consistency in the background, at the cost of not learning good flow fields for the foreground object of interest. We further illustrate this with a qualitative example in Figure 2. Without masking the VFL, we see that the flow field does not learn an object-targeted flow field, leading to a partially segmented object. This illustrates the importance of masking the VFL, as we do in our method.

(a) Masking (b) No masking

Figure 2: Qualitative ablation for masking the Visual Flow Loss (VFL). Our VFL only penalizes the differences between the *masked* warped previous frame and *masked* target frame. With masking in (a), our flow field accurately captures target object motion, whereas without masking in (b), we only partially capture object motion, which can result in object confusion or partial segmentations (second row). Best viewed in color.

We also note that this foreground masking means some regions of the flows far from the object are not as meaningful. However, this does not impact the quality of object warping, as seen in the warped frames (Figure 4, c3 in our paper and Figure 3, c3 here); the warped masks preserve details from the previous frame well and propagate them to refine a strong final segmentation (c).

6.2 Additional Qualitative Comparisons

Here, we show some additional qualitative results of our method. Figure 3 shows our model's improvements over STM-cycle [6] in segmentation detail, while Figures 5 and 6 show our model's improvements over [6] in temporal consistency.

In Figure 3, we extend Figure 4 in our paper and show additional examples of where our flow module's flow fields capture detailed motion and object boundaries in order to preserve object detail and consistency. In row 1, notice how the flow field in (c1) captures the movement of the hand, which enables a more complete segmentation in (c). The flow field in row 2 discretely captures the two dogs, which allows for propagating their masks separately and preventing detail artifacts that are present in (b). In row 3, despite an occlusion, our model warps both the object down-left (yellow) and occlusion up-right (red) to steadily shift the object position. In row 4, the overlapping objects are segmented properly due to the flow field that captures their fast motion. (c3) also shows the object detail preserved in the warping operation. In all cases, notice that our method preserves segmentation detail and consistency as compared to the state-of-the-art STM-cycle [6].

As noted in Sec. 4.5 of our paper, Figure 5 qualitatively compares our approach's tempo-

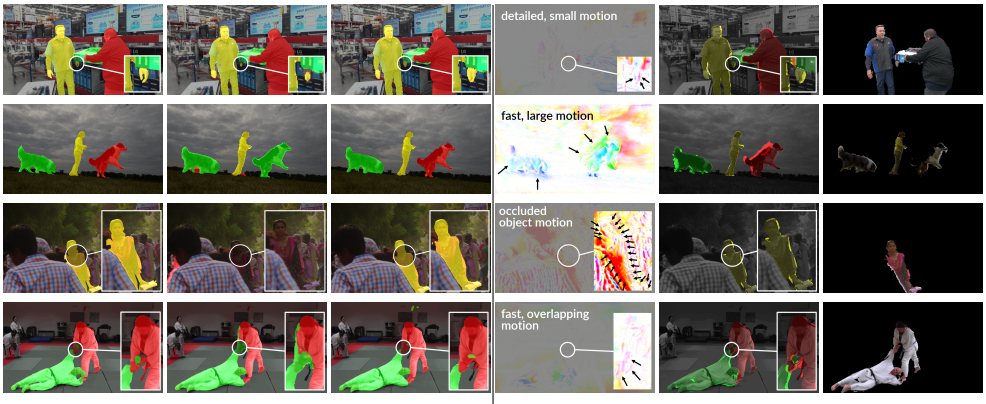

Figure 3: Additional qualitative comparisons with STM-cycle [6] on DAVIS17 validation (a, b, c), and our method's intermediate outputs (c1-3). Following [3], we color-code flow fields (c1) with polar coordinate displacements as detailed in Figure 1. (c2) brightens pixels that exist in the previous, but not the warped mask, highlighting motion that corresponds to the flows. (c3) shows that our warping operation accurately preserves object detail. In row 1, our flow field (c1) captures the detail in the small hand movement, preserving the structure of the hand. In row 2, our flow field captures the large movements of the dogs discretely, enabling in (c) the accurate tracking of the dogs as separate entities across time and preservation of detail as compared to (b). In row 3, despite an occlusion, our model warps both the object down-left (yellow) and occlusion up-right (red) to steadily shift the object position. In row 4, our flow field also captures the fast motions of the wrestlers as they overlap, allowing accurate segmentation of their boundaries. Best viewed in color.

ral consistency to that of STM-cycle [6] in videos from the DAVIS17 and YouTubeVOS validation sets. Owing to our visual warping for mask propagation, our method exhibits stronger temporal consistency for predicted object masks despite challenges of fast and abrupt object motion (video 1) and distractor objects with similar appearances (video 2).

In Figure 6, we extend Figure 5 and illustrate additional cases where our model exhibits object-level consistency (videos 1 and 2) and detail preservation and consistency (videos 3 and 4). Notice how in contrast to STM-cycle [6], our model can accurately track the correct boundaries of objects, such as the person's limbs in video 1 despite fast motion, and the whole panda body in video 2 despite camera motion blur and occlusions. Owing to our visual warping mechanism, we can also preserve stronger detail, as in the fins of fish in video 3 despite similarity between the fish and similarity to the background, and the full upper-left and right legs of the spider in video 4 throughout very small movements.

6.3 Error Cases and Analysis

We also examine some errors made by our method in Figure 4. In the first video, note that the last four frames are temporally consecutive in the dataset. These frames exhibit abrupt jump-cuts with drastic discontinuities in the visible parts of the object, as well as extreme motion blur. Even though visual warping can handle relatively large object motions, it does expect frames to be reasonably continuous—thus, it is difficult to produce felicitous warping

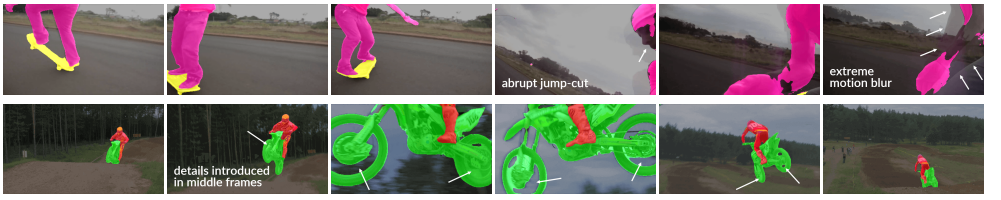

Figure 4: Error analysis of our method with examples of failure cases from YouTubeVOS validation (first video) and DAVIS17 validation (second video). In the first video, note that the last four frames are temporally consecutive in the dataset. They have abrupt jump-cuts that create discontinuities in object appearance, as well as extreme motion blur, both of which are difficult for flow fields to handle. In the second video, the reference mask in the first frame does not contain object details such as the wheel spokes; these details are challenging for our method to propagate through time because they were only visible in the middle frames. Best viewed in color.

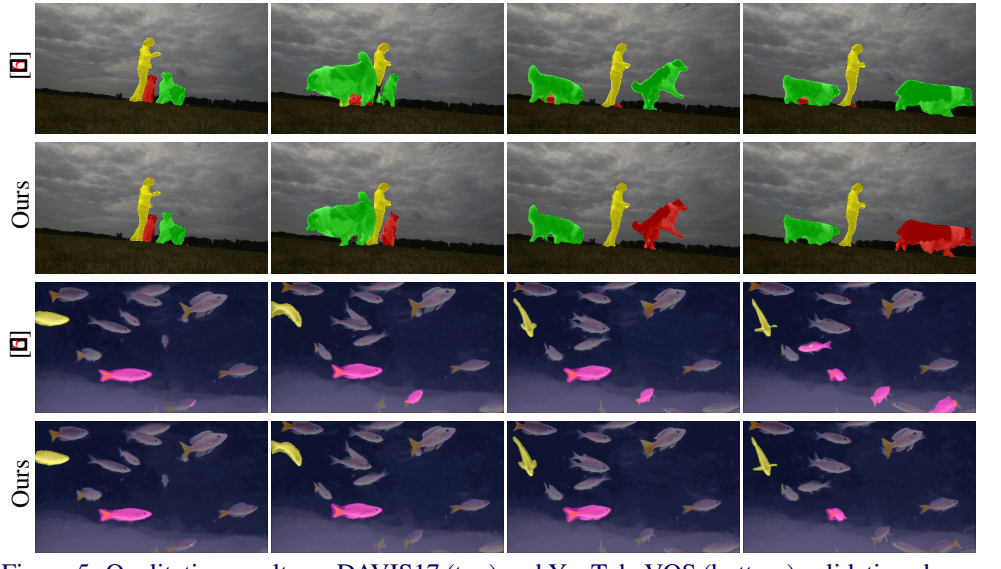

Figure 5: Qualitative results on DAVIS17 (top) and YouTubeVOS (bottom) validation showing our method's temporal consistency. We track objects and details well across time despite fast motion (top) and distractors (bottom), whereas state-of-the-art STM-cycle [6] has deteriorating results, confusing the dogs in the first video and tracking extra fish in the second. Best viewed in color.

due to these discontinuities. Despite the jump-cuts, our method still captures much of the object structure. In the final frame, we see an increase in motion blur that causes both the blur and the object to be equally transparent; this makes the object boundary ambiguous. This ambiguity in object motion is challenging for visual warping; as shown, our method fails to predict the middle portion of the object where this blur is most pronounced.

In the second video, the initial reference mask for the far-away object does not contain object details such as the bike's wheel spokes. These details are only introduced in the middle frames, once the object nears the camera. Since these details are only visible in the middle frames of the video and were not provided in the initial reference mask, they are difficult

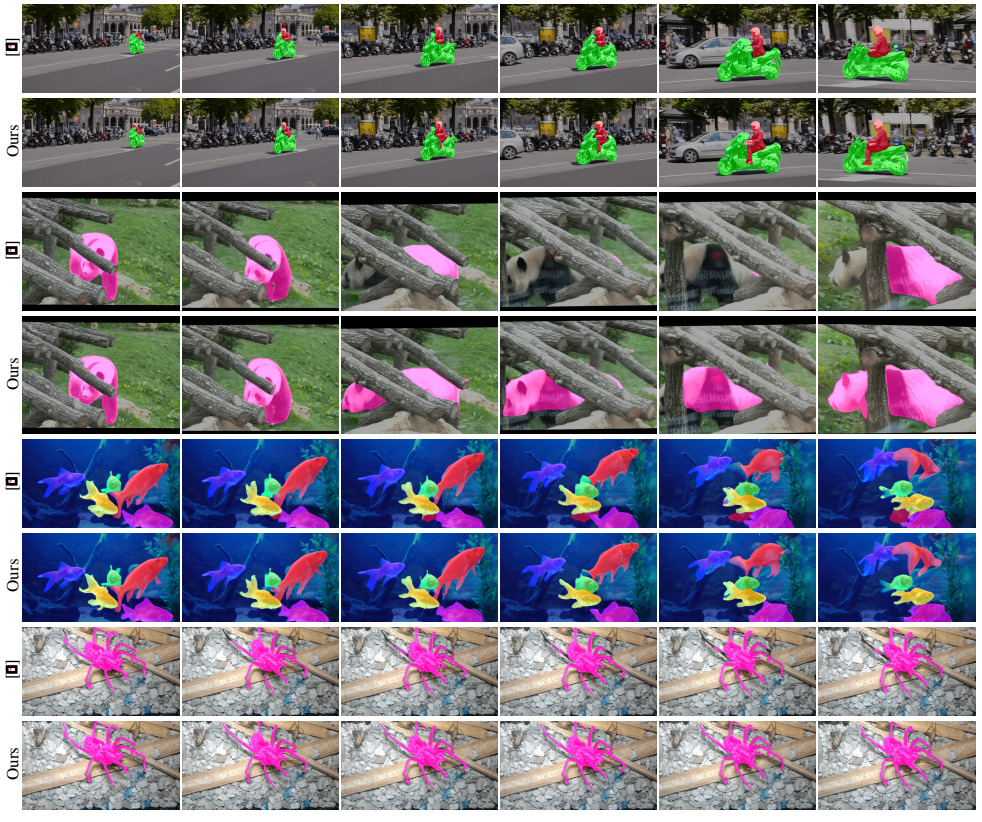

Figure 6: Additional qualitative results on DAVIS17 (videos 1 and 3) and YouTubeVOS (videos 2 and 4) validation showing our method's temporal consistency. Videos 1 and 2 show object-level consistency, while videos 3 and 4 show detail-level consistency. Our method tracks objects and details well across time despite several challenges: overlapping objects with fast motion (video 1), camera perspective changes, motion blur, and occlusions (video 2), interaction between detailed, but visually similar objects and the background (video 3), and detailed, but very small object articulations (video 4). In contrast, the state-of-the-art STM-cycle [6] has deteriorating results, such as the lost human leg in video 1, failure to identify the panda after occlusions and camera motion in video 2, missing fin details and confusion with the bottom fish in video 3, and the fragmented upper-left and right legs despite little movement in video 4. Best viewed in color.

for our flow module to propagate through time if the segmentation module does not capture them once they do appear. Our method thus struggles to capture these details, even though it still successfully propagates the overall object and larger details like the gaps in the wheels.

Alternating freezing frequency E_a	$\mathcal{J}\%$	$\mathcal{J}\&\mathcal{F}\%$
0^*	70.3	72.8
1	70.1	73.0
5	**70.6**	**73.2**
10	70.0	72.8
20	69.6	72.4
∞^*	70.2	72.6

(a)

Varying freezing frequency on the DAVIS17 validation set. ^{*0} indicates no freezing of segmentation weights at all, while ∞ indicates segmentation weights are always frozen.

MFL, VFL, segmentation loss weights

$\lambda_{MF,C}$	$\lambda_{MF,I}$	λ_{VF}	λ_{seg}	$\mathcal{J}\%$	$\mathcal{J}\&\mathcal{F}\%$
1.0	**1.0**	**1.0**	**1.0**	**70.6**	**73.2**
1.0	1.0	1.0	0.5	69.8	72.5
0.5	0.5	0.5	1.0	70.4	**73.2**
0.25	0.25	0.5	1.0	70.1	72.8
0.25	0.75	1.0	1.0	69.9	72.6
0.75	0.25	1.0	1.0	70.4	73.0
0.25	0.75	1.0	0.5	69.8	72.5
0.75	0.25	1.0	0.5	69.6	72.5
0.5	1.0	1.0	1.0	69.9	73.1
1.0	0.5	1.0	1.0	70.0	73.0
1.0	1.0	0.5	1.0	70.1	73.0
1.0	1.0	0.5	0.5	69.9	72.4

(b)

Varying Mask Flow Loss (MFL), Visual Flow Loss (VFL), and segmentation loss weights on the DAVIS17 validation set.

Table 2: Analysis of the effect of alternating freezing and loss weighting on our method. Experiments are conducted on DAVIS17. (a) shows alternating E_a while holding $p = 0.5$ and equal loss weighting constant. (b) shows varying the weights of the losses while keeping $p = 5, E_a = 5$ constant. $\mathcal{J}$ is the Jaccard (IOU) mean; $\mathcal{J}\&\mathcal{F}$ is the combined Jaccard and contour F-score.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[2] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[3] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017.

[4] Boyuan Jiang, MengMeng Wang, Weihao Gan, Wei Wu, and Junjie Yan. Stm: Spatiotemporal and motion encoding for action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[5] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *Proceedings of the International Conference on Learning Representations (ICLR)*, May 2015.

[6] Yuxi Li, Ning Xu, Jinlong Peng, John See, and Weiyao Lin. Delving into the cyclic mechanism in semi-supervised video object segmentation. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1218–1228. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/0d5bd023a3ee11c7abca5b42a93c4866-Paper.pdf.

[7] Huaijia Lin, Xiaojuan Qi, and Jiaya Jia. Agss-vos: Attention guided single-shot video object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[8] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017.

[9] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.