Additional Info: VideoMatch: Matching based Video Object Segmentation

Yuan-Ting Hu[1], Jia-Bin Huang[2], and Alexander G. Schwing[1]

[1]University of Illinois at Urbana-Champaign [2]Virginia Tech
{ythu2,aschwing}@illinois.edu jbhuang@vt.edu

In this additional document, we first present some more discussion about the proposed method. Subsequently, we show the visual results of our approach on DAVIS-16 [1], YouTube-Objects [2], JumpCut [3] and DAVIS-17 [4] datasets and then discuss the failure cases. In addition, we randomly choose some sequences from each dataset and show the visual results of our approach in videos along with this supplementary document. Please see the html file to browse the video results.

1 Discussion

The objective of our method during training is to learn how to *match* rather than learn how to *classify* [5–9, 8]. This explains why our method generalizes better on different datasets. For instance, the experiments on the YouTube-Objects dataset [2, 10] presented in Section 4.4 of the main paper show that our method outperforms the best baselines OnAVOS [8], where online adaptive fine-tuning is employed, despite the fact that our method is not fine-tuned on the groundtruth of the first frame nor trained on this dataset at all. Similar observation can be found on the JumpCut dataset [3]. In general, we found our approach without fine-tuning to be very competitive among baselines on the benchmark datasets, even comparing to methods requiring fine-tuning. In our experiments we found that there is a trade-off between generalization and adaptation. The performance of our fine-tuned network falls behind a computationally expensive and meticulously tuned state-of-the-art approach [7] on DAVIS-16 [1], indicating that there is less capacity in our network to adapt to the provided groundtruth after fine-tuning.

Memory consumption: For a video with resolution 480×854, our approach requires 8.5 GB of GPU memory to train with the original resolution and a batchsize of 1. However, it only requires 4.9 GB of GPU memory during testing. In contrast, the baseline OnAVOS [8] requires 10.7 GB GPU memory during online fine-tuning and testing. Hence, our matching based model reduces the memory on devices during online testing by a factor of 2.

2 Visual Results on the DAVIS-16 Dataset

We present the visual results of our approach on the DAVIS-16 [1] validation set. There are 20 video sequences in total in the validation set. We show the results of all sequences in Figure 1, Figure 2 and we uniformly sample six frames from every sequence. The performance in mIoU of our approach on every sequence is shown in parentheses.

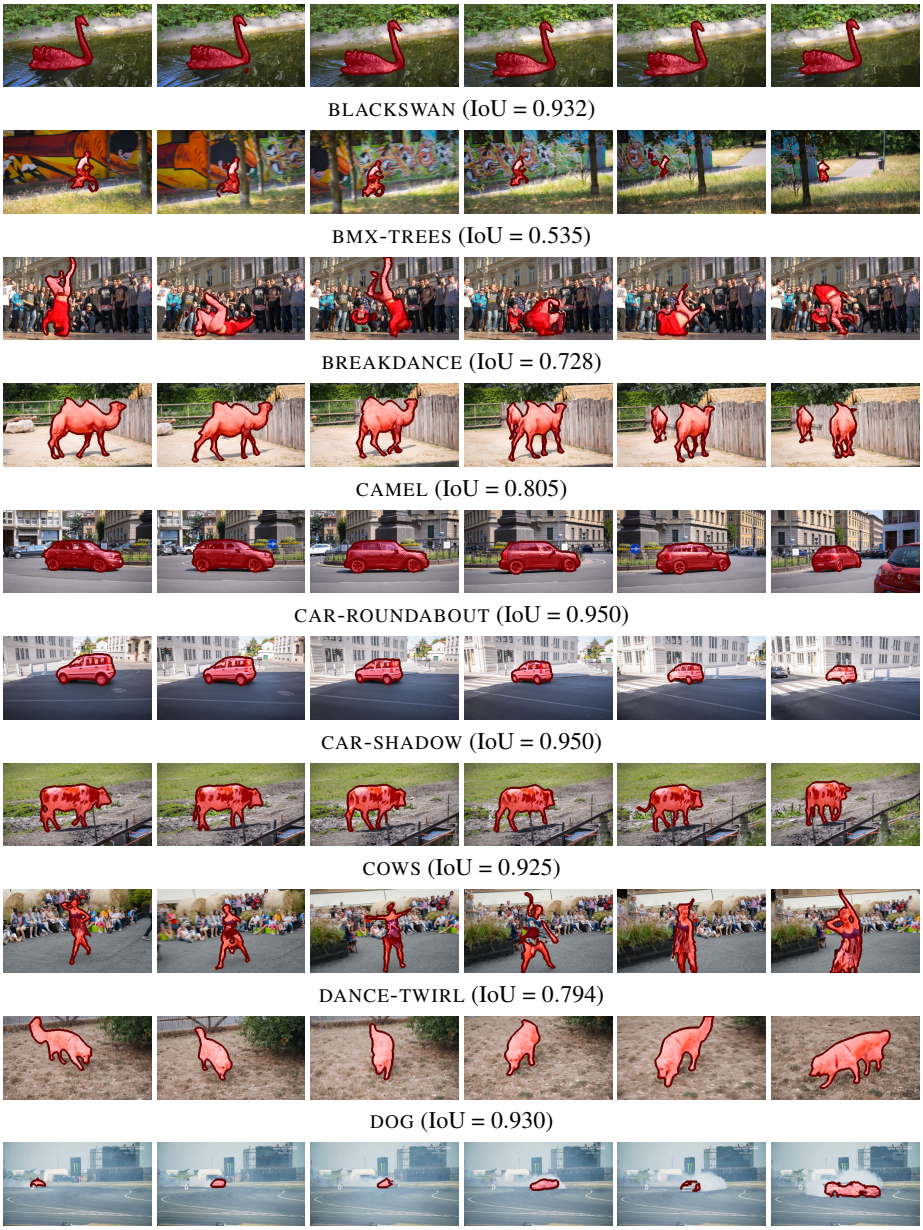

Fig. 1: Results of our approach on the DAVIS-16 dataset. The six frames are uniformly sampled from each video.

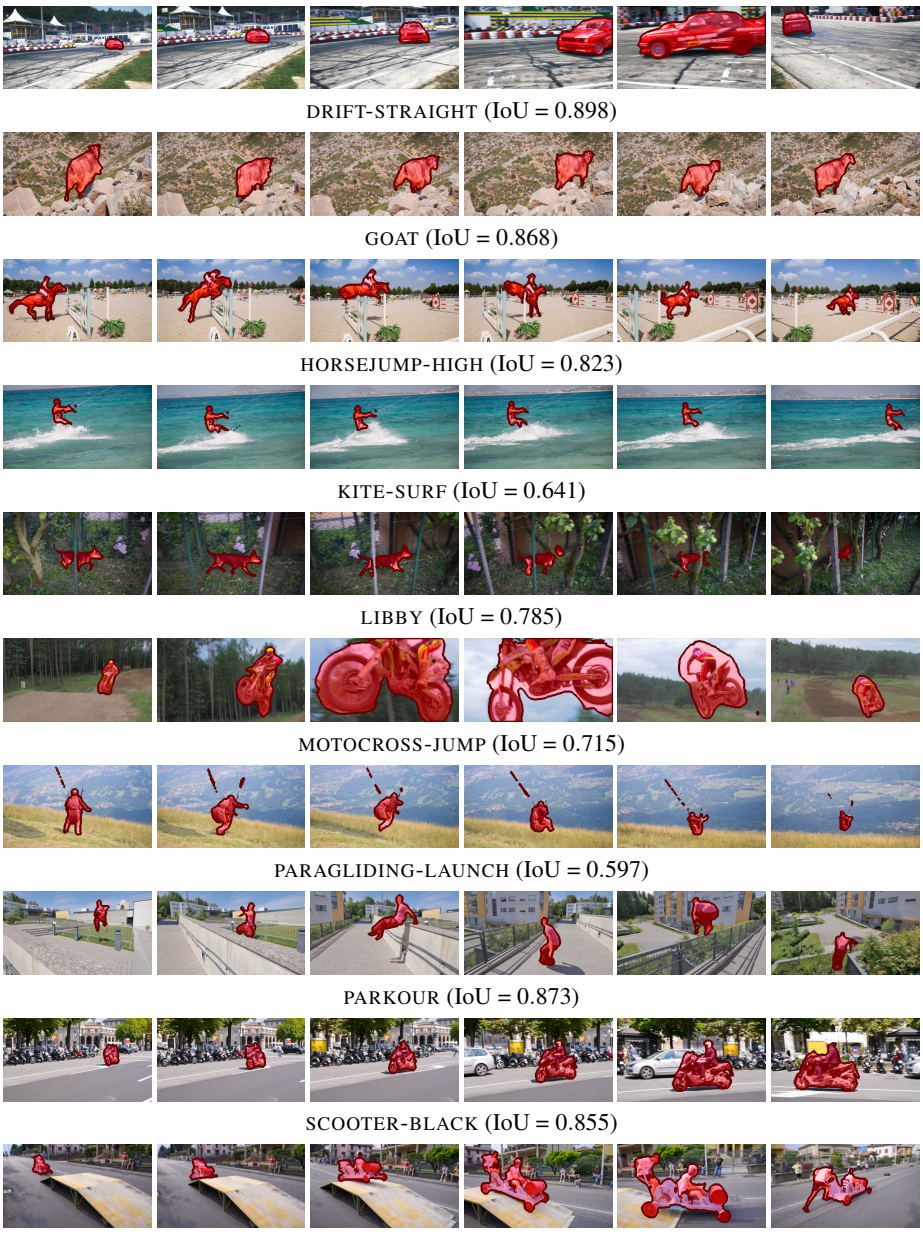

Fig. 2: Results of our approach on the DAVIS-16 dataset. The six frames are uniformly sampled from each video.

3 Visual Results on the YouTube-Objects Dataset

We present the visual results of our approach on the YouTube-Objects dataset [2, 10]. There are 126 video sequences in total in this dataset. We show the results of randomly sampled sequences in Figure 3, Figure 4, Figure 5. The six frames are uniformly sampled from every sequence. The performance of our approach in mIoU on each sequence is shown in parentheses.

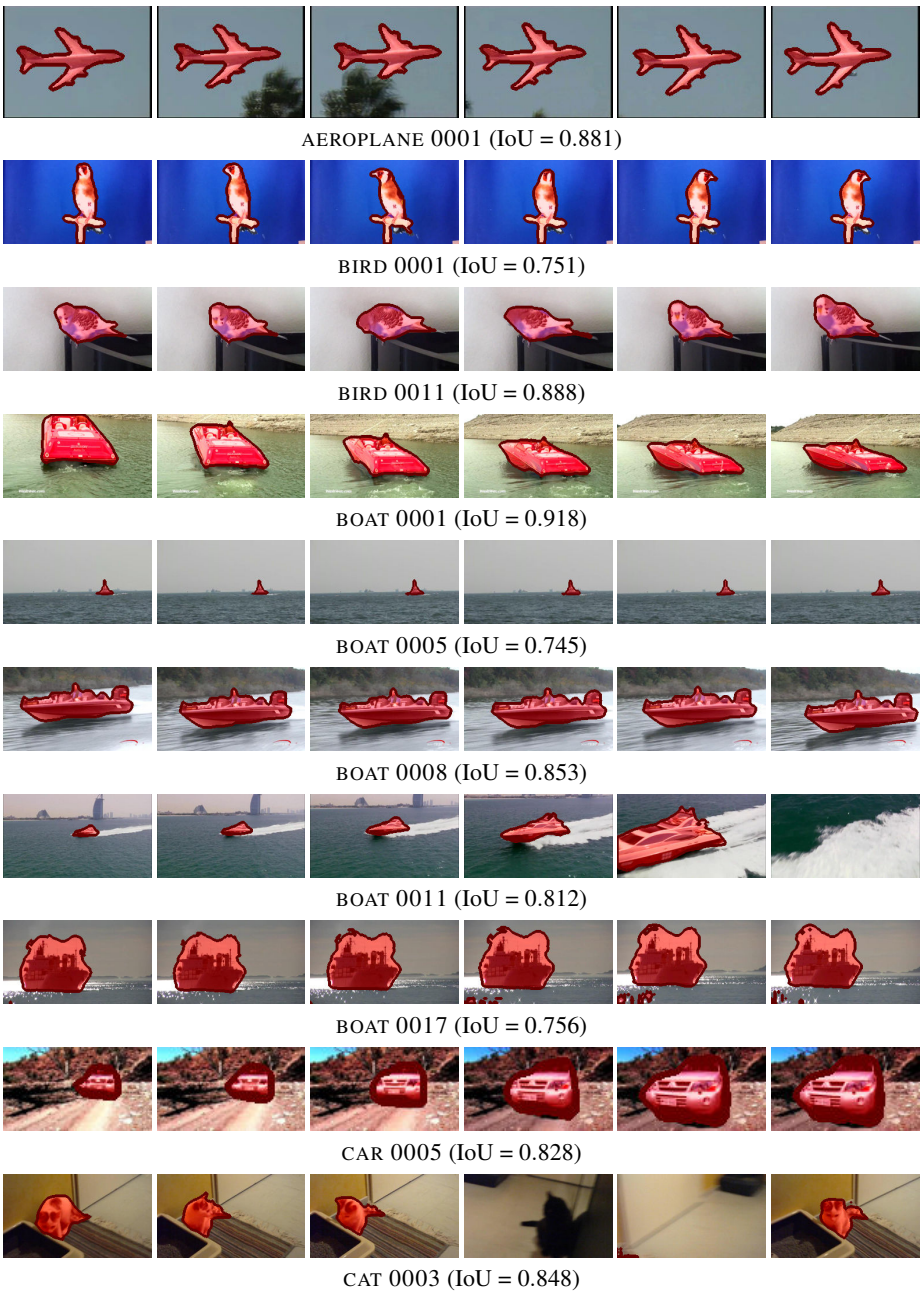

Fig. 3: Results of our approach on the YouTube-Objects dataset. The six frames are uniformly sampled from each video.

Fig. 4: Results of our approach on the YouTube-Objects dataset. The six frames are uniformly sampled from each video.

Fig. 5: Results of our approach on the YouTube-Objects dataset. The six frames are uniformly sampled from each video.

4 Visual Results on the JumpCut Dataset

We present the visual results of our approach on the JumpCut dataset [3]. There are 22 video sequences in total in this dataset. We show the results of all sequences in Figure 6, Figure 7, Figure 8. We transfer the groundtruth mask from the key frames $i \in \{0, 16, ..., 96\}$ to frames at $i + d$ and show the transferring results here. The transfer distance d is equal to 16. The error rate of our approach on each sequence is shown in parentheses.

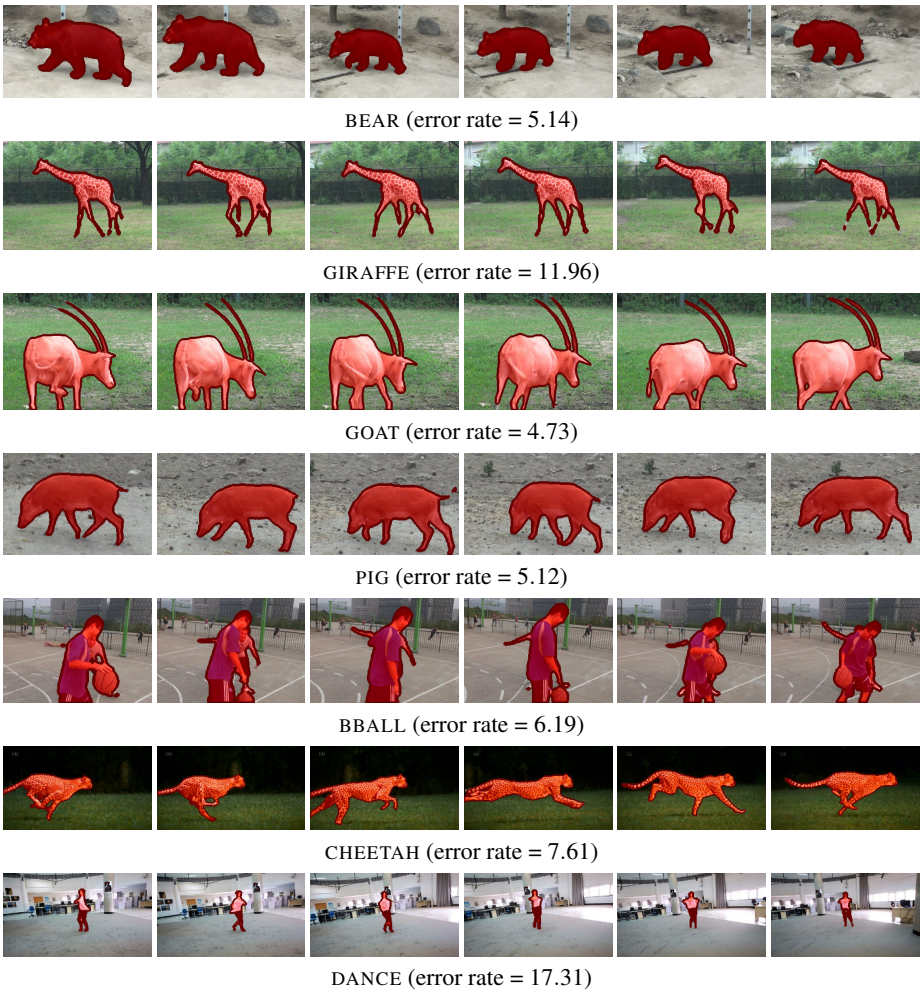

Fig. 6: Results of our approach on the JumpCut dataset. The six frames are uniformly sampled from each video.

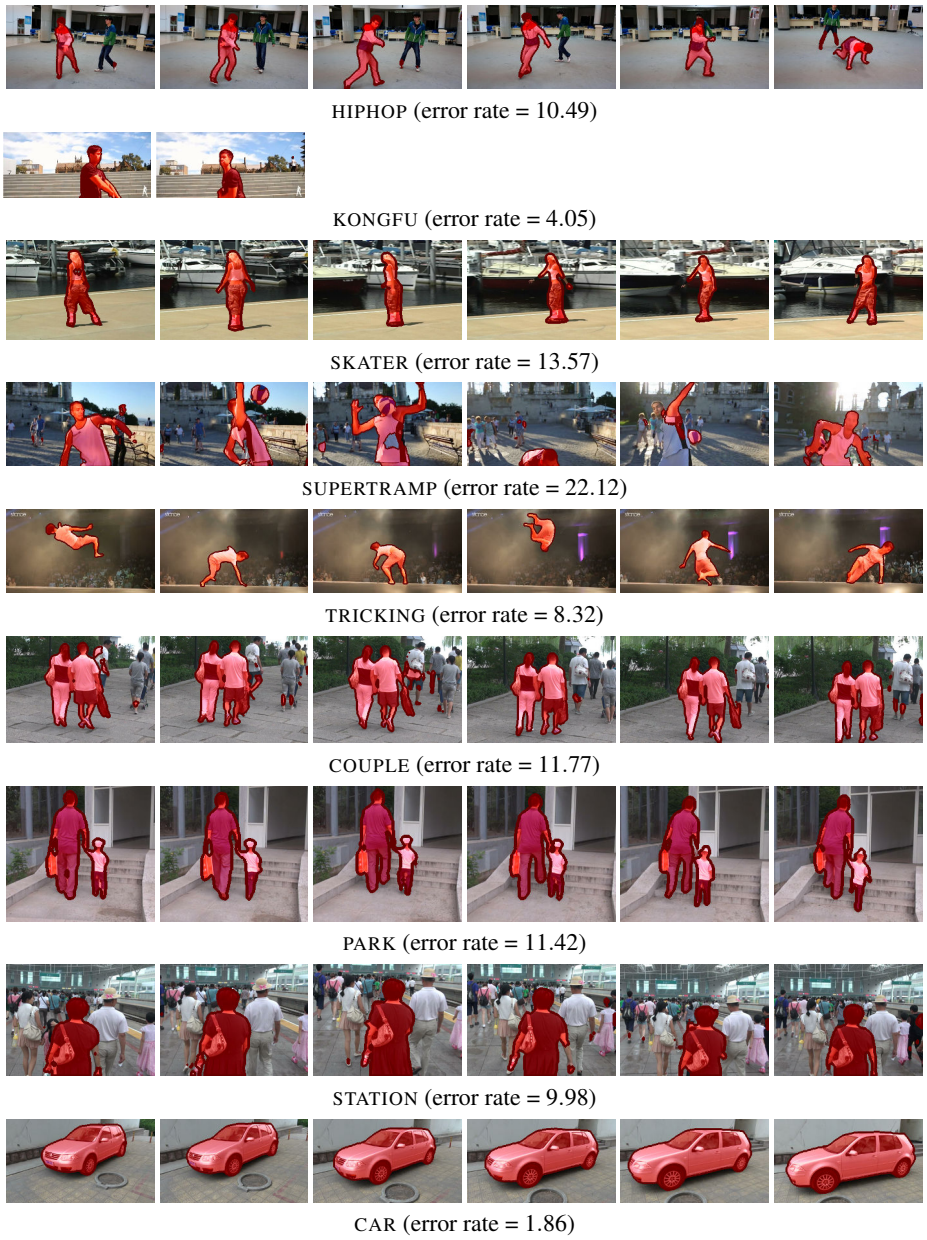

Fig. 7: Results of our approach on the JumpCut dataset. The six frames are uniformly sampled from each video.

Fig. 8: Results of our approach on the JumpCut dataset. The six frames are uniformly sampled from each video.

5 Visual Results on the DAVIS-17 Dataset

We present the visual results of our approach on the DAVIS-17 [4] validation set. There are 30 video sequences in total in the validation set. We show the results of all sequences in Figure 9, Figure 10, Figure 11 and we uniformly sample six frames from every sequence. We show the average mIoU of objects in each sequence in parentheses. Note the results here is the fine-tuned results (OURS-FT).

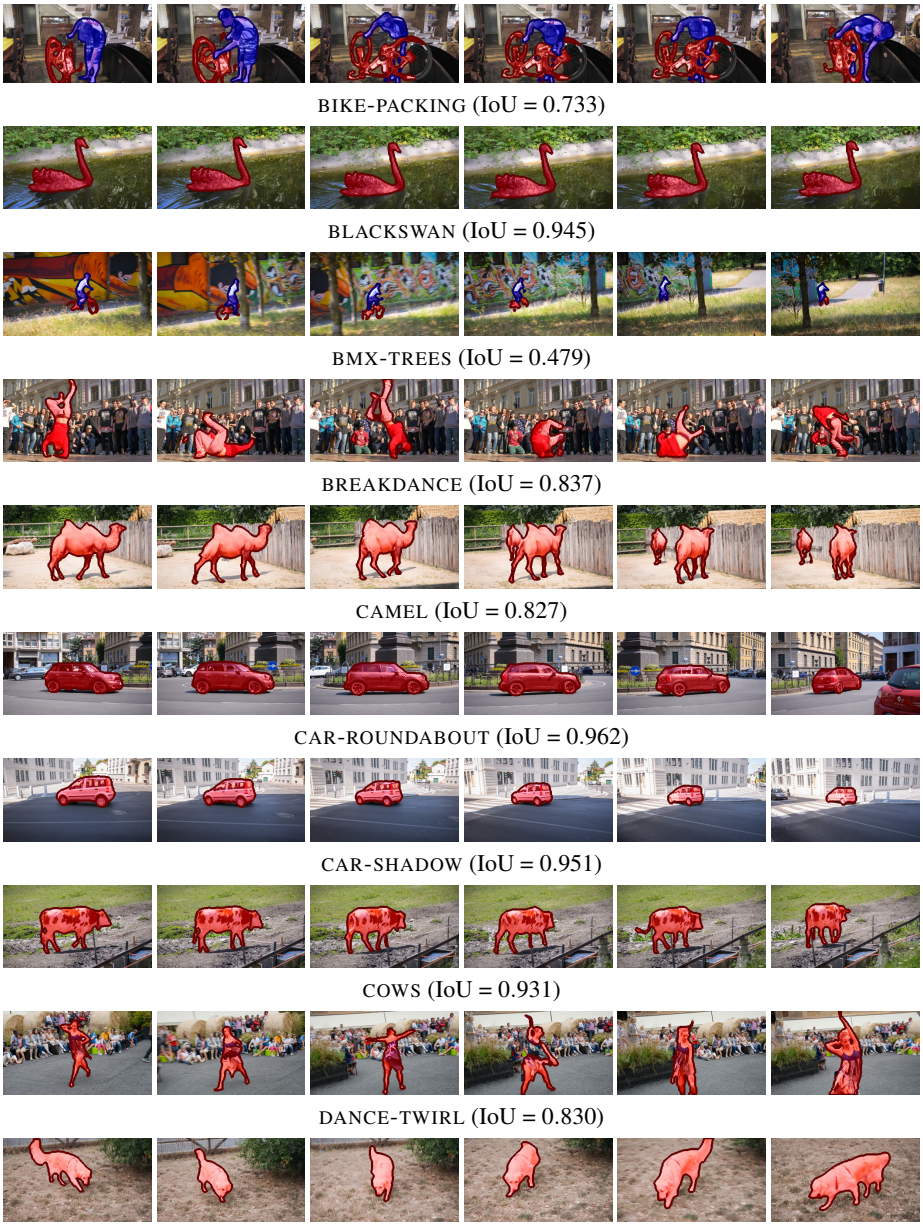

Fig. 9: Results of our approach on the DAVIS-17 dataset. The six frames are uniformly sampled from each video.

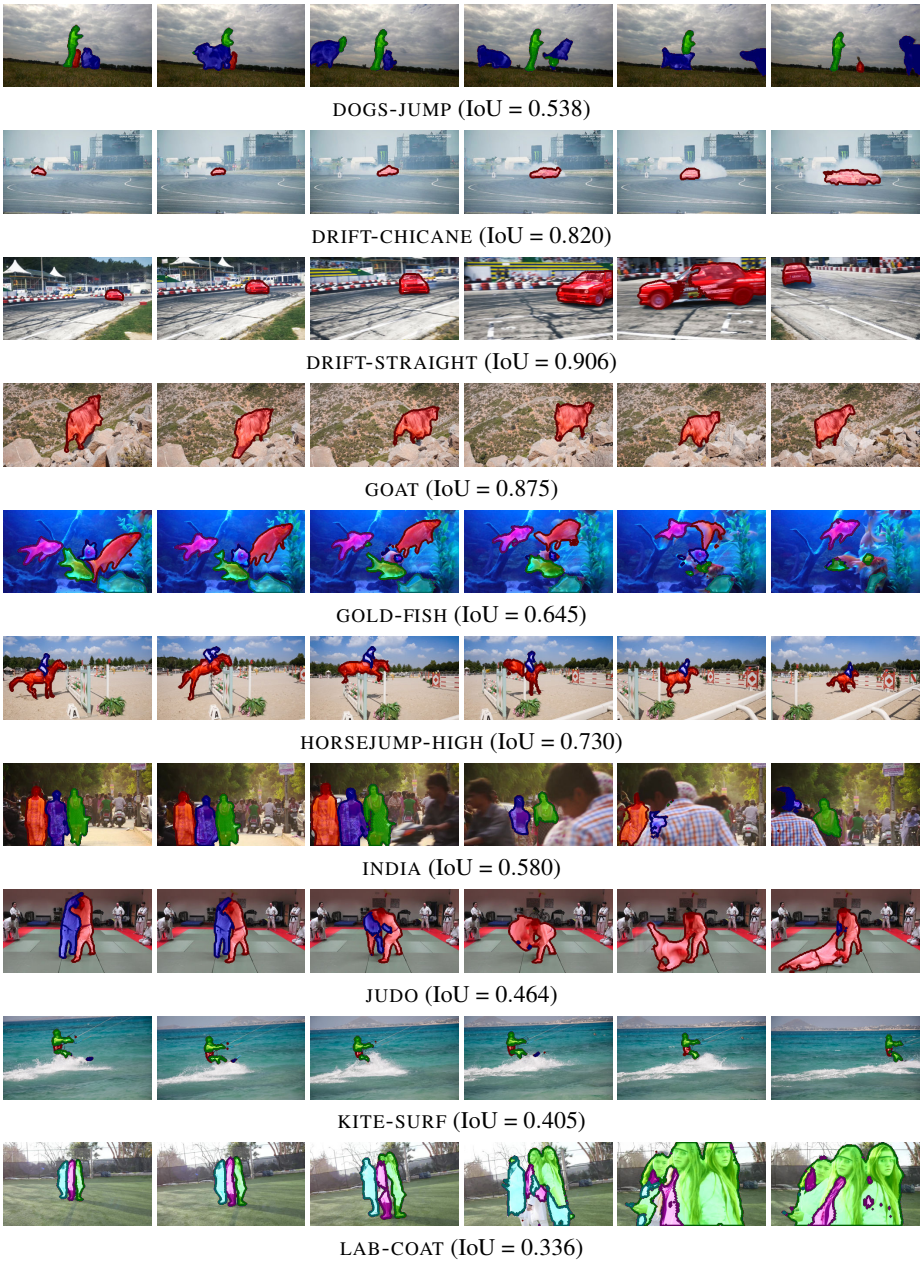

Fig. 10: Results of our approach on the DAVIS-17 dataset. The six frames are uniformly sampled from each video.

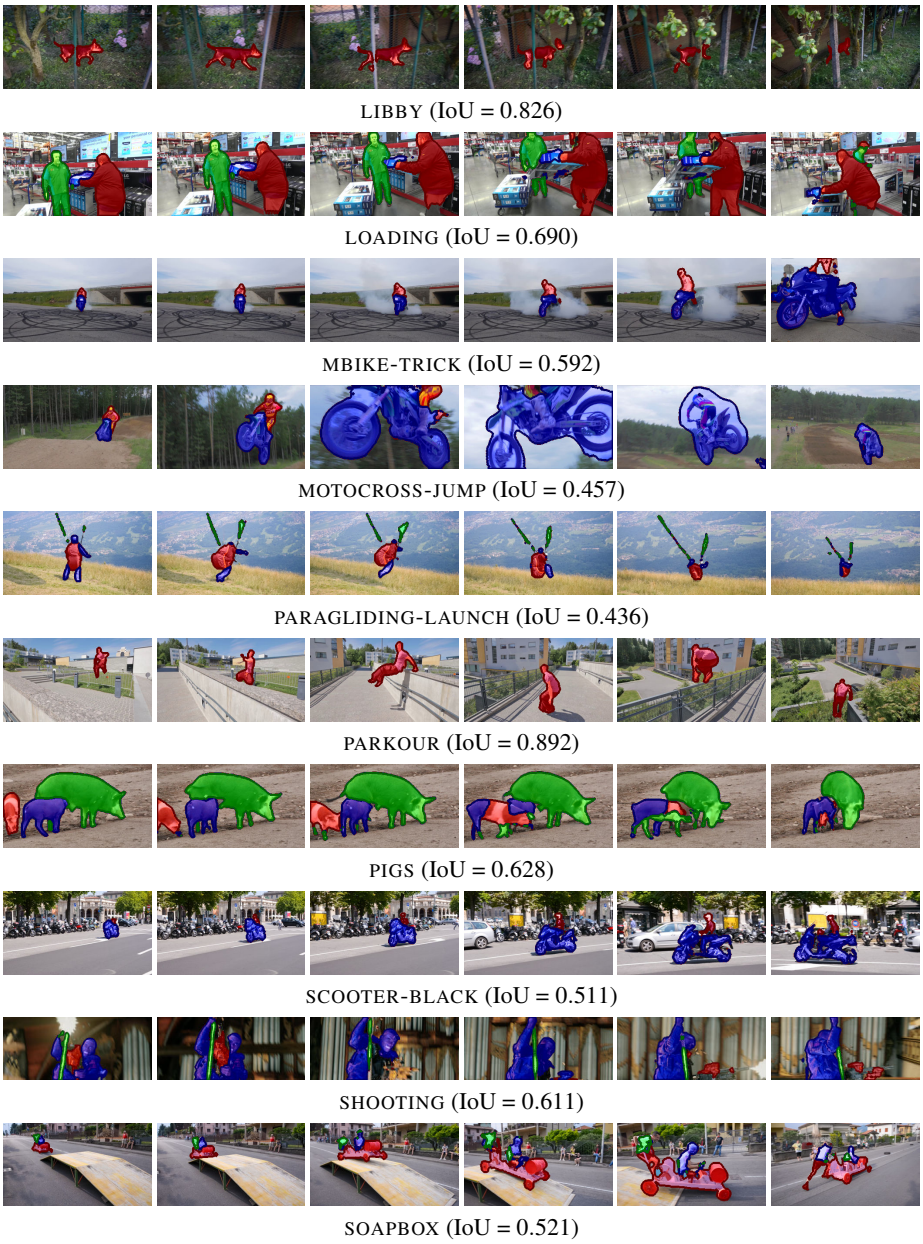

Fig. 11: Results of our approach on the DAVIS-17 dataset. The six frames are uniformly sampled from each video.

6 Failure Cases

We show the failure cases of our method in Figure 12. Possible reasons for our method to fail include similar appearance of different instances and tiny objects.

Fig. 12: Failure cases of our approach.

References

1. Perazzi, F., Pont-Tuset, J., McWilliams, B., Gool, L.V., Gross, M., Sorkine-Hornung, A.: A benchmark dataset and evaluation methodology for video object segmentation. In: Proc. CVPR. (2016)
2. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: Proc. CVPR. (2012)
3. Fan, Q., Zhong, F., Lischinski, D., Cohen-Or, D., Chen, B.: JumpCut: Non-successive mask transfer and interpolation for video cutout. ACM TOG (Proc. SIGGRAPH) (2015)
4. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017)
5. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: Proc. CVPR. (2017)
6. Khoreva, A., Perazzi, F., Benenson, R., Schiele, B., A.Sorkine-Hornung: Learning video object segmentation from static images. In: Proc. CVPR. (2017)
7. Khoreva, A., Benenson, R., Ilg, E., Brox, T., Schiele, B.: Lucid data dreaming for object tracking. arXiv preprint arXiv:1703.09554 (2017)
8. Voigtlaender, P., Leibe, B.: Online adaptation of convolutional neural networks for video object segmentation. BMVC (2017)
9. Hu, Y.T., Huang, J.B., Schwing, A.: MaskRNN: Instance level video object segmentation. In: NIPS. (2017)
10. Jain, S.D., Grauman, K.: Supervoxel-consistent foreground propagation in video. In: Proc. ECCV. (2014)