Learning Video Object Segmentation from Static Images
Supplementary material

[*]Federico Perazzi[1,2] [*]Anna Khoreva[3] Rodrigo Benenson[3]
Bernt Schiele[3] Alexander Sorkine-Hornung[1]

[1]Disney Research [2]ETH Zurich
[3]Max Planck Institute for Informatics, Saarbrücken, Germany

1. Content

In the supplementary material we provide additional quantitative and qualitative results. In details:

- Section 2 provides additional quantitative results for DAVIS, YoutubeObjects, and SegTrackv-2 (see Tables S1 - S4).
- Detailed attribute-based evaluation is reported in Section 3 and Table S5.
- The dataset specific tuning for additional ingredients is described in Section 4.
- Additional qualitative results with first frame box and segment supervision are presented in Section 5 and Figure S1.
- Examples of mask generation for the extra input channel are shown in Section 6 and Figure S2.
- Examples of optical flow magnitude images are presented in Section 7 and Figure S3.

2. Additional quantitative results

In this section we present additional quantitative results for three different datasets: DAVIS [8], YoutubeObjects [10], and SegTrackv-2 [5].

DAVIS Table S1 presents a more detailed evaluation on DAVIS using evaluation metrics proposed in [8]. Three measures are used: region similarity in terms of intersection over union (J), contour accuracy (F), and temporal instability of the masks (T). We outperform the competitive methods on all three measures.

We present the per-sequence comparison with other state-of-the-art methods on DAVIS in Table S2.

SegTrack-v2 Table S3 reports the per-sequence comparison with other state-of-the-art methods on SegTrack-v2.

YoutubeObjects The per-category comparison with other state-of-the-art methods on YoutubeObjects is shown in Table S4.

3. Attribute-based evaluation

Table S5 presents a more detailed evaluation on DAVIS using video attributes.

The attribute based evaluation shows that our generic model, `MaskTrack`, is robust to various video challenges present in DAVIS. It compares favourably on any subset of videos sharing the same attribute, except camera-shake, where `ObjFlow` [12] marginally outperforms our approach.

We observe that `MaskTrack` handles well fast-motion, appearance change and out-of-view, where competitive methods are failing [6, 12].

Furthermore, incorporating optical flow information and CRF post-processing into `MaskTrack` substantially increases robustness on all categories, reaching over 70% mIoU on each subcategory. In particular, `MaskTrack+Flow+CRF` better discriminates cases of low resolution, scale-variation and appearance change.

4. Dataset specific tuning

By adding additional ingredients specifically tuned for different datasets, such as optical flow and CRF post-processing, we can push the results even further, reaching 80.3 mIoU on DAVIS, 72.6 on YoutubeObjects and 70.3 on SegTrackv-2. In this section we discuss the dataset specific tuning.

Optical flow Although optical flow can provide interesting gains, we found it to be brittle when going across different datasets. Therefore we explored different strategies to handle optical flow.

[*]The first two authors contributed equally.

Method	DAVIS, mIoU						
	J			F			T
	Mean ↑	Recall ↑	Decay ↓	Mean ↑	Recall ↑	Decay ↓	Mean ↓
Box oracle	45.1	39.7	**-0.7**	21.4	6.7	**1.8**	**1.0**
Grabcut oracle	67.3	76.9	1.5	65.8	77.2	2.9	34.0
NLC [3]	64.1	73.1	8.6	59.3	65.8	8.6	35.8
FCP [9]	63.1	77.8	3.1	54.6	60.4	3.9	28.5
BVS [6]	66.5	76.4	26.0	65.6	77.4	23.6	31.6
ObjFlow [12]	71.1	80.0	22.7	67.9	78.0	24.0	22.1
MaskTrack	74.8	87.8	14.1	75.0	84.7	14.3	18.3
MaskTrack+Flow+CRF	**80.3**	**93.5**	8.9	**75.8**	**88.2**	9.5	18.3

Table S1: Comparison of segment tracking results on DAVIS. Our model improves over previous results.

Sequence	Method, mIoU			
	BVS [6]	ObjFlow [12]	TRS [13]	MaskTrack
bird of paradise	89.7	87.1	**90.0**	84.0
birdfall	65.3	52.9	**72.5**	56.6
bmx#1	67.1	**87.9**	86.1	81.9
bmx#2	3.2	4.0	**40.3**	0.1
cheetah#1	5.4	25.9	61.2	**69.3**
cheetah#2	9.2	37.2	**39.4**	17.4
drift#1	68.5	**77.9**	70.7	47.4
drift#2	32.7	27.4	70.7	**70.9**
frog	76.1	78.4	80.2	**85.3**
girl	86.5	84.2	86.4	**86.8**
hummingbird#1	53.2	**67.2**	53.0	39.0
hummingbird#2	28.7	68.5	**70.5**	49.6
monkey	85.7	87.8	83.1	**89.3**
monkeydog#1	40.5	47.1	**74.0**	25.3
monkeydog#2	17.1	21.0	**39.6**	31.7
parachute	93.7	93.3	**95.9**	93.7
penguin#1	81.6	80.4	53.2	**93.7**
penguin#2	82.0	83.5	72.9	**85.2**
penguin#3	78.5	83.9	74.4	**90.1**
penguin#4	76.4	86.2	57.2	**90.5**
penguin#5	47.8	**82.3**	63.5	78.4
penguin#6	84.3	87.3	65.7	**89.3**
soldier	55.3	**86.8**	76.3	82.0
worm	65.4	83.2	**82.4**	80.4
Mean	58.4	67.5	**69.1**	67.4

Table S3: Per-sequence results on the SegTrack-v2 dataset.

Category	Method, mIoU		
	BVS [6]	ObjFlow [12]	MaskTrack
aeroplane	80.8	**85.3**	81.6
bird	76.4	**83.1**	82.9
boat	60.1	70.6	**74.7**
car	56.7	**68.8**	66.9
cat	52.7	60.6	**69.6**
cow	64.8	71.5	**75.0**
dog	61.6	71.6	**75.2**
horse	53.1	62.3	**64.9**
motorbike	41.6	**59.9**	49.8
train	62.1	74.7	**77.7**
Mean per object	59.7	70.1	**71.7**
Mean per class	61.0	70.9	**71.9**

Table S4: Per-category results on the YoutubeObjects dataset.

Given a video sequence we compute the optical flow using EpicFlow [11] with Flow Fields matches [1] and convolutional boundaries [7]. In parallel to the MaskTrack with RGB images, we proceed to compute a second output mask using the magnitude of the optical flow field as input image (replicated into a three channel image). We then fuse by averaging the output scores given by the two parallel networks (using RGB image and optical flow magnitude as inputs).

For DAVIS we use the original MaskTrack model (trained with RGB images) as-is, without retraining. However, this strategy fails on YoutubeObjects and SegTrackv-2, mainly due to the failure modes of the optical flow algorithm and its sensitivity to the video data quality. To overcome this limitation we additionally trained the MaskTrack model using optical flow magnitude images on video data instead of RGB images. Training on optical flow magnitude images helps the network to be robust to the optical flow errors during the test time and provides a marginal improvement on YoutubeObjects and SegTrackv-2.

Overall integrating optical flow on top of MaskTrack provides $1\sim4\%$ on each dataset.

CRF post-processing As have been shown in [2] adding on top a well-tuned post-processing CRF [4] can gain a couple of mIoU points. Therefore following [2] we cross-validate the parameters of the fully connected CRF per each dataset based on the available first frame segment anno-

Sequence	Method, mIoU			
	BVS [6]	ObjFlow [12]	MaskTrack	MaskTrack+Flow+CRF
bear	**95.5**	94.6	92.8	93.1
blackswan	94.3	**94.7**	91.9	90.3
bmx-bumps	43.4	48.0	39.6	**57.1**
bmx-trees	38.2	14.9	32.1	**57.5**
boat	64.4	**80.8**	78.2	54.7
breakdance	50.0	49.6	59.4	**76.1**
breakdance-flare	72.7	76.5	**89.2**	77.6
bus	86.3	68.2	79.1	**89.0**
camel	66.9	**86.7**	80.4	80.1
car-roundabout	85.1	90.0	82.8	**96.0**
car-shadow	57.8	84.6	90.3	**93.5**
car-turn	84.4	87.6	**92.3**	88.6
cows	89.5	91.0	**91.9**	88.2
dance-jump	74.5	**80.4**	66.2	78.8
dance-twirl	49.2	56.7	67.8	**84.4**
dog	72.3	89.7	86.8	**90.8**
dog-agility	34.5	**86.0**	83.7	78.9
drift-chicane	3.3	17.5	0.5	**86.2**
drift-straight	40.2	31.4	46.0	**56.0**
drift-turn	29.9	3.5	**86.5**	86.0
elephant	84.9	87.9	**91.4**	87.2
flamingo	**88.1**	87.3	70.9	79.0
goat	66.1	**86.5**	85.8	84.5
hike	75.5	**93.4**	74.5	93.1
hockey	82.9	**85.0**	84.0	83.4
horsejump-high	80.1	**86.2**	78.4	81.8
horsejump-low	60.1	**82.2**	79.6	80.6
kite-surf	42.5	**70.2**	58.7	60.0
kite-walk	**87.0**	85.0	77.4	64.5
libby	77.6	59.4	**78.8**	77.5
lucia	90.1	89.7	88.4	**91.1**
mallard-fly	**60.6**	55.0	56.7	57.2
mallard-water	90.7	89.9	**91.0**	90.4
motocross-bumps	40.1	48.5	53.9	**59.9**
motocross-jump	34.1	59.4	**69.0**	68.3
motorbike	56.3	47.8	46.5	**56.7**
paragliding	87.5	94.7	93.2	**95.9**
paragliding-launch	**64.0**	63.7	58.9	62.1
parkour	75.6	**86.1**	85.3	88.2
rhino	78.2	89.5	**93.2**	91.1
rollerblade	58.8	**88.6**	33.0	78.7
scooter-black	33.7	76.5	64.9	**82.4**
scooter-gray	50.8	29.6	81.7	**82.9**
soapbox	78.9	68.9	86.1	**89.9**
soccerball	84.4	8.0	85.8	**89.0**
stroller	76.7	**87.7**	86.2	85.4
surf	49.2	**95.6**	92.6	92.8
swing	78.4	60.4	80.7	**81.9**
tennis	73.7	81.8	**87.3**	86.2
train	87.2	**91.7**	90.8	90.4
Mean	66.5	71.1	74.8	**80.3**

Table S2: Per-sequence results on the DAVIS dataset.

1st frame annotation Results with $\texttt{MaskTrack}_{Box}$ and MaskTrack, the frames are chosen equally distant based on the video sequence length

Figure S1: Qualitative results of $\texttt{MaskTrack}_{Box}$ and MaskTrack on Davis using 1st frame annotation supervision (box or segment). By propagating annotation from the 1st frame, either from segment or just bounding box annotations, our system generates results comparable to ground truth.

Attribute	Method, mIoU				
	BVS [6]	ObjFlow [12]	MaskTrack	MaskTrack+Flow	MaskTrack+Flow+CRF
Appearance change	0.46	0.54	**0.65**	0.75	**0.76**
Background clutter	0.63	0.68	**0.77**	0.78	**0.79**
Camera-shake	0.62	**0.72**	0.71	0.77	**0.78**
Deformation	0.7	0.77	**0.77**	0.78	**0.8**
Dynamic background	0.6	0.67	**0.69**	0.75	**0.76**
Edge ambiguity	0.58	0.65	**0.68**	**0.74**	**0.74**
Fast-motion	0.53	0.55	**0.66**	0.74	**0.75**
Heterogeneous object	0.63	0.66	**0.71**	0.77	**0.79**
Interacting objects	0.63	0.68	**0.74**	0.75	**0.77**
Low resolution	0.59	0.58	**0.6**	0.75	**0.77**
Motion blur	0.58	0.6	**0.66**	0.72	**0.74**
Occlusion	0.68	0.66	**0.74**	0.75	**0.77**
Out-of-view	0.43	0.53	**0.66**	**0.71**	**0.71**
Scale variation	0.49	0.56	**0.62**	0.72	**0.73**
Shape complexity	0.67	0.69	**0.71**	0.72	**0.75**

Table S5: Attribute based evaluation on DAVIS.

Figure S2: Examples of training mask generation. From one annotated image, multiple training masks are generated. The generated masks mimic plausible object shapes on the preceding frame.

Figure S3: Examples of optical flow magnitude images for different datasets.

tations of all video sequences. We employ coarse-to-fine search scheme for tuning CRF parameters and fix the number of mean field iterations to 10. We apply the CRF on a temporal window of 3 frames to improve the temporal stability of the results. The color (RGB) and the spatio-temporal (XYT) standard deviation of the *appearance kernel* are set, respectively, to 10 and 5. The pairwise term weight is set to 5. We employ an additional *smoothness kernel* to remove small isolated regions. Both its weight and the spatial (XY) standard deviation are set to 1.

5. Additional qualitative results

In this section we provide additional qualitative results for the $\texttt{MaskTrack}_{Box}$ and MaskTrack systems. Figure S1 shows the video object segmentation results when considering different types of annotations on DAVIS. Starting from segment annotations or even only from box annotations on the first frame, our model generates high quality segmentations, making the system suitable for diverse applications.

6. Examples of training mask generation

In the main paper we show that the main factor affecting the quality is using any form of mask deformations when creating the training samples (both for offline and online training). The mask deformation ingredient is crucial for our MaskTrack approach, making the segmentation estimation more robust at test time to the noise in the input mask. Figure S2 shows examples of generated masks using affine transformation as well as non-rigid deformations via thin-plate splines.

7. Examples of optical flow magnitude images

We propose to employ optical flow magnitude as a source of additional information to guide the segmentation. The flow magnitude roughly looks like a gray-scale object and captures useful object shape information, therefore complementing the MaskTrack model with RGB images as inputs. Examples of optical flow magnitude images are shown in Figure S3.

References

[1] C. Bailer, B. Taetz, and D. Stricker. Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *ICCV*, 2015. 2

[2] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv:1606.00915*, 2016. 2

[3] A. Faktor and M. Irani. Video segmentation by non-local consensus voting. In *BMVC*, 2014. 2

[4] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*. 2011. 2

[5] F. Li, T. Kim, A. Humayun, D. Tsai, and J. M. Rehg. Video segmentation by tracking many figure-ground segments. In *ICCV*, 2013. 1

[6] N. Maerki, F. Perazzi, O. Wang, and A. Sorkine-Hornung. Bilateral space video segmentation. In *CVPR*, 2016. 1, 2, 3, 5

[7] K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. V. Gool. Convolutional oriented boundaries. 2

[8] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 1

[9] F. Perazzi, O. Wang, M. Gross, and A. Sorkine-Hornung. Fully connected object proposals for video segmentation. In *ICCV*, 2015. 2

[10] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012. 1

[11] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015. 2

[12] Y.-H. Tsai, M.-H. Yang, and M. J. Black. Video segmentation via object flow. In *CVPR*, 2016. 1, 2, 3, 5

[13] F. Xiao and Y. J. Lee. Track and segment: An iterative unsupervised approach for video object proposals. In *CVPR*, 2016. 2