Video object segmentation

"One-shot Video Object Segmentation"

Caelles et.al. - CVPR 2017

"Online Adaptation of Convolutional Neural Networks for Video Object Segmentation"

Voigtlaender et. al. – BMVC 2017

Presented by Emanuela Haller

OSVOS – “One-Shot Video Object Segmentation”

Figure 1. Example result of our technique: The segmentation of the first frame (red) is used to learn the model of the specific object to track, which is segmented in the rest of the frames independently (green). One every 20 frames shown of 90 in total.

OSVOS

- Problem:
  - Semi-supervised video object segmentation

- Contributions:
  - Adapt a CNN to a particular object instance given a single annotated image
    - generic semantic information -> knowledge of the usual shapes of objects -> particular object segmentation
  - Temporal consistency, but not explicitly imposed
  - Speed accuracy trade-off
    - Fine-tuning level (181 ms – 71.5% -> 7.83 s – 79.7%)
    - Number of annotated frames (1 frame – 79.8% -> 4 frames – 86.9%)

OSVOS

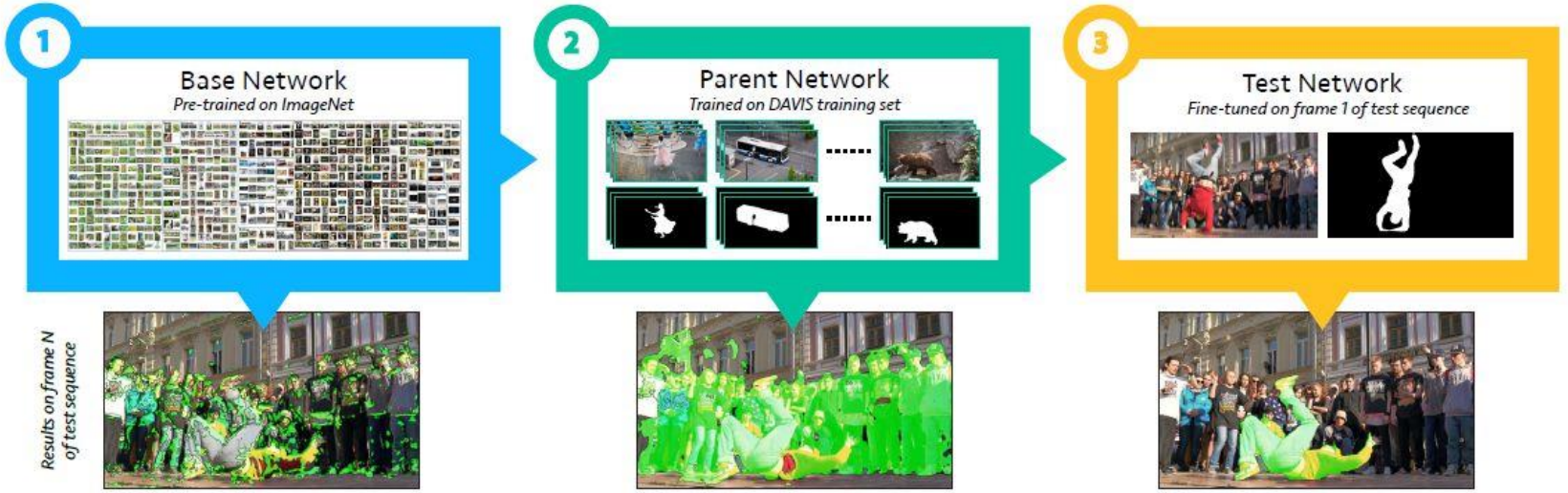


Figure 2. **Overview of OSVOS**: (1) We start with a pre-trained base CNN for image labeling on ImageNet; its results in terms of segmentation, although conform with some image features, are not useful. (2) We then train a *parent network* on the training set of DAVIS; the segmentation results improve but are not focused on an specific object yet. (3) By fine-tuning on a segmentation example for the specific target object in a single frame, the network rapidly focuses on that target.

- “It is an object” -> “It is this particular object”

CNN architecture

- Goals:
  - Accurately localized dense predictions
  - Relatively small number of parameters
  - Relatively fast at testing time

- CNN architecture used for biomedical image segmentation*
  - Based on VGG**
  - Fully convolutional

*Maninis et. al. “Deep retinal image understanding”. MICCAI 2016

**Simonyan et. al. “Very deep convolutional networks for large-scale image recognition”. ICLR 2015

CNN architecture

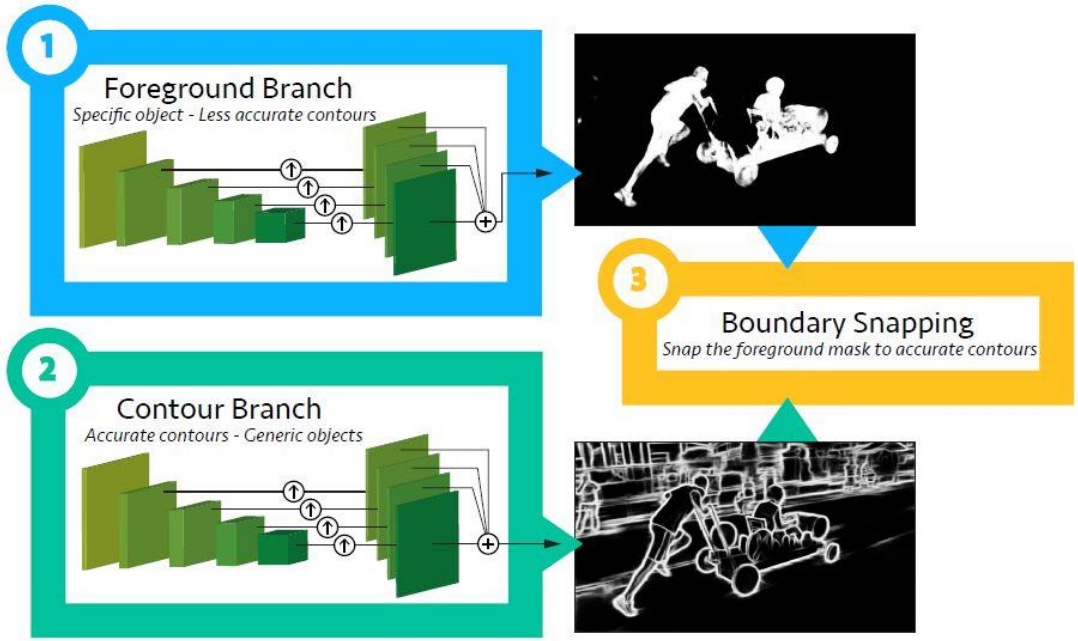


Figure 4. **Two-stream FCN architecture**: The main foreground branch (1) is complemented by a contour branch (2) which improves the localization of the boundaries (3).

CNN

- Binary classification
  - Foreground vs. background
- Cross-entropy loss

$$\mathcal{L}(\mathbf{W}) = -\sum_j y_j \log P(y_j=1|X;\mathbf{W}) + (1-y_j)\log(1-P(y_j=1|X;\mathbf{W}))$$
$$= -\sum_{j\in Y_+} \log P(y_j=1|X;\mathbf{W}) - \sum_{j\in Y_-} \log P(y_j=0|X;\mathbf{W})$$

- Imbalance between classes

$$\mathcal{L}_{mod} = -\beta \sum_{j\in Y_+} \log P(y_j=1|X) - (1-\beta) \sum_{j\in Y_-} \log P(y_j=0|X)$$

$$\beta = |Y_-|/|Y|$$

Training

- Base network – offline training
  - VGG trained on ImageNet for image labeling
- Parent network – offline training
  - Fully convolutional network further trained on DAVIS training set for object segmentation
- Test network – online training
  - Fine-tune the parent network
  - Segment a particular entity in a video given the image and the segmentation of the first frame

OSVOS

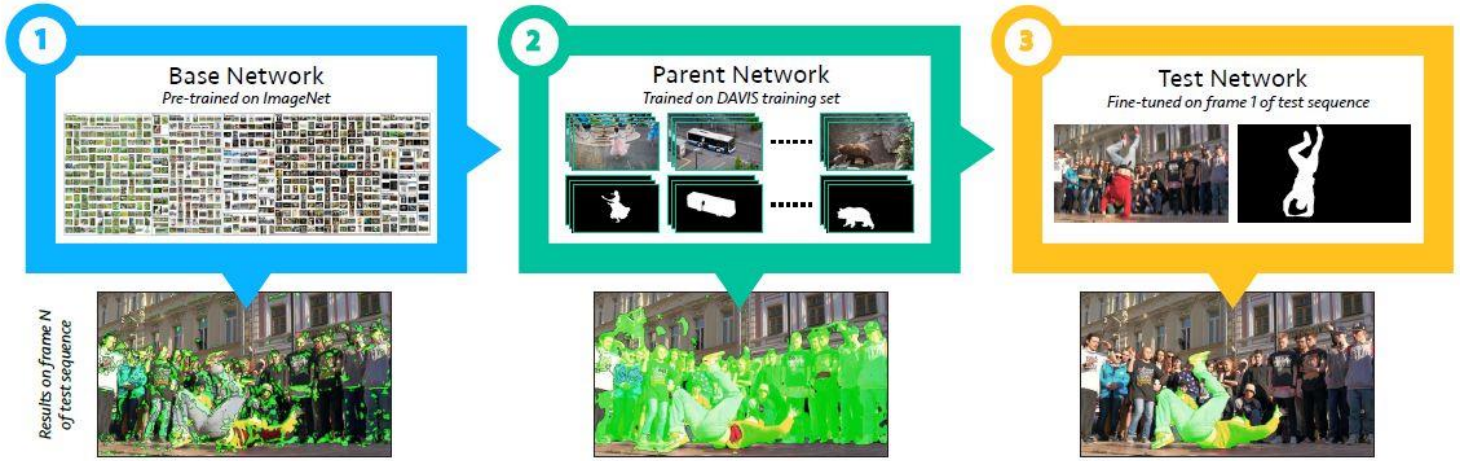


Figure 2. **Overview of OSVOS**: (1) We start with a pre-trained base CNN for image labeling on ImageNet; its results in terms of segmentation, although conform with some image features, are not useful. (2) We then train a *parent network* on the training set of DAVIS; the segmentation results improve but are not focused on an specific object yet. (3) By fine-tuning on a segmentation example for the specific target object in a single frame, the network rapidly focuses on that target.

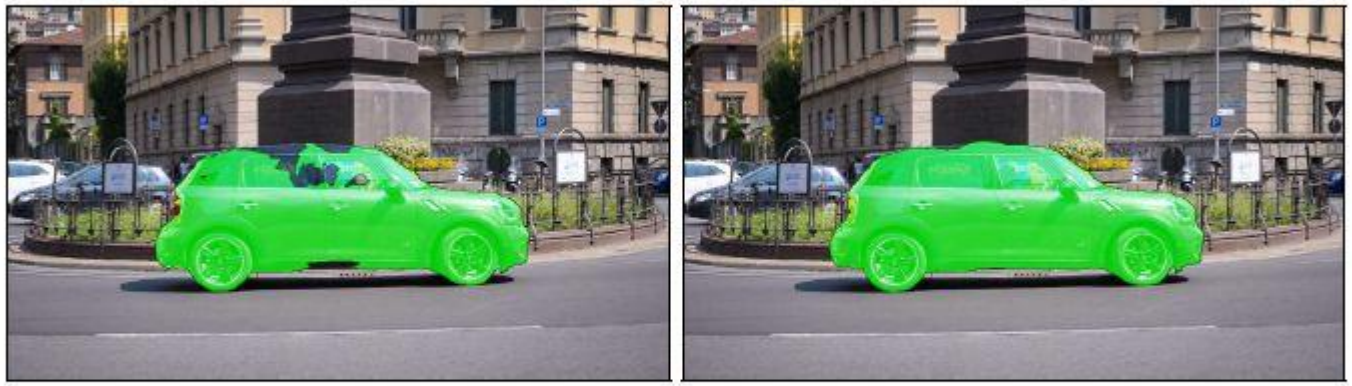


Figure 3. **Qualitative evolution of the fine tuning**: Results at 10 seconds and 1 minute per sequence.

Contour snapping

- Complementary CNN
  - Detect object contours
  - Individual training
    - Only offline
    - Trained on PASCAL-Context Dataset
      - Contour annotations for the whole scene

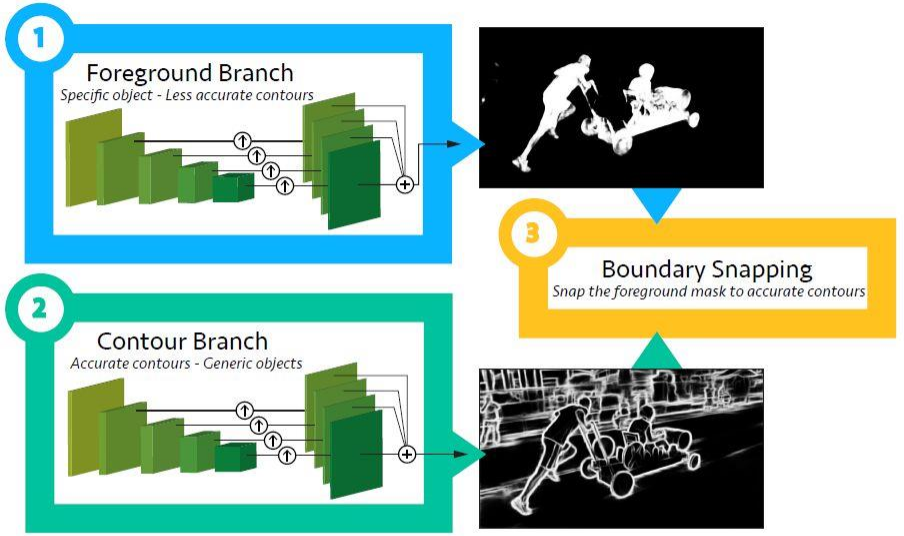


Figure 4. **Two-stream FCN architecture**: The main foreground branch (1) is complemented by a contour branch (2) which improves the localization of the boundaries (3).

- Boundary snapping
  - Compute superpixels that align to the computed contours
    - Ultrametric Contour Map (UCM)*
  - Final mask – superpixels that overlap more than 50% with foreground mask

*Arbelaez et. al. “Contour detection and hierarchical image segmentation”. TPAMI 2011

*Pont-Tuset et. al. “Multiscale combinatorial grouping for image segmentation and object proposal generation”. TPAMI 2017

Ablation study on DAVIS dataset

	Measure	Ours	-BS		-PN-BS		-OS-BS		-PN-OS-BS	
$\mathcal{J}$	Mean $\mathcal{M}\uparrow$	**79.8**	77.4	*2.4*	64.6	*15.2*	52.5	*27.3*	17.6	*62.2*
	Recall $\mathcal{O}\uparrow$	**93.6**	91.0	*2.6*	70.5	*23.2*	57.7	*35.9*	2.3	*91.3*
	Decay $\mathcal{D}\downarrow$	14.9	17.4	*2.5*	27.8	*13.0*	**−1.9**	*16.7*	1.8	*13.1*
$\mathcal{F}$	Mean $\mathcal{M}\uparrow$	**80.6**	78.1	*2.5*	66.7	*13.9*	47.7	*32.9*	20.3	*60.4*
	Recall $\mathcal{O}\uparrow$	**92.6**	92.0	*0.6*	74.4	*18.3*	47.9	*44.7*	2.4	*90.2*
	Decay $\mathcal{D}\downarrow$	15.0	19.4	*4.5*	26.4	*11.4*	**0.6**	*14.3*	2.4	*12.6*
$\mathcal{T}$	Mean $\mathcal{M}\downarrow$	37.6	**33.5**	*4.0*	60.9	*23.3*	53.8	*16.2*	46.0	*8.4*

Table 1. **Ablation study on DAVIS**: Comparison of OSVOS against downgraded versions without some of its components.

Error analysis

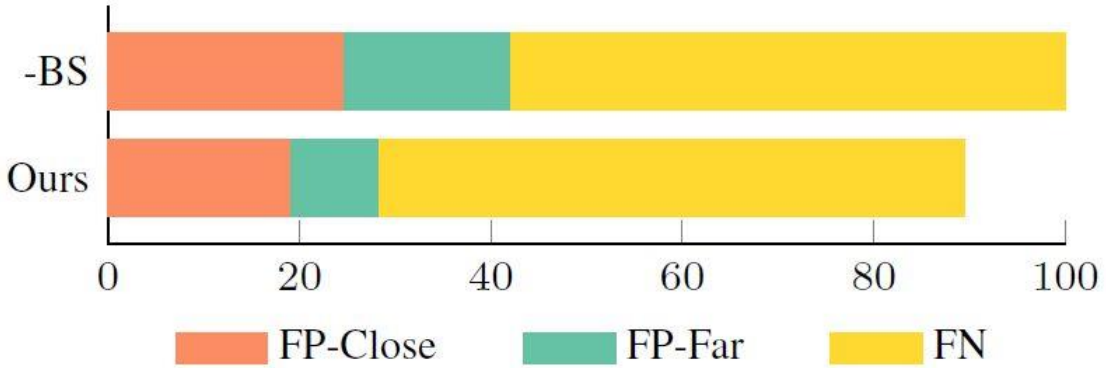


Figure 5. **Error analysis of our method**: Errors divided into False Positives (FP-Close and FP-Far) and False Negatives (FN). Values are total error pixels relative to the error in the -BS case.

Results - DAVIS

Measure	OnAVOS	OSVOS	MSK	SFL	CTN	VPN	PLM	OFL	BVS	FCP	JMP	HVS	SEA
J Mean ↑	**86.1**	79.8	79.7	76.1	73.5	70.2	70.2	68.0	60.0	58.4	57.0	54.6	50.4
J Recall ↑	**96.1**	93.6	93.1	90.6	87.4	82.3	86.3	75.6	66.9	71.5	62.6	61.4	53.1
J Decay ↓	5.2	14.9	8.9	12.1	15.6	12.4	11.2	26.4	28.9	**-2.0**	39.4	23.6	36.4
F Mean ↑	**84.9**	80.6	75.4	76.0	69.3	65.5	62.5	63.4	58.8	49.2	53.1	52.9	48.0
F Recall ↑	89.7	**92.6**	87.1	85.5	79.6	69.0	73.2	70.4	67.9	49.5	54.2	61.0	46.3
F Decay ↓	5.8	15.0	9.0	10.4	12.9	14.4	14.7	27.2	21.3	**-1.1**	38.4	22.7	34.5
T (GT 8.8) ↓	19.0	37.8	21.8	18.9	22.0	32.4	31.8	22.2	34.7	30.6	15.9	36.0	**15.4**

Measure	ARP	LVO	FSEG	LMP	SFL	FST	CUT	NLC	MSG	KEY	CVOS	TRC
J Mean ↑	**76.2**	75.9	70.7	70.0	67.4	55.8	55.2	55.1	53.3	49.8	48.2	47.3
J Recall ↑	**91.1**	89.1	83.5	85.0	81.4	64.9	57.5	55.8	61.6	59.1	54.0	49.3
J Decay ↓	7.0	0.0	1.5	1.3	6.2	**0.0**	2.2	12.6	2.4	14.1	10.5	8.3
F Mean ↑	70.6	**72.1**	65.3	65.9	66.7	51.1	55.2	52.3	50.8	42.7	44.7	44.1
F Recall ↑	**83.5**	83.4	73.8	79.2	77.1	51.6	61.0	51.9	60.0	37.5	52.6	43.6
F Decay ↓	7.9	**1.3**	1.8	2.5	5.1	2.9	3.4	11.4	5.1	10.6	11.7	12.9
T (GT 8.8) ↓	39.3	26.5	32.8	57.2	28.2	36.6	27.7	42.5	30.1	26.9	**25.0**	39.1

Attribute-based performance

Attr	Ours		OFL		BVS		FCP		JMP		HVS		SEA	
AC	**80.6**	*–1.2*	56.6	*17.6*	48.6	*17.6*	52.8	*8.6*	52.4	*7.0*	41.4	*20.4*	43.2	*11.1*
DB	**74.3**	*6.5*	44.3	*27.9*	31.9	*33.0*	53.4	*5.9*	40.7	*19.1*	42.9	*13.9*	31.1	*22.7*
FM	**76.5**	*5.1*	49.6	*28.2*	44.8	*23.3*	50.7	*11.9*	45.2	*18.0*	34.5	*31.0*	30.9	*30.1*
MB	**73.7**	*11.0*	55.5	*22.8*	53.7	*11.5*	50.9	*13.6*	50.9	*11.1*	42.3	*22.5*	39.3	*20.3*
OCC	**77.2**	*3.7*	67.3	*1.0*	67.3	*–10.4*	49.2	*13.2*	45.1	*16.9*	48.7	*8.5*	38.2	*17.5*

Table 3. **Attribute-based performance**: Quality of the techniques on sequences with a certain attribute and the gain with respect to this quality in the sequences without it (in italics and smaller font). See DAVIS [37] for the meaning of the acronyms.

AC – appearance change; DB – dynamic background;

FM- fast motion; MB – motion blur; OCC- occlusion

Number of training images – parent network

Training data	100	200	600	1000	2079
Quality ($\mathcal{J}$)	74.6	76.9	77.2	77.3	77.4

Table 4. **Amount of training data**: Region similarity ($\mathcal{J}$) as a function of the number of training images. Full DAVIS is 2079.

Speed accuracy trade-off

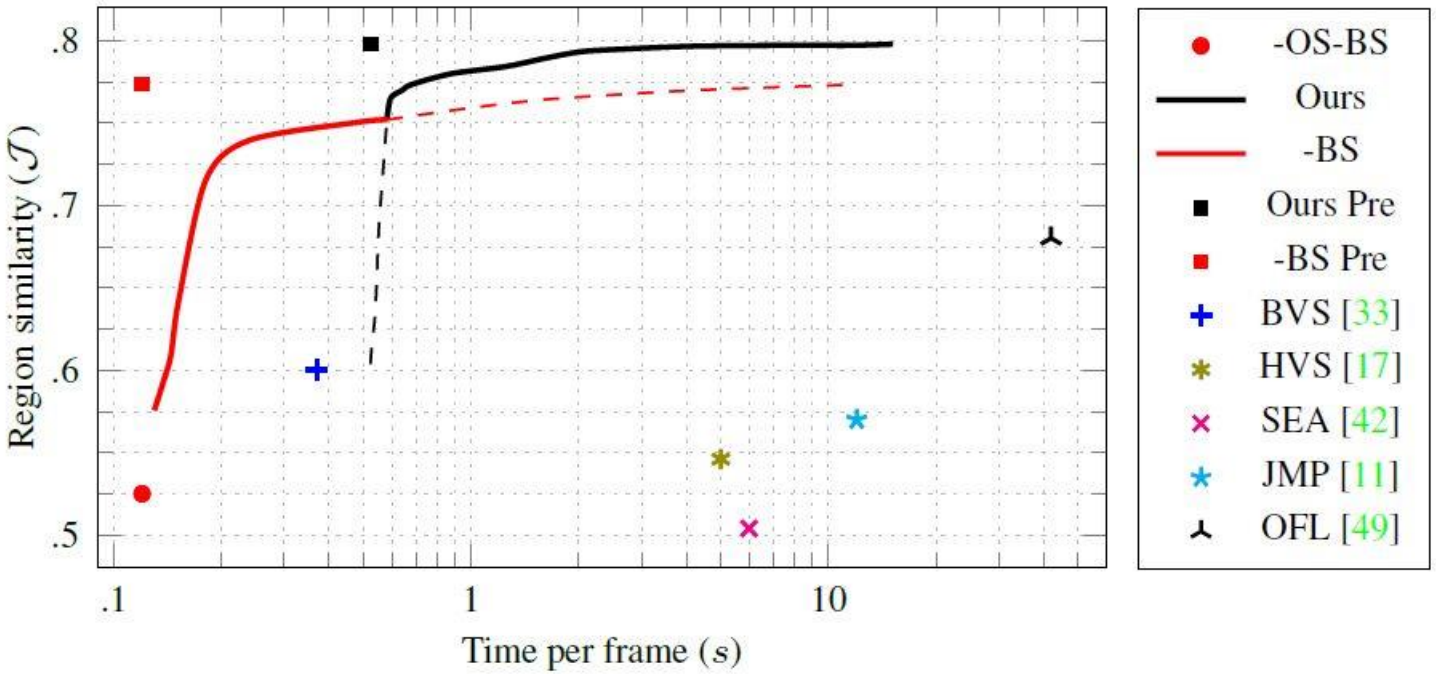
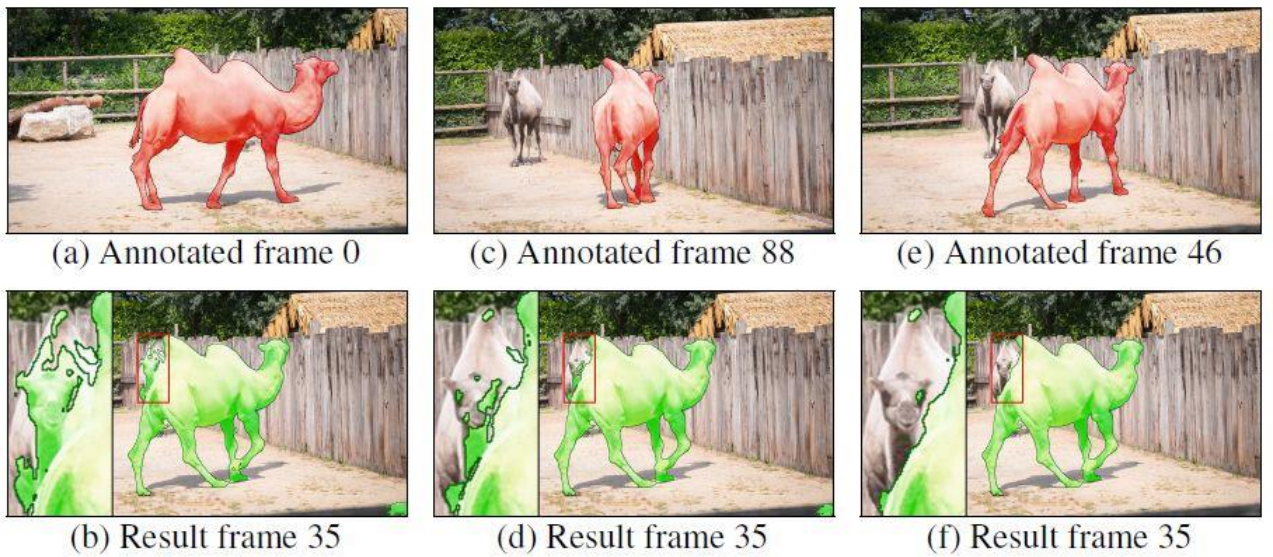


Figure 8. **Quality versus timing**: Region similarity with respect to the processing time per frame.

Progressive refinement

Annotations	0	1	2	3	4	5	All
Quality ($\mathcal{J}$)	58.5	79.8	84.6	85.9	86.9	87.5	88.7

Table 5. **Progressive refinement**: Quality achieved with respect to the number of annotated frames OSVOS trains from.



(a) Annotated frame 0 (c) Annotated frame 88 (e) Annotated frame 46

(b) Result frame 35 (d) Result frame 35 (f) Result frame 35

Figure 9. **Qualitative incremental results**: The segmentation on frame 35 improves after frames 0, 88, and 46 are annotated.

Evaluation as a tracker

Overlap	0.5	0.6	0.7	0.8	0.9
Ours	**78.2**	**72.2**	**65.8**	**59.4**	**49.6**
MDNET [32]	66.4	57.8	43.4	29.5	14.7

Table 6. **Evaluation as a tracker**: Percentage of bounding boxes that match with the ground truth at different levels of overlap.

Results – YouTube Objects

Category	Ours	OFL	JFS	BVS	SCF	AFS	FST	HBT	LTV
Aeroplane	88.2	**89.9**	89.0	86.8	86.3	79.9	70.9	73.6	13.7
Bird	**85.7**	84.2	81.6	80.9	81.0	78.4	70.6	56.1	12.2
Boat	**77.5**	74.0	74.2	65.1	68.6	60.1	42.5	57.8	10.8
Car	79.6	**80.9**	70.9	68.7	69.4	64.4	65.2	33.9	23.7
Cat	**70.8**	68.3	67.7	55.9	58.9	50.4	52.1	30.5	18.6
Cow	77.8	**79.8**	79.1	69.9	68.6	65.7	44.5	41.8	16.3
Dog	**81.3**	76.6	70.3	68.5	61.8	54.2	65.3	36.8	18.0
Horse	**72.8**	72.6	67.8	58.9	54.0	50.8	53.5	44.3	11.5
Motorbike	73.5	**73.7**	61.5	60.5	60.9	58.3	44.2	48.9	10.6
Train	75.7	76.3	**78.2**	65.2	66.3	62.4	29.6	39.2	19.6
Mean	**78.3**	77.6	74.0	68.0	67.6	62.5	53.8	46.3	15.5

Table 7. **Youtube-Objects evaluation**: Per-category mean intersection over union ($\mathcal{J}$).

Qualitative results

Figure 7. **Qualitative results**: First row, an especially difficult sequence which OSVOS segments well. Second row, OSVOS' worst result.

OnAVOS – ”Online Adaptation of Convolutional Neural Networks for Video Object Segmentation”

- Online adaptation of OSVOS
- Because OSVOS is not able to adapt to large changes in object appearance
- Select training examples by choosing pixels for which the network is very certain

OnAVOS

- Contributions:
  - Online updated – adapt to changes in appearance
  - More recent network architecture
  - Additional objectness pretraining step
    - **OSVOS**:
      - base network -> parent network -> test network
    - **OnAVOS**:
      - base network -> objectness network -> domain specific objectness network -> test network

OnAVOS

- Base network
  - ResNet
  - ImageNet, COCO, PASCAL
- Objectness network*
  - Trained PASCAL dataset
    - Foreground vs background
- Domain specific objectness network
  - Trained on DAVIS training set

*Jain et. al. “Fusionseg: Learning to combine motion and appearance for fully automatic segmentation of generic objects in videos”. CVPR 2017

OnAVOS

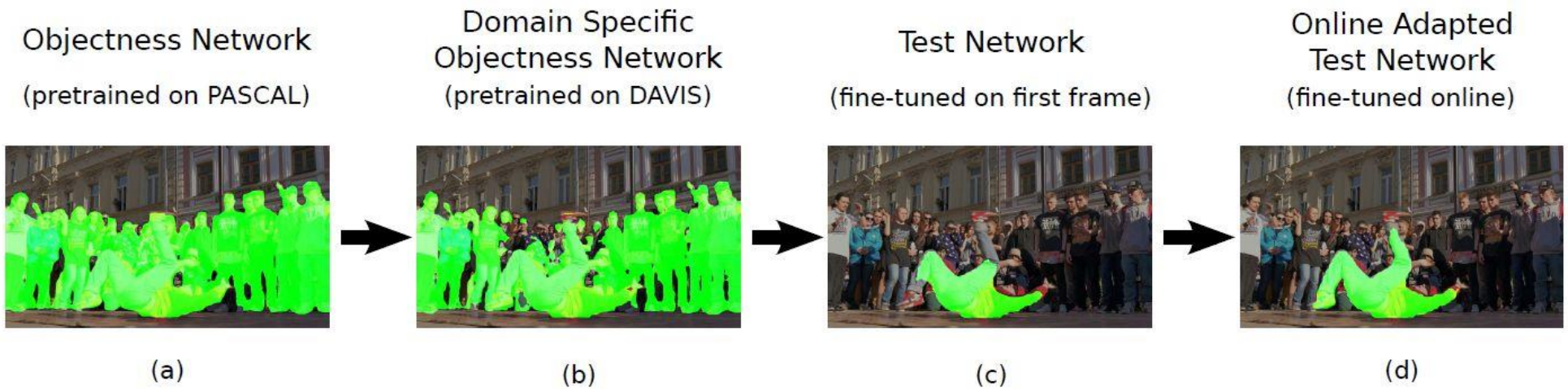


Figure 2: The pipeline of *OnAVOS*. Starting from pretrained weights, the network is first pretrained for objectness on PASCAL (a). Afterwards we pretrain on DAVIS to incorporate domain specific information (b). During test time, we fine-tune on the first frame, to obtain the test network (c). On the following frames, the network is then fine-tuned online to adapt to the changes in appearance (d).

Online adaptation

- Why?
  - Object of interest changes over time
  - New background objects can appear
- Current frame samples
  - Positive samples
    - Use pixels with very confident predictions as training examples
    - Adaptation retains memory
  - Negative samples
    - Pixels that are far away from the last predicted object mask
  - Don’t care area
- First frame is also used for online fine-tuning
  - Avoid drifting

OnAVOS – qualitative results

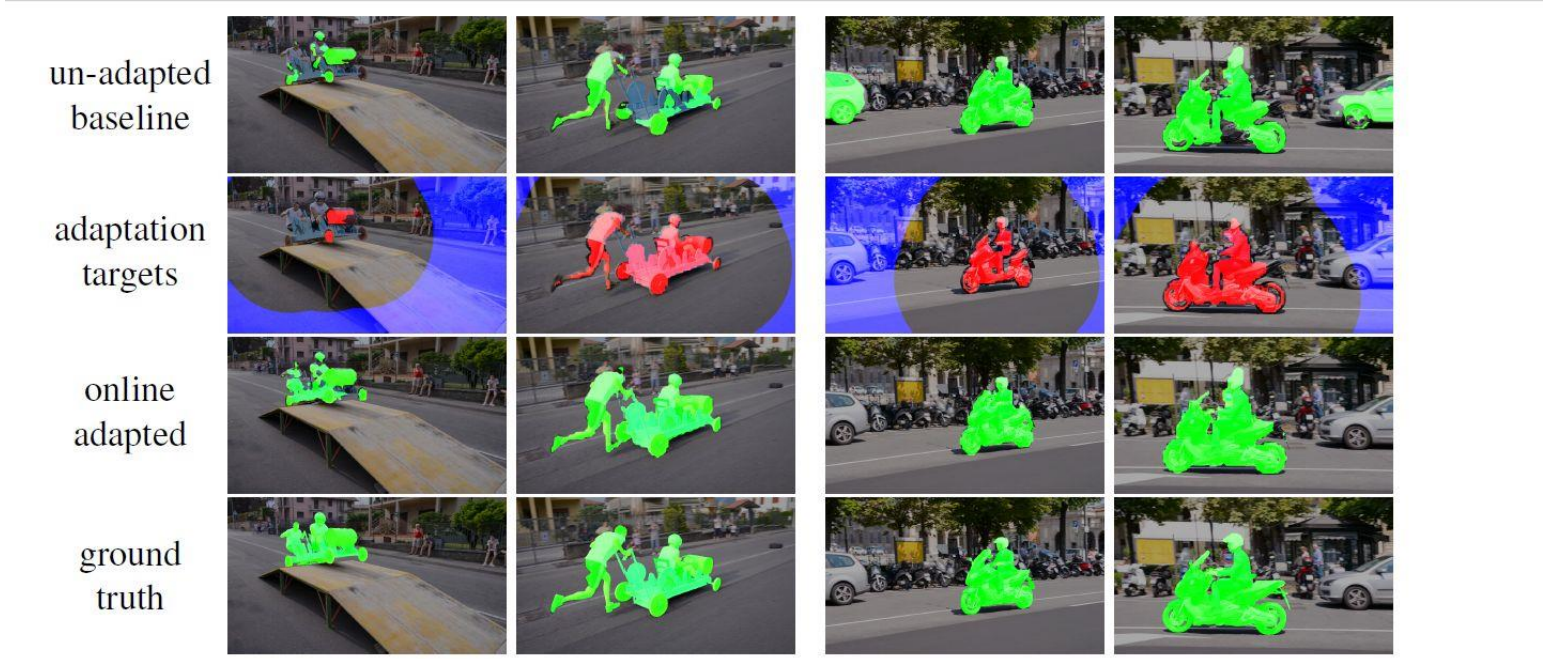


Figure 1: Qualitative results on two sequences of the DAVIS validation set. The second row shows the pixels selected as positive (red) and negative (blue) training examples. It can be seen that after online adaptation, the network can deal better with changes in viewpoint (left) and new objects appearing in the scene (the car in the right sequence).

Online adaptation

Algorithm 1 Online Adaptive Video Object Segmentation (*OnAVOS*)

Input: Objectness network $\mathcal{N}$, positive threshold α, distance threshold d, total online steps n_{online}, current frame steps n_{curr}

Fine-tune $\mathcal{N}$ for 50 steps on $frame(1)$
$lastmask \leftarrow ground\_truth(1)$
for $t = 2 \ldots T$ **do**
    $lastmask \leftarrow erosion(lastmask)$
    $dtransform \leftarrow distance\_transform(lastmask)$
    $negatives \leftarrow dtransform > d$
    $posteriors \leftarrow forward(\mathcal{N}, frame(t))$
    $positives \leftarrow (posteriors > \alpha) \setminus negatives$
    if $lastmask \neq \emptyset$ **then**
        interleaved:
            Fine-tune $\mathcal{N}$ for n_{curr} steps on $frame(t)$ using *positives* and *negatives*
            Fine-tune $\mathcal{N}$ for $n_{online} - n_{curr}$ steps on $frame(1)$ using $ground\_truth(1)$
    end if
    $posteriors \leftarrow forward(\mathcal{N}, frame(t))$
    $lastmask \leftarrow (posteriors > 0.5) \setminus negatives$
    Output *lastmask* for frame t
end for

Ablation study - DAVIS

PASCAL	DAVIS	First frame	mIoU [%]
✓	✓	✓	**80.3** $\pm$ 0.4
	✓	✓	78.0 $\pm$ 0.1
✓		✓	77.6 $\pm$ 0.4
✓	✓		72.7
✓			65.3
	✓		71.0
		✓	65.2 $\pm$ 1.0

Table 1: Effect of (pre-)training steps on the DAVIS validation set. As can be seen, each of the three training steps are useful. The objectness pretraining step on PASCAL significantly improves the results.

Ablation study - DAVIS

Method	mIoU [%]
No adaptation	80.3 ± 0.4
Full adaptation	$\mathbf{82.8} \pm 0.5$
Only negatives	82.4 ± 0.3
Only positives	81.6 ± 0.3
No first frame during online adaptation	69.1 ± 0.2

Table 2: Online adaptation ablation experiments on the DAVIS validation set. As can be seen, mixing in the first frame during online updates is essential, and negative examples are more important than positive ones.

Results

Method	DAVIS mIoU [%]	YouTube-Objects mIoU [%]
OnAVOS (ours), no adaptation	80.3 ± 0.4	76.1 ± 1.3
+CRF	81.7 ± 0.5	76.4 ± 0.2
+CRF +Test time augmentations	81.7 ± 0.2	76.6 ± 0.1
OnAVOS (ours), online adaptation	82.8 ± 0.5	76.8 ± 0.1
+CRF	84.3 ± 0.5	77.2 ± 0.2
+CRF +Test time augmentations	**85.7** ± 0.6	**77.4** ± 0.2
OSVOS [7]	79.8	72.5
MaskTrack [35]	79.7	72.6
LucidTracker [24] †	80.5	*76.2*
VPN [22]	75.0	-

Table 3: Comparison to the state of the art on the DAVIS validation set and the YouTube-Objects dataset. †: Concurrent work only published on arXiv. More results are shown in the supplementary material.

Results - DAVIS

Measure	OnAVOS	OSVOS	MSK	SFL	CTN	VPN	PLM	OFL	BVS	FCP	JMP	HVS	SEA
J Mean ↑	**86.1**	79.8	79.7	76.1	73.5	70.2	70.2	68.0	60.0	58.4	57.0	54.6	50.4
J Recall ↑	**96.1**	93.6	93.1	90.6	87.4	82.3	86.3	75.6	66.9	71.5	62.6	61.4	53.1
J Decay ↓	5.2	14.9	8.9	12.1	15.6	12.4	11.2	26.4	28.9	**-2.0**	39.4	23.6	36.4
F Mean ↑	**84.9**	80.6	75.4	76.0	69.3	65.5	62.5	63.4	58.8	49.2	53.1	52.9	48.0
F Recall ↑	89.7	**92.6**	87.1	85.5	79.6	69.0	73.2	70.4	67.9	49.5	54.2	61.0	46.3
F Decay ↓	5.8	15.0	9.0	10.4	12.9	14.4	14.7	27.2	21.3	**-1.1**	38.4	22.7	34.5
T (GT 8.8) ↓	19.0	37.8	21.8	18.9	22.0	32.4	31.8	22.2	34.7	30.6	15.9	36.0	**15.4**

Measure	ARP	LVO	FSEG	LMP	SFL	FST	CUT	NLC	MSG	KEY	CVOS	TRC
J Mean ↑	**76.2**	75.9	70.7	70.0	67.4	55.8	55.2	55.1	53.3	49.8	48.2	47.3
J Recall ↑	**91.1**	89.1	83.5	85.0	81.4	64.9	57.5	55.8	61.6	59.1	54.0	49.3
J Decay ↓	7.0	0.0	1.5	1.3	6.2	**0.0**	2.2	12.6	2.4	14.1	10.5	8.3
F Mean ↑	70.6	**72.1**	65.3	65.9	66.7	51.1	55.2	52.3	50.8	42.7	44.7	44.1
F Recall ↑	**83.5**	83.4	73.8	79.2	77.1	51.6	61.0	51.9	60.0	37.5	52.6	43.6
F Decay ↓	7.9	**1.3**	1.8	2.5	5.1	2.9	3.4	11.4	5.1	10.6	11.7	12.9
T (GT 8.8) ↓	39.3	26.5	32.8	57.2	28.2	36.6	27.7	42.5	30.1	26.9	**25.0**	39.1