Supplementary Material for
Fast Online Object Tracking and Segmentation: A Unifying Approach

Qiang Wang* — CASIA — qiang.wang@nlpr.ia.ac.cn

Li Zhang* — University of Oxford — lz@robots.ox.ac.uk

Luca Bertinetto* — Five AI — luca.bertinetto@five.ai

Weiming Hu — CASIA — wmhu@nlpr.ia.ac.cn

Philip H.S. Torr — University of Oxford — philip.torr@eng.ox.ac.uk

1. Network architecture details

Network backbone. Table 1 illustrates the details of our *backbone* architecture (f_θ in the main paper). For both variants, we use a ResNet-50 [2] until the final convolutional layer of the 4-th stage. In order to obtain a higher spatial resolution in deep layers, we reduce the output stride to 8 by using convolutions with stride 1. Moreover, we increase the receptive field by using dilated convolutions [1]. Specifically, we set the stride to 1 and the dilation rate to 2 in the 3×3 conv layer of `conv4_1`. Differently to the original ResNet-50, there is no downsampling in `conv4_x`. We also add to the backbone an *adjust* layer (a 1×1 convolutional layer with 256 output channels). Examplar and search patches share the network's parameters from `conv1` to `conv4_x`, while the parameters of the *adjust* layer are not shared. The output features of the adjust layer are then depth-wise cross-correlated, resulting a feature map of size 17×17.

Network heads. The network architecture of the branches of both variants are shows in Table 2 and 3. The `conv5` block in both variants contains a normalisation layer and ReLU non-linearity while `conv6` only consists of a 1×1 convolutional layer.

Mask refinement module. With the aim of producing a more accurate object mask, we follow the strategy of [5], which merges low and high resolution features using multiple *refinement* modules made of upsampling layers and skip connections. Figure 1 illustrates how a mask is generated with stacked refinement modules. Figure 2 gives an example of refinement module U_3.

block	*examplar* output size	*search* output size	*backbone*
conv1	61×61	125×125	7×7, 64, stride 2
conv2_x	31×31	63×63	3×3 max pool, stride 2 $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	15×15	31×31	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$
conv4_x	15×15	31×31	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$
adjust	15×15	31×31	1×1, 256
xcorr	17 × 17		depth-wise

Table 1: Backbone architecture. Details of each building block are shown in square brackets.

block	score	box	mask
conv5	1×1, 256	1×1, 256	1×1, 256
conv6	1×1, $2k$	1×1, $4k$	1×1, (63×63)

Table 2: Architectural details of the *three-branch* head. k denotes the number of anchor boxes per RoW.

block	score	mask
conv5	1×1, 256	1×1, 256
conv6	1×1, 1	1×1, (63×63)

Table 3: Architectural details of the *two-branch* head.

*Equal contribution. Work done while at University of Oxford.

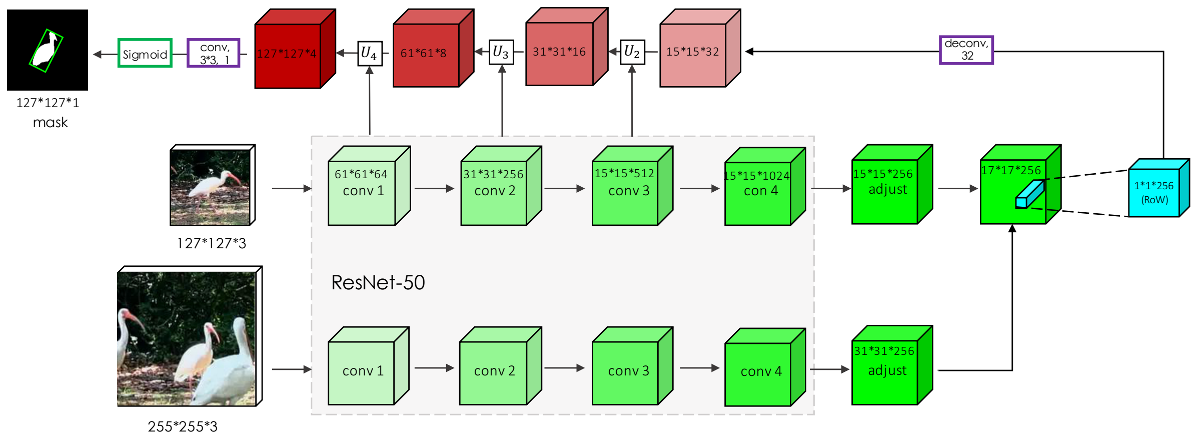

Figure 1: Schematic illustration of mask generation with stacked refinement modules.

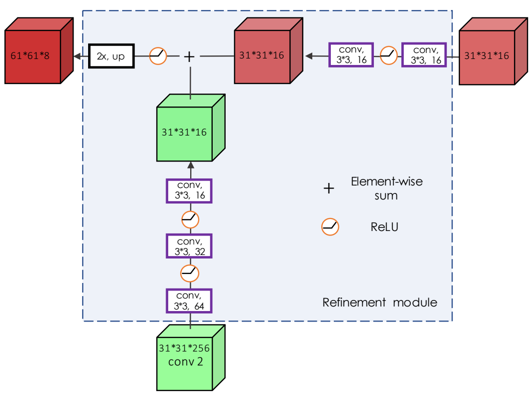

Figure 2: Example of a refinement module U_3.

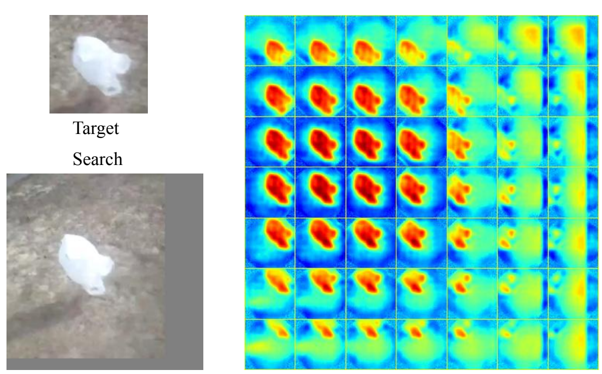

Figure 3: Score maps from Mask branch at different locations.

2. Further qualitative results

Different masks at different locations. Our model generates a mask for each RoW. During inference, we rely on the score branch to select the final output mask (using the location attaining the maximum score). The example of Figure 3 illustrates the multiple output masks produced by the mask branch, each corresponding to a different RoW.

Benchmark sequences. More qualitative results for VOT and DAVIS sequences are shown in Figure 4 and 5.

References

[1] 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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[3] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pfugfelder, L. C. Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey, G. Fernandez, and et al. The sixth visual object tracking vot2018 challenge results. In *European Conference on Computer Vision workshops*, 2018.

[4] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *European Conference on Computer Vision*, 2016.

[6] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Figure 4: Further qualitative results of our method on sequences from the visual object tracking benchmark VOT-2018 [3].

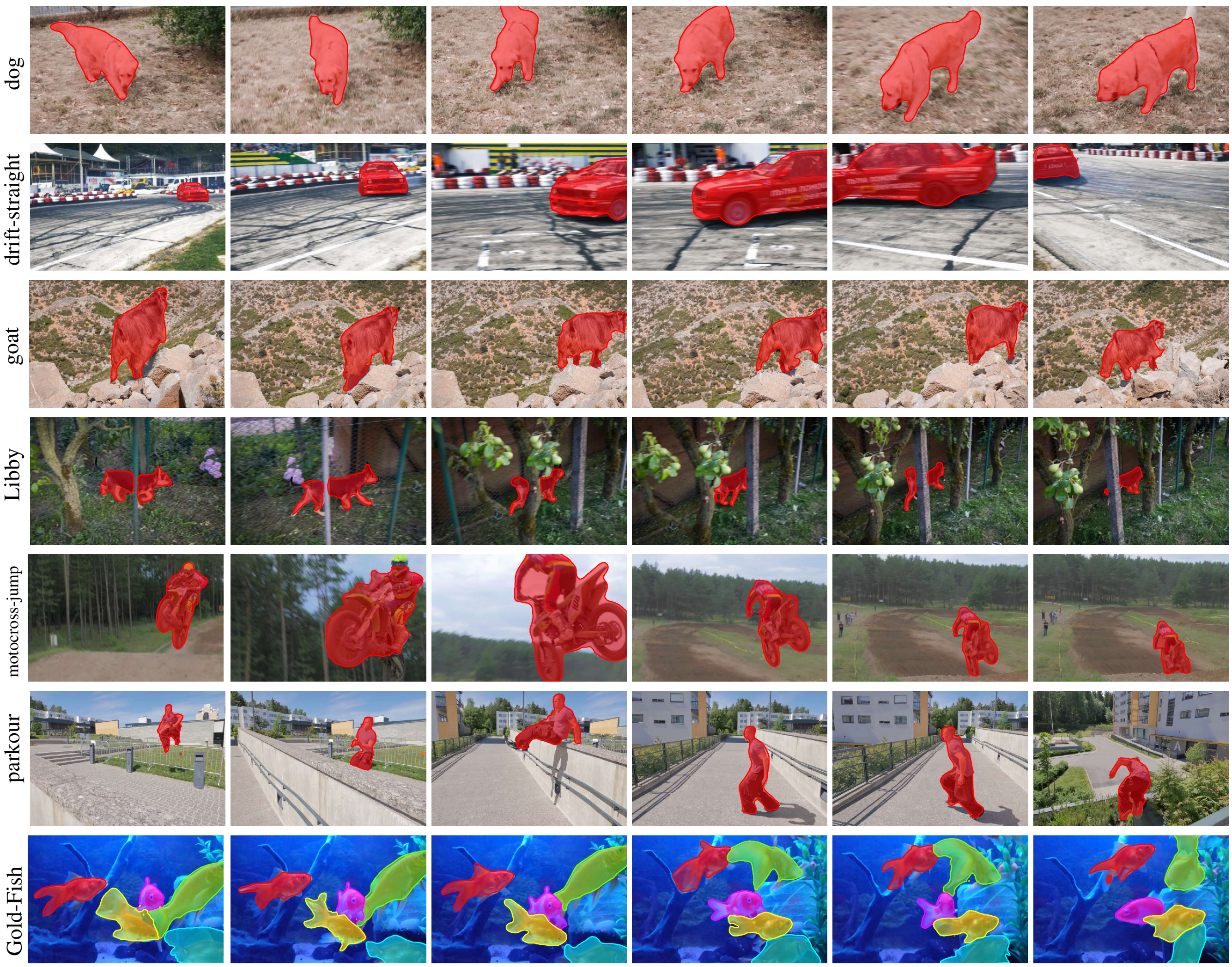

Figure 5: Further qualitative results of our method on sequences from the semi-supervised video object segmentation benchmarks DAVIS-2016 [4] and DAVIS-2017 [6].