Appendix
SALISA: Saliency-based Input Sampling for Efficient Video Object Detection

Babak Ehteshani Bejnordi, Amirhossein Habibian, Fatih Porikli, and Amir Ghodrati

Qualcomm AI Research*
{behtesha,ahabibia,fporikli,ghodrati}@qti.qualcomm.com

1 Training protocol

We first trained the resampling module independently from the detection network using the regularization loss described in Section 3.2.1 of the main paper. For this purpose, we first generated saliency maps from the ground-truth annotations of the UA-DETRAC training dataset and trained the sampler using Adam optimizer [2] with a learning rate of $1e^{-3}$ for 10 epochs. Note that we did not pre-train our resampling module on the ImageNet-VID dataset as the pretrained weights from UA-DETRAC were already suitable for ImageNet-VID as well. For both datasets, we then trained the EfficientDet networks, pre-trained on MS-COCO [3], in an image-based fashion using SGD optimizer with momentum 0.9, weight decay $4e^{-5}$, and an initial learning rate of 0.01. For ImageNet-VID, we trained the models for 7 epochs and the learning rate was dropped with a factor of 0.1 at epochs 3 and 6. For UA-DETRAC, we trained the models for 4 epochs and dropped the learning rate with a factor of 0.1 at epoch 3. In the final step, we fine-tuned the resampling module and the object detection networks end-to-end using SGD with a learning rate of $1e^{-3}$ for 3 epochs. The models were trained with a mini-batch size of 4 using four GPUs and synchronized batch-norm. We used standard data augmentations [4] commonly used to train EfficientDet in our experiments.

2 Thin-plate Spline transformation

In this section, we explain the mathematical details of the Thin Plate Spline (TPS) warping function. Given a set of N control points on a 2D grid $\dot{P} \in \mathbb{R}^{N\times 2}$, and their transformed positions $\dot{V} \in \mathbb{R}^{N\times 2}$, we solve for two functions $f_{x'}$ and $f_{y'}$, from which we can sample discrete displacements along x and y coordinates, as follows:

$$f_{x'}(x, y) = a_1 + a_2 x + a_3 y + \sum_{i=0}^{N} \alpha_i U(\|(x_i, y_i) - (x, y)\|) \tag{1}$$

$$f_{y'}(x, y) = a_4 + a_5 x + a_6 y + \sum_{i=0}^{N} \beta_i U(\|(x_i, y_i) - (x, y)\|) \tag{2}$$

[*] Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc

The term $r_{ij} = \|(x_i, y_i) - (x_j, y_j)\|$ represents the distance between the control points $\dot{P}_j = (x_j, y_j)$ and $\dot{P}_i = (x_i, y_i)$, and $U(r) = r^2 log(r)$ is the radial basis kernel. The six parameters $a_1, a_2, ..., a_6$ correspond to the global affine transformation of TPS, and $2N$ parameters $(\alpha_i, \beta_i) \in \{1, 2, ..., N\}$ correspond to local transformation. Let us define the matrices K, P, W, and V as follows:

$$K = \begin{bmatrix} 0 & U(r_{12}) & ... & U(r_{1N}) \\ U(r_{21}) & 0 & ... & U(r_{2N}) \\ ... & ... & ... & ... \\ U(r_{N1}) & U(r_{N2}) & ... & 0 \end{bmatrix}, N \times N, \qquad P = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_1 & y_1 \\ ... & ... & ... \\ 1 & x_N & y_N \end{bmatrix}, N \times 3, \tag{3}$$

$$W = \begin{bmatrix} \alpha_1 & \beta_1 \\ \alpha_1 & \beta_1 \\ ... & ... \\ \alpha_N & \beta_N \\ a_1 & a_4 \\ a_2 & a_5 \\ a_3 & a_6 \end{bmatrix}, (N+3) \times 2, \qquad V = \begin{bmatrix} x'_1 & y'_1 \\ x'_1 & y'_1 \\ ... & ... \\ x'_N & y'_N \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, (N+3) \times 2, \tag{4}$$

The TPS coefficients can be calculated by solving the following linear problem:

$$W = \underbrace{\left[\begin{array}{c|c} K & P \\ \hline P^T & O \end{array}\right]^{-1}}_{L} \times V \tag{5}$$

Once W is calculated, we plug it into equations (1) and (2), to sample $f_{x'}$ and $f_{y'}$ given a point (x, y).

3 Ablations

3.1 Analysis of the area threshold.

The resampling module preserves the resolution of all salient objects, however, it gives extra focus in preserving the resolution of small object. The area threshold α determines which objects should be considered as small in the saliency map. As seen in Table 1, generally, a smaller value of α improves small object detection without hurting the detection performance of medium and large objects. This is because smaller α values increasingly move the focus of the sampler to preserve the resolution of smaller objects. Extremely small α values (*e.g.* 0.1%), however, may lead to reduced small object detection performance as just limited number of small objects receive extra focus.

3.2 Comparison to tracking baselines on UA-DETRAC

We compared SALISA with two tracking baselines in Table 2. D1+D0 is a baseline that performs detection on the key frame with EfficientDet-D1 and uses EfficientDet-D0

Table 1: Effect of α on the mAP of SALISA with EfficientDet-D1.

Threshold	Small	Medium	Large
$\alpha = 2.0$	13.1	58.1	76.3
$\alpha = 1.0$	13.3	58.2	76.6
$\alpha = 0.5$	**14.9**	58.2	76.6
$\alpha = 0.1$	13.4	58.1	76.9

for the next 7 subsequent frames (no tracking). D1+Copy is a baseline that performs detection on the key frame with EfficientDet-D1 and copies the obtained boxes for the next 7 frames. In contrast, D1+SiamFC updates the boxes in non-key frames using SiamFC tracker [1]. SALISA with the same setup (EfficientDet-D1 as the key frame detector and EfficientDet-D0 for the next 7 frames), outperforms all these methods by a large margin.

Table 2: Comparison of SALISA with several tracking baselines.

Method	mAP	FLOPs
D1 + D0	52.2	2.9 G
D1 + Copy	30.6	0.8 G
D1 + SiamFC [1]	51.6	3.1 G
D1 + SALISA (D0)	56.0	2.9 G

3.3 Impact of τ for saliency map generation

In this ablation, we analyze the effect of changing the parameter τ for generating the saliency maps in section 3.1. By reducing τ, we can include more uncertain detections in our saliency maps and potentially allow for a more accurate decision for those detections in subsequent frames. As can be seen in Table 3, decreasing τ to 0.3 gives a small but consistent improvement for all models. Decreasing τ further down to 0.1 still comes with a small accuracy improvement compared to $\tau = 0.5$. Therefore, in practice including uncertain detections can increase the performance, however, in case a model has many false positives, this could also crowd the saliency map and defeat the purpose of focused zooming.

Table 3: Effect of τ on the mAP of SALISA.

Threshold	D0	D1	D2
$\tau = 0.5$	70.2	72.8	75.5
$\tau = 0.3$	70.4	73.0	75.6
$\tau = 0.1$	70.4	72.8	75.6

References

1. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.: Fully-convolutional siamese networks for object tracking. In: European conference on computer vision. pp. 850–865. Springer (2016)
2. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Int. Conf. Learn. Represent. (2014)
3. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
4. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10781–10790 (2020)