Efficient Visual Tracking with Exemplar Transformers (Supplementary Material)

Philippe Blatter*,[1] Menelaos Kanakis*,[1] Martin Danelljan[1] Luc Van Gool[1,2]
[1]ETH Zürich [2]KU Leuven

In this supplementary material, we first provide an ablation study of the number of query vectors S^2 in Sec. A. We present the pseudocode of our Exemplar Transformer Layer in Sec. B. We provide additional results on the VOT2020 [10] real-time challenge in Sec. C. In Sec. D we analyze the qualitative results of our tracker on a selection of sequences, while in Sec. E we further examine the results of our tracker on the LaSOT dataset [7] with respect to the specific attributes. Finally, we present the success plots of the NFS [9], OTB-100 [14], and UAV-123 [12] datasets in Sec. F. The code and the instructions to reproduce our results are included in the supplementary material folder, and will be made available upon publication.

A. Ablation of Number of Query Vectors

As explained in Sec. 3 of the main paper, we set $S = 1$ in our experiments based on the assumption that one global token encapsulates sufficient information for the task of single object tracking. To further evaluate this hypothesis, we ablate the parameter S. Specifically, the input feature map is divided into $S \times S$ patches, for which we compute individual query vectors. Table S.1 presents the results of our experiments. The results confirm our assumption that utilizing a single token as global representation yields the best results.

	S=1	S=2	S=4
NFS	59.0	46.6	46.7
OTB-100	67.8	55.5	57.5
LaSOT	59.1	43.7	42.6

Table S.1: Ablation experiment of the different values for S reported in terms of AUC on NFS, OTB, and LaSOT datasets. Utilizing a global query token ($S = 1$) yields consistently better results. The best score is highlighted in blue.

B. Algorithm

We present below the pseudocode for the Exemplar Attention layer (Eq. 6) , depicted on the right side of Fig. 2 in Algorithm 1.

Algorithm 1 Pseudocode of the Exemplar Attention layer, Eq. 6.

```
function ExemplarAttention($X$):
    $Q \leftarrow \Psi_S(X) W_Q$                                Eq. 3
    $\hat{K} \leftarrow \hat{W}_K$
    $\hat{V} \leftarrow W_V$
    sim $\leftarrow$ softmax$(Q \cdot \hat{K}^T)$
    sim $\leftarrow$ sim$/\sqrt{d_k}$
    $W_A =$ sim $\cdot \hat{V}$
    $A(X) \leftarrow W_A \circledast X$

    return $A(X)$
end
```

C. VOT-RT2020

We evaluate bounding box predicting trackers on the anchor-based short term tracking dataset of VOT-RT2020 [10], similar to Sec. 4.2. The results are presented in Table S.2. While the performance of our model is comparable to LT-Mobile [16] in terms of accuracy, our model is nearly 6% better in terms of robustness. We find that learning exemplar representations from the dataset coupled with an image-level query representation significantly increases the tracker's robustness compared to its convolutional counterpart.

D. Video Visualizations

We additionally provide sequence comparisons between E.T.Track and LT-Mobile [16]. Table S.3 lists the sequences compared, and reports their performance. In addition, the associated videos can be found in the supplementary folder.

person8-2 The person8-2 sequence of the UAV-123

[*] P. Blatter and M. Kanakis contributed equally to this work.

	non-realtime				realtime		
	SiamFC [2]	ATOM [5]	DiMP [3]	SuperDiMP [1]	KCF [8]	LT-Mobile [16]	**E.T.Track** **(Ours)**
EAO	0.172	0.237	0.241	0.289	0.154	0.217	**0.227**
Accuracy	0.422	0.440	0.434	0.472	0.406	0.418	**0.418**
Robustness	0.479	0.687	0.700	0.767	0.434	0.607	**0.663**
CPU Speed	6	20	15	15	95	47	47

Table S.2: Comparison of bounding box predicting trackers on the VOT-RT2020 dataset. We report the Expected Average Overlap (EAO), the Accuracy and Robustness. The best score is highlighted in blue while the best realtime score is highlighted in red. We additionally report CPU runtime speeds in *FPS*.

dataset [12] of a man running on grass nicely demonstrates that our tracker does not lose track of the target even when he partially moves out of the frame. Specifically, E.T.Track is able to completely recover when the target moves back into the frame. LT-Mobile [16] yields comparable results.

Human7 Human7 from the OTB dataset [14] films a woman walking. Even though the video appears to be jittery, the appearance and shape of the target object changes only marginally. Our model achieves an average overlap of 88% which is 7% higher than LT-Mobile [16].

boat-9 The boat-9 from the UAV-123 dataset [12] depicts a target which not only changes appearance, but also significantly decreases in size due to an increasing distance to the camera. We find that E.T.Track can still handle such scenarios, and unlike LT-Mobile, it maintains track of the boat even after a 180-degree turn. E.T.Track is therefore more robust than LT-Mobile, attributed to the increased capacity introduced by the Exemplar Transformer layers.

basketball-3 In the basketball-3 sequence of NFS [9], the increased robustness introduced by the Exemplar Transformer layer enables the separation between the player's head and the basketball, unlike LT-Mobile.

drone-2 The drone-2 sequence of LaSOT [7] shows a target that shortly moves completely out of the frame, and later re-enters the scene with a different appearance to the initial frame. Furthermore, the target object's location deviates from the tracker's search range when re-entering the scene. These two aspects pose a challenge both for our model, as well as LT-Mobile [16], and are inherent limitations of the tracking inference pipeline used in both approaches [17]. Specifically, the tracking pipeline contains a post-processing step in which the predicted bounding boxes are refined. Changes in size, as well as changes of the bounding box aspect ratios, are therefore penalized. In addition, both models search only within a small image patch around the previously predicted target location. This challenge can potentially be addressed by integrating our Exemplar Transformer layer into trackers that directly predicts bounding boxes without any post-processing. We did not investigate this further, but consider this an interesting direction for future research.

	Dataset	LT-Mobile [16]	**E.T.Track** **(Ours)**
person8-2	UAV-123 [12]	0.889	**0.915**
Human7	OTB [14]	0.813	**0.883**
boat-9	UAV-123 [12]	0.483	**0.803**
basketball-3	NFS [9]	0.259	**0.707**
drone-2	LaSOT [7]	0.192	**0.887**

Table S.3: Direct per-sequence comparison of E.T.Track and LT-Mobile [16] on various sequences in terms of Average Overlap (AO). The best performance is highlighted in blue.

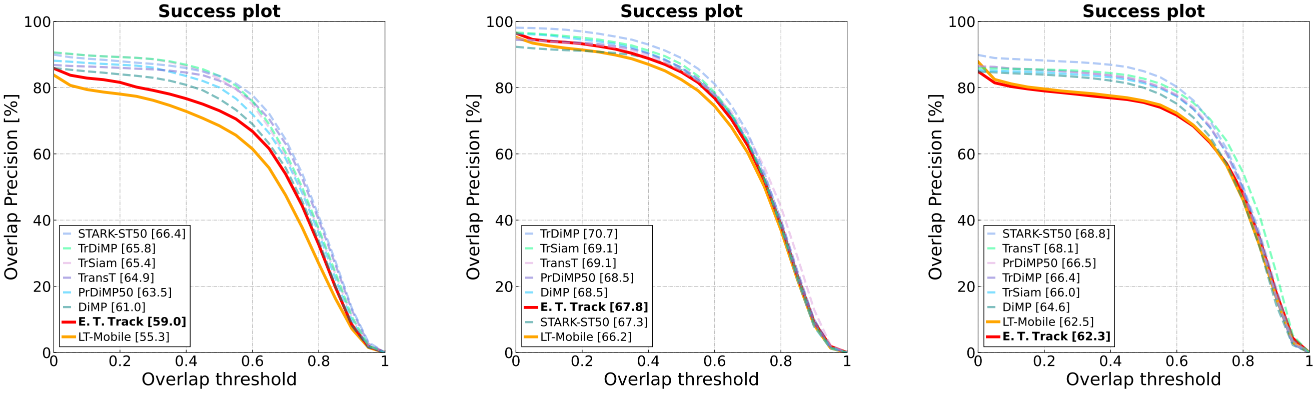
E. Attributes

Table S.4 presents the results of various trackers on different sequence attributes of the LaSOT dataset [7]. We consistently outperform the other realtime trackers by a significant margin in every attribute. The attribute with the largest performance gains compared to LT-Mobile [16] are *Full Occlusion* with 10.4%, *Motion Blur* with 9.7%, *Background Clutter* with 8.5%, and *Fast Motion* with 7.5%. These attributes are either known limitation of the tracking pipeline utilized [17], as discussed in Sec. D, or can benefit from increased network capacity. We find that the incorporation of our Exemplar Transformer layers increases robustness and improves attributes that are even known limitations of the overall framework.

When comparing our model to the non-realtime state-of-the-art STARK [15], our model observes an average performance drop of -7.3%. The most challenging attributes are *Viewpoint Change*, *Full Occlusion*, *Fast Motion*, *Out-of-View*, and *Low Resolution*. This analysis paves the path for future research in the design of novel modules for efficient tracking, specifically tackling the identified challenging attributes.

	Illumination	Partial		Motion	Camera		Background	Viewpoint	Scale	Full	Fast		Low	Aspect	
	Variation	Occlusion	Deformation	Blur	Motion	Rotation	Clutter	Change	Variation	Occlusion	Motion	Out-of-View	Resolution	Ration Change	Total
STARK-ST50	66.8	64.3	66.9	62.9	69.0	66.1	57.3	67.8	66.1	58.7	53.8	62.1	59.4	64.9	66.4
TransT	65.2	62.0	67.0	63.0	67.2	64.3	57.9	61.7	64.6	55.3	51.0	58.2	56.4	63.2	64.9
TrDiMP	67.5	61.1	64.4	62.4	68.1	62.4	58.9	62.8	63.4	56.4	53.0	60.7	58.1	62.3	63.9
TrSiam	63.8	60.1	63.8	61.1	65.5	62.0	55.1	60.8	62.5	54.5	50.6	58.9	56.0	61.2	62.6
PrDiMP50	63.3	57.1	61.3	58.0	64.0	59.1	55.4	61.7	60.1	51.6	49.2	57.0	54.8	59.0	60.5
DiMP	59.5	52.1	56.6	54.6	59.3	54.5	49.7	56.7	55.8	47.5	45.6	49.5	49.1	54.5	56.0
SiamRPN++	53.0	46.6	52.8	44.2	51.3	48.5	44.9	44.4	49.4	36.6	31.6	41.6	38.5	47.2	49.5
LT-Mobile	55.0	48.9	57.3	45.8	52.9	51.2	43.3	49.9	51.9	38.3	33.6	43.7	40.8	49.9	52.1
SiamFC	34.6	30.6	35.1	30.8	33.3	31.0	30.8	28.6	33.2	24.5	19.5	25.6	25.2	30.8	33.6
E.T.Track (Ours)	**61.3**	**55.7**	**61.0**	**55.5**	**60.2**	**58.1**	**51.8**	**55.9**	**58.8**	**48.7**	**41.1**	**51.1**	**48.8**	**56.9**	**59.1**

Table S.4: LaSOT attribute-based analysis. Each column corresponds to the results computed on all sequences in the dataset with the corresponding attribute. The trackers that do not run in real-time are highlighted in grey. The overall best score is highlighted in blue while the best realtime score is highlighted in red.



(a) Success plot on the NFS dataset. Our tracker outperforms LT-Mobile [16] by a significant margin.

(b) Success plot on the OTB-100 dataset. Our tracker outperforms LT-Mobile [16] by a small margin.

(c) Success plot on the UAV-123 dataset. The performance of our tracker is comparable to the performance of LT-Mobile [16].

Figure S.1: Success plots. The CPU realtime trackers are indicated by continuous lines in warmer colours, while the non-realtime trackers are indicated by dashed lines in colder colours.

F. Additional Success Plots

We depict the success plot of the NFS dataset [9] in Fig. S.1a, the success plot of the OTB-100 dataset [14] in Fig. S.1b and the success plot of the UAV-123 dataset [12] in Fig. S.1c. For efficient trackers, we limited our comparison to the mobile architecture presented in LightTrack [16], as SiamRPN++ [11] and SiamFC [2] were consistently outperformed by a large margin. We additionally report the non-realtime transformer-based trackers STARK-ST50 [15], TrDimp [13], TrSiam [13] and TransT [4], as well as the seminal trackers DiMP [3] and PrDiMP [6].

The results presented in Fig. S.1 correspond to our evaluation results, and therefore deviate slightly from those reported in Sec. 4.2 as those were directly acquired from their respective papers. As it can be seen, our model outperforms LT-Mobile [16] on all but one benchmark dataset. More importantly, we want to highlight the shrinking gap between the complex transformer-based trackers and our realtime CPU tracker. Closing this gap even further while maintaining the realtime speed will be a crucial part for future work in order to deploy high-performing trackers on computationally limited edge devices.

References

[1] Pytracking. https://github.com/visionml/pytracking. Accessed: 2021-11-16. 2

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016. 2, 3

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6182–6191, 2019. 2, 3

[4] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8126–8135, June 2021. 3

[5] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Atom: Accurate tracking by overlap maximization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2019. 2

[6] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7183–7192, 2020. 3

[7] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5374–5383, 2019. 1, 2

[8] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):583–596, 2014. 2

[9] Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Need for speed: A benchmark for higher frame rate object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1125–1134, 2017. 1, 2, 3

[10] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kämäräinen, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, Ondrej Drbohlav, et al. The eighth visual object tracking vot2020 challenge results. In *European Conference on Computer Vision*, pages 547–601. Springer, 2020. 1

[11] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019. 3

[12] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *European conference on computer vision*, pages 445–461. Springer, 2016. 1, 2, 3

[13] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1571–1580, June 2021. 3

[14] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2411–2418, 2013. 1, 2, 3

[15] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. *arXiv preprint arXiv:2103.17154*, 2021. 2, 3

[16] Bin Yan, Houwen Peng, Kan Wu, Dong Wang, Jianlong Fu, and Huchuan Lu. Lighttrack: Finding lightweight neural networks for object tracking via one-shot architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15189, 2021. 1, 2, 3

[17] Zhipeng Zhang, Houwen Peng, Jianlong Fu, Bing Li, and Weiming Hu. Ocean: Object-aware anchor-free tracking. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 771–787. Springer, 2020. 2