Leveraging the Power of Data Augmentation for Transformer-based Tracking (Supplementary Material)

Jie Zhao[1], Johan Edstedt[2], Michael Felsberg[2], Dong Wang[1], Huchuan Lu[1]
[1]Dalian University of Technology [2]Linköping University

A. Analysis of General Data Augmentation

A.1. Summary of data augmentation strategies in prior works

As shown in Tab. 2, we summarize the detailed data augmentation strategies of 40 trackers published in recent five years. All trackers adopt the similar augmentation pattern which is random cropping along with several general image transformations. In particular, among 12 transformer-based trackers, eight of them follow the same strategy as prior CNN-based trackers, which are random cropping, grayscale, flip and brightness jitter. In our work, we explore the impact of common used image transformations on modern transformer-based trackers, like flip, brightness jitter, blur, and rotation, and the impact of jitter degree of random cropping.

A.2. Preliminary experiments on general data augmentation

Table 1. **Preliminary experiments for blur and rotation.**

Method →	Blur			Rotation		
Probability →	0.2	0.5	0.2	0.2	0.5	0.2
Magnitude →	1	1	2	15	15	45
LaSOT [13]	68.1	67.4	67.3	67.9	67.8	67.3
LaSOT_EXT [12]	47.2	46.1	45.7	46.6	46.6	46.0

In our work, we add (blur or rotation) or remove (brightness jitter or flip) each augmentation to explore their impact on transformer-based trackers separately. To avoid negative effects from inappropriately tuned probability and magnitude, we perform a set of preliminary experiments with different parameter settings for these two added approaches, *i.e*. blur and rotation, and then choose the best-performing parameters in our systematic experiments. Success plots (AUC) on LaSOT [13] and LaSOT_EXT [12] are shown in Tab. 1, where the chosen settings for our systematic experiments are highlighted with colors. We find that for both of them, the setting of low probability with low magnitude is able to obtain relatively better performance.

B. Attribute analysis

To analyze the effect of our DATr in the face of different challenges, we report the AUC gains from our DATr on different attributes. As shown in Fig. 1, the baseline tracker here is OSTrack256 [37]. The dotted lines represent the AUC gains on each benchmark, while the red solid line represents the average performance gain. We can see that our DATr improves models mainly from challenges like viewpoint change, out-of-view, fast motion, low resolution and motion blur, which often accompany by failure tracking and obvious shifts of objects' positions between adjacent frames. In particular, for each challenge like aspect ratio change, scale variation, background clutter, and occlusion, our DATr also obtains more than 1.5% AUC average gain on the two benchmarks.

Table 2. **Summary of data augmentation strategies of 40 trackers published in recent five years.**

	Method	Source	Grayscale	Random cropping	Flip	Brightness jitter	Blur	Rotation
CNN	SiamRPN [19]	CVPR18		✔		✔	✔	
	DaSiamRPN [44]	ECCV18		✔		✔	✔	
	ATOM [9]	CVPR19	✔	✔		✔		
	SiamDW [40]	CVPR19		✔		✔	✔	
	SiamMask [31]	CVPR19	✔	✔			✔	
	SiamRPN++ [18]	CVPR19		✔		✔	✔	
	DiMP [1]	ICCV19	✔	✔		✔		
	GradNet [20]	ICCV19		✔		✔		
	D3S [23]	CVPR20		✔		✔		
	PrDiMP [10]	CVPR20	✔	✔		✔		
	ROAM [36]	CVPR20		✔				
	SiamAttn [38]	CVPR20		✔	✔	✔	✔	
	SiamBAN [6]	CVPR20		✔		✔	✔	
	SiamCAR [17]	CVPR20		✔		✔	✔	
	SiamRCNN [29]	CVPR20	✔	✔	✔	✔	✔	
	CLNet [11]	ECCV20		✔		✔	✔	
	DCFST [41]	ECCV20	✔	✔		✔		
	KYS [2]	ECCV20	✔	✔		✔		
	Ocean [42]	ECCV20		✔		✔		
	PG-Net [21]	ECCV20		✔				
	SiamFC++ [34]	AAAI20		✔				
	SiamGAT [16]	CVPR21		✔		✔	✔	
	SiamRN [7]	CVPR21		✔		✔	✔	
	STMTrack [14]	CVPR21		✔				
	AutoMatch [39]	ICCV21		✔		✔		
	KeepTrack [26]	ICCV21	✔	✔		✔	✔	
	SAOT [43]	ICCV21	✔	✔	✔	✔		
	RTS [27]	ECCV22	✔	✔	✔	✔		✔
Transformer	TransT [5]	CVPR21	✔	✔		✔		
	TrDiMP [30]	CVPR21	✔	✔	✔	✔		
	STARK [35]	ICCV21	✔	✔	✔	✔		
	CSWinTT [28]	CVPR22	✔	✔	✔	✔		
	MixFormer [8]	CVPR22	✔	✔	✔	✔		
	SBT [33]	CVPR22	✔	✔		✔		
	ToMP [25]	CVPR22	✔	✔	✔	✔		
	UTT [24]	CVPR22	✔	✔		✔		
	SwinTrack [22]	NIPS22		✔				
	AiATrack [15]	ECCV22	✔	✔	✔	✔		
	OSTrack [37]	ECCV22	✔	✔	✔	✔		
	SimTrack [3]	ECCV22	✔	✔	✔	✔		
Amount	40		21	40	12	34	13	1

C. Analysis of hyper-parameters

To demonstrate the impact of different hyper-parameters on our approaches, we conduct a series of ablation studies with different hyper-parameter settings, shown as Fig. 2. Considering that the results can be used as the basis for setting parameters in future work, we apply our augmentations on the baseline step by step, in the order of dynamic selection of search radius factor γ, generating boundary samples, and synthesizing hard samples by our TFMix. As a reference, the performance of the baseline tracker OSTrack256 is plotted as the dotted auxiliary lines in Fig. 2.

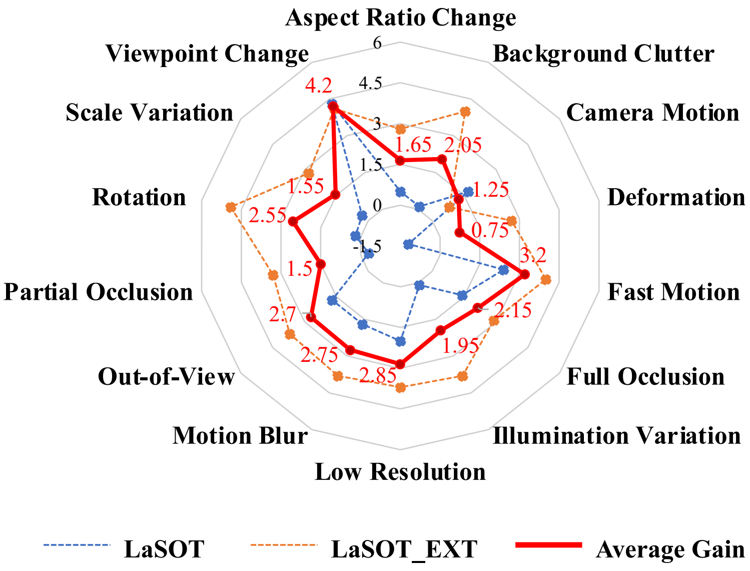
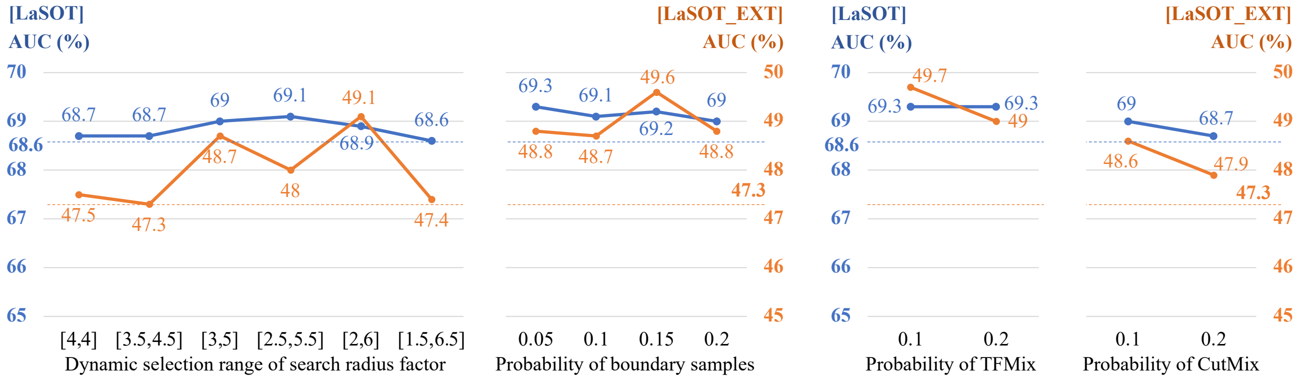


Figure 1. **Performance gain on different attributes.**



(a) Robustness of hyper-parameters in ORC

(b) Comparison of our TFMix and CutMix

Figure 2. **Robustness for hyper-parameters of our approaches.** Dotted auxiliary lines indicate the performance of the baseline tracker.

Parameters in ORC. We set different dynamic selection ranges of search radius factor, and probability of boundary samples for our optimized random cropping (ORC), respectively. As shown in Fig. 2 (a), our dynamic γ mechanism is able to obtain performance gain when we set an appropriate selection range, which is symmetric to the value of γ in the inference, *e.g*. [3,5] to [2,6]. In addition, our ORC also keeps superior performance under different probability settings of boundary samples compared with the baseline.

Comparison of TFMix and CutMix. To avoid the risk of model drifting by the synthetic hard samples, we evaluate TFMix under two relatively low probabilities, and also image-level bbox mixing (CutMix) for comparison. As shown in Fig. 2 (b), our TFMix performs superior to the image-level CutMix under different probability settings. Besides, setting a lower probability (like 0.1) for these synthetic hard samples seems to be able to help train a more robust model.

D. Analysis of limitations

D.1. Generalization of TFMix on CNN backbones

Settings of experiments on CNN backbones. To further demonstrate the generalization ability of our methods, we apply the proposed data augmentations to a hybrid CNN-Transformer-based tracker (STARK with ResNet50), and a CNN-based tracker (SiamFC++). For our optimized random cropping (ORC), we apply it to STARK using the same settings as

Table 3. **Quantitative analysis of performance bias**

		LaSOT		LaSOT_EXT		UAV123		NFS		ALL	
		Num	Gain	Num	Gain	Num	Gain	Num	Gain	Num	Gain
OSTrack256	①	51	+8.1	82	+5.6	21	+16.0	17	+8.8	171	+7.9
	②	229	-1.2	68	-1.1	102	-0.2	83	-1.1	482	-1.0
OSTrack384	①	38	+4.4	72	+2.8	17	+5.8	15	+3.1	142	+3.6
	②	242	-0.3	78	-0.1	106	-0.9	85	-1.5	511	-0.6
MixFormer	①	51	+2.2	75	+4.0	18	+2.0	16	+5.1	160	+3.3
	②	229	-0.5	75	-0.0	105	-0.4	84	-0.1	493	-0.3

MixFormer, where the dynamic search factor range is set to [4.0, 6.0], and the probability of simulating boundary samples is set to 0.05. While for SiamFC++, the parameter used to control the scope of the context is turned from a fixed value (0.5) to a dynamic range ([0.3,0.7]). The probability of boundary samples is also set to 0.05.

For the token-level feature mixing (TFMix), we take STARK as an example, and explore the effect of our TFMix on CNN-based backbones. Specifically, we transfer the feature vectors belonging to the distractor to the corresponding positions on search feature maps extracted from the last layer of the CNN backbone.

Analysis of TFMix. As we claimed in the main paper, since our TFMix is customized based on the characteristics of transformer models, *i.e.* global correlation between independent tokens, it shows to be less effective for CNN-based backbones. The potential reason might be the strong inductive bias in CNN networks.

First, the main reason might be the locality of CNN networks. In feature maps extracted from CNN networks, important target information is contained on a wide range of spatial vectors, instead of on the only pixels belonging to the target area. Replacing several pixels in the search feature map with distractor vectors will not only bring an incomplete representation of the distractor, but also miss part of the target information, which might affect target representations from CNN backbones.

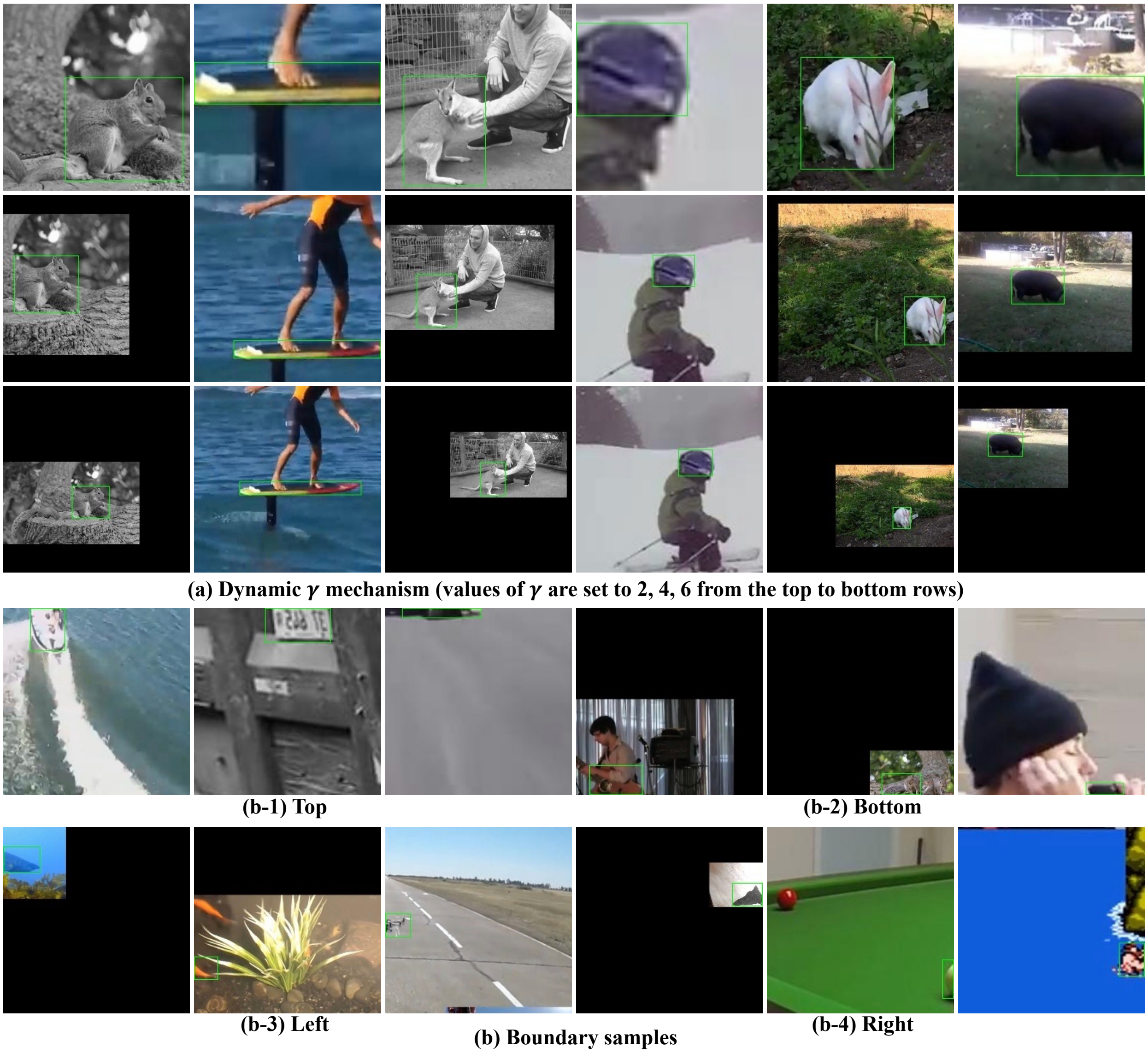
Besides, because of the shift invariance in CNN networks, trackers with CNN backbones usually distinguish objects and other distractors in the late feature fusion stage, instead of the feature extraction stage. This type of tracking framework weakens the role of our TFMix to some extent in improving the discriminative ability of the model to represent different objects. In contrast, the pure transformer models usually merge the feature extraction and feature fusion stages. Our TFMix simulates training samples with background interference by introducing another object (*i.e.* distractor) in the search area, which provides the model more chances during training to learn discriminative representations for different objects.

Nevertheless, due to the significant superiority of pure transformer frameworks, many trackers [4, 32] with Transformer backbones have emerged in the past two years. Therefore, the applicability of our TFMix is still considerable.

D.2. Performance bias on different benchmarks and models

As we mentioned in the main paper ("Limitation"), our methods tend to improve models in terms of tracking robustness, instead of accuracy. Therefore, the advantages of our methods are more evident under challenging settings, like small resolution, or facing unseen categories and challenging situations. Being more effective on hard cases also potentially causes a performance bias of our method on different trackers and benchmarks.

To evidence this conjecture, we dynamically separate each benchmark as two subsets based on the baseline trackers' performance. Subset ① includes challenging sequences where the success plots (AUC) of the baseline trackers is lower than 50.0, while the subset ② includes relatively simple sequences where AUC of not less than 50.0 can be obtained by baseline trackers. We record the sequence numbers (Num) of each subset in each benchmark, and corresponding AUC gains (Gain) brought from our DATr, shown as Tab. 3. Although the performance gains from our DATr show a bias on different benchmarks, our DATr performs consistently on the same subset, *i.e.* improving challenging sequences (subset ①), instead of simple cases (subset ②). However, due to the different proportions of challenge sequences in each benchmark, and also the different tracking robustness of baseline trackers, the overall performance evaluation shows a bias on different benchmarks. For example, even though there is a slight decline for MixFormer on LaSOT and UAV123, our method still improves the tracking robustness for those challenging sequences inside benchmarks.



(a) Dynamic γ mechanism (values of γ are set to 2, 4, 6 from the top to bottom rows)

(b-1) Top

(b-2) Bottom

(b-3) Left

(b-4) Right

(b) Boundary samples

Figure 3. **Examples of diverse samples generated by our optimized random cropping.** (a): We set different values of the search radius factor γ for the same image to visualize the variety that can be brought by our dynamic γ mechanism. (b): Visualization of boundary samples with different directions.

E. Additional Visualizations

E.1. Visualization of diverse samples produced by OCR

We visualize several training samples generated by our optimized random cropping (OCR) in Fig. 3. First, Fig. 3 (a) shows that our dynamic γ mechanism is able to enrich training samples from the perspective of contextual information. Second, Fig. 3 (b) visualizes boundary samples with different directions. Models are able to be more flexible to boundary targets due to such boundary training samples. Compared with the existing random cropping, which has a fixed search radius (see the second row in Fig. 3 (a)), our OCR could crop samples with a more diverse context scope, object scale, and object position.

E.2. Qualitative comparison with state-of-the-art trackers

We visualize several qualitative results as shown in Fig. 4. It can be demonstrated that our models are more robust to challenging situations, like severe background interference (see the 1-st, 2-nd, 5-th, and 6-th rows), fast motion (see the 3-th,

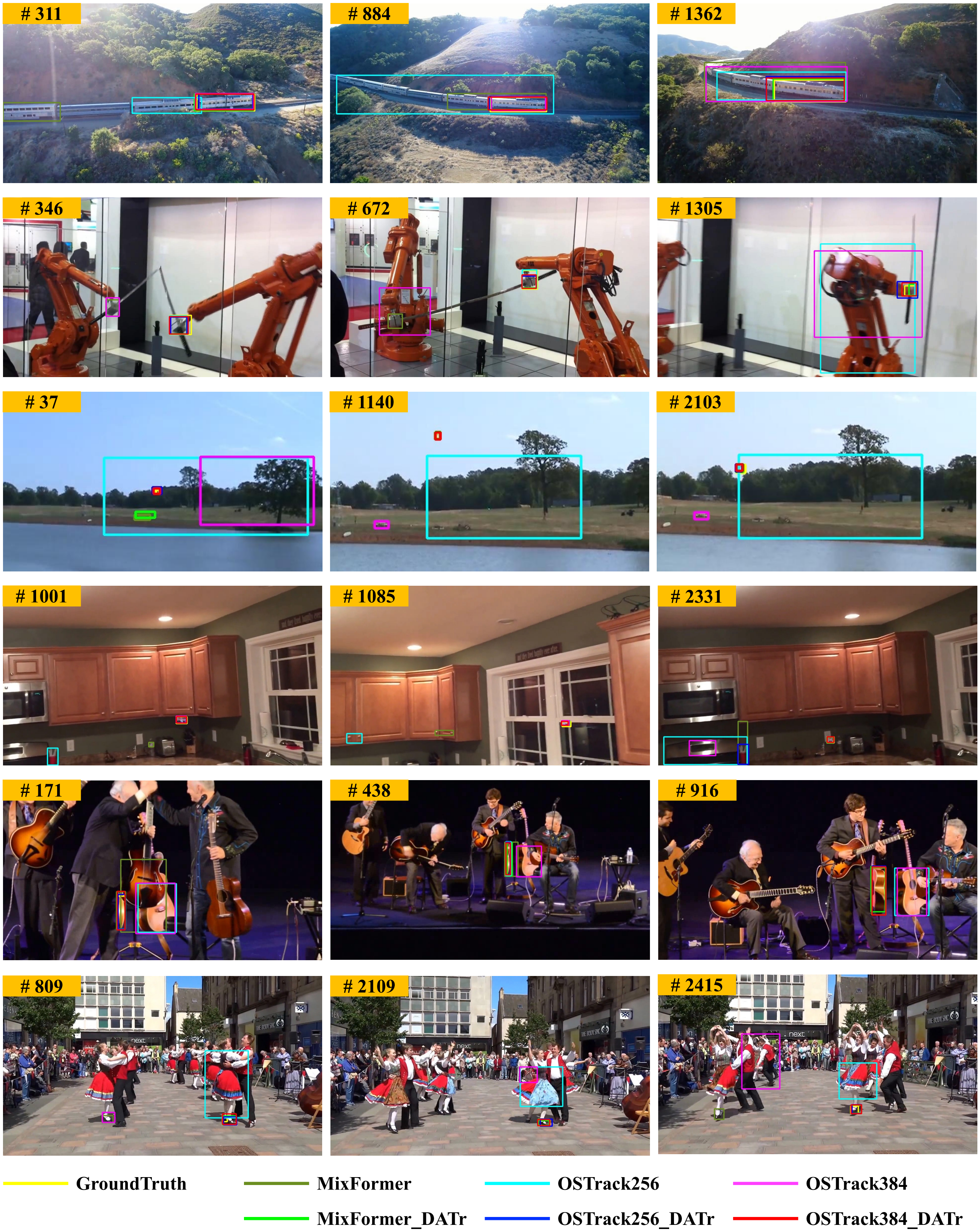


Figure 4. **Qualitative comparison with state-of-the-art trackers.** "_DATr" indicates our method.

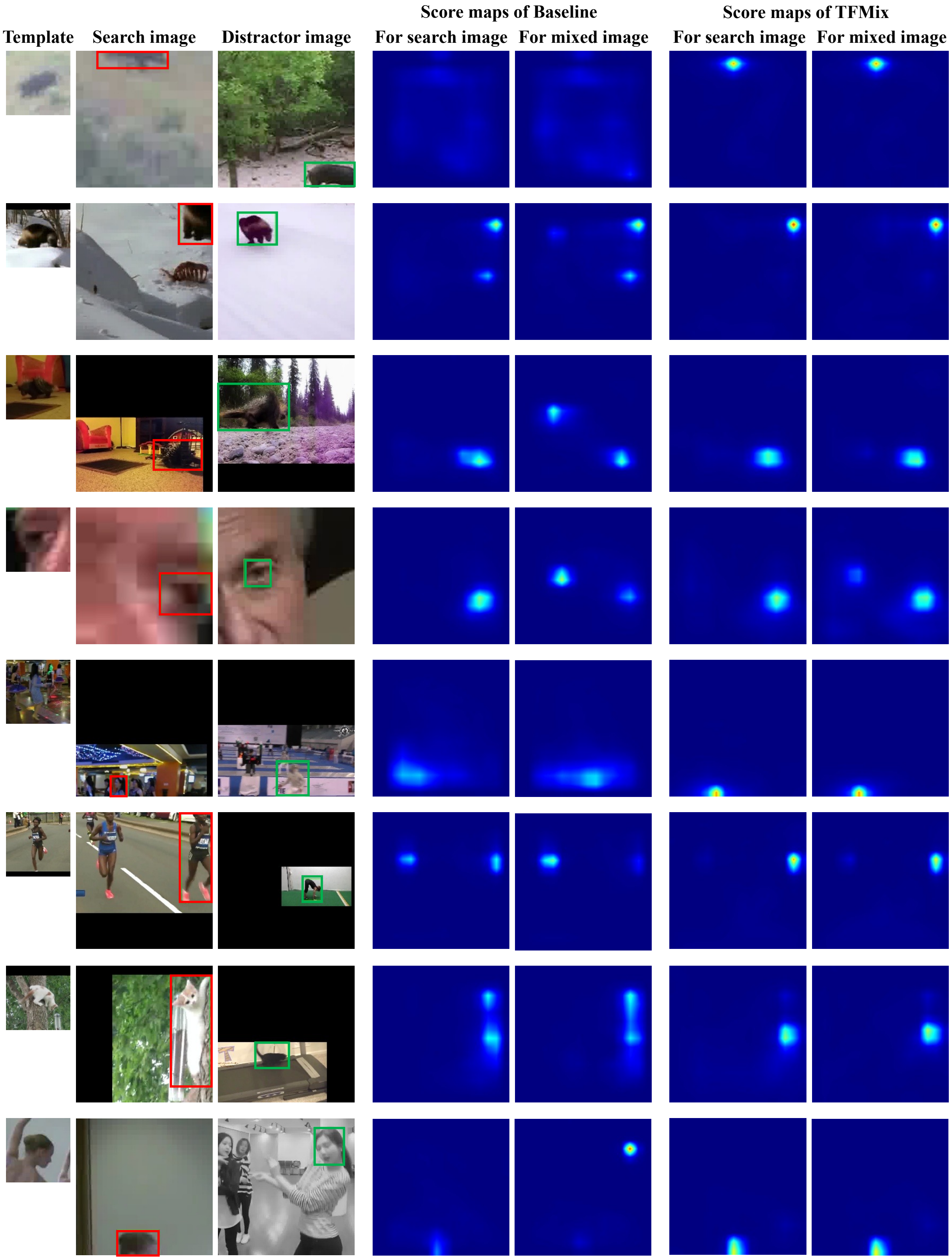


Figure 5. **Visualization of discriminative ability brought by our TFMix.** Objects (framed by red bounding boxes) in the search images are to be tracked, while objects (framed by green bounding boxes) in the distractor images are used to be mixed into the search patches as distractors. Score maps for search images represent that the model processes the original search image, while score maps for mixed images represent that the model processes the mixed search patch with distractors.

4-th, and 6-th rows), and frequent occlusion (see the 1-st and 5-th rows).

In addition, we also provide a video ("qualitative_results.avi" in the zip file) to show more qualitative comparisons on three challenging sequences, which include challenges like scale variation, occlusion, fast motion, and distractors. Compared with baseline trackers, we can see that our models perform more robustly in the face of extremely challenging situations.

E.3. Comparison of discriminative ability

In addition to the "Fig. 5" shown in the main paper, we visualize more examples here to show the comparison of discriminative ability between the baseline model and our methods DATr. As shown in Fig. 5, our TFMix enables the model responses much weaker for distractors. Besides, when we only compare the score maps for the original search image, we can still conclude that our models are more powerful to deal with some challenging situations, like low resolution (see the first row), distractor (see the 2-nd, 6-th rows), scale variation (see the seventh row), and out-of-view (see the 5-th, 8-th rows).

References

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

[2] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Know your surroundings: Exploiting scene information for object tracking. In *European Conference on Computer Vision*, pages 205–221, 2020. 2

[3] Boyu Chen, Peixia Li, Lei Bai, Lei Qiao, Qiuhong Shen, Bo Li, Weihao Gan, Wei Wu, and Wanli Ouyang. Backbone is all your need: a simplified architecture for visual object tracking. In *European Conference on Computer Vision*, pages 375–392, 2022. 2

[4] Xin Chen, Houwen Peng, Dong Wang, Huchuan Lu, and Han Hu. Seqtrack: Sequence to sequence learning for visual object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14572–14581, 2023. 4

[5] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8126–8135, 2021. 2

[6] Zedu Chen, Bineng Zhong, Guorong Li, Shengping Zhang, and Rongrong Ji. Siamese box adaptive network for visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6668–6677, 2020. 2

[7] Siyuan Cheng, Bineng Zhong, Guorong Li, Xin Liu, Zhenjun Tang, Xianxian Li, and Jing Wang. Learning to filter: Siamese relation network for robust tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4421–4431, 2021. 2

[8] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. MixFormer: End-to-end tracking with iterative mixed attention. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13608–13618, 2022. 2

[9] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ATOM: Accurate tracking by overlap maximization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4660–4669, 2019. 2

[10] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7183–7192, 2020. 2

[11] Xingping Dong, Jianbing Shen, Ling Shao, and Fatih Porikli. Clnet: A compact latent network for fast adjusting siamese trackers. In *European Conference on Computer Vision*, pages 378–395, 2020. 2

[12] Heng Fan, Hexin Bai, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Mingzhen Huang, Juehuan Liu, Yong Xu, et al. LaSOT: A high-quality large-scale single object tracking benchmark. *International Journal of Computer Vision*, 129:439–461, 2021. 1

[13] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5374–5383, 2019. 1

[14] Zhihong Fu, Qingjie Liu, Zehua Fu, and Yunhong Wang. Stmtrack: Template-free visual tracking with space-time memory networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13774–13783, 2021. 2

[15] Shenyuan Gao, Chunluan Zhou, Chao Ma, Xinggang Wang, and Junsong Yuan. AiATrack: Attention in attention for transformer visual tracking. In *European Conference on Computer Vision*, pages 146–164, 2022. 2

[16] Dongyan Guo, Yanyan Shao, Ying Cui, Zhenhua Wang, Liyan Zhang, and Chunhua Shen. Graph attention tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9543–9552, 2021. 2

[17] Dongyan Guo, Jun Wang, Ying Cui, Zhenhua Wang, and Shengyong Chen. SiamCAR: Siamese fully convolutional classification and regression for visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6269–6277, 2020. 2

[18] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. SiamRPN++: Evolution of siamese visual tracking with very deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019. 2

[19] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8971–8980, 2018. 2

[20] Peixia Li, Boyu Chen, Wanli Ouyang, Dong Wang, Xiaoyun Yang, and Huchuan Lu. GradNet: Gradient-guided network for visual object tracking. In *IEEE International Conference on Computer Vision*, pages 6162–6171, 2019. 2

[21] Bingyan Liao, Chenye Wang, Yayun Wang, Yaonong Wang, and Jun Yin. Pg-net: Pixel to global matching network for visual tracking. In *European Conference on Computer Vision*, pages 429–444, 2020. 2

[22] Liting Lin, Heng Fan, Zhipeng Zhang, Yong Xu, and Haibin Ling. SwinTrack: A simple and strong baseline for transformer tracking. In *Advances in Neural Information Processing Systems*, 2022. 2

[23] Alan Lukezic, Jiri Matas, and Matej Kristan. D3s-a discriminative single shot segmentation tracker. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7133–7142, 2020. 2

[24] Fan Ma, Mike Zheng Shou, Linchao Zhu, Haoqi Fan, Yilei Xu, Yi Yang, and Zhicheng Yan. Unified transformer tracker for object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8781–8790, 2022. 2

[25] Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8731–8740, 2022. 2

[26] Christoph Mayer, Martin Danelljan, Danda Pani Paudel, and Luc Van Gool. Learning target candidate association to keep track of what not to track. In *IEEE International Conference on Computer Vision*, pages 13444–13454, 2021. 2

[27] Matthieu Paul, Martin Danelljan, Christoph Mayer, and Luc Van Gool. Robust visual tracking by segmentation. In *European Conference on Computer Vision*, pages 571–588, 2022. 2

[28] Zikai Song, Junqing Yu, Yi-Ping Phoebe Chen, and Wei Yang. Transformer tracking with cyclic shifting window attention. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8791–8800, 2022. 2

[29] Paul Voigtlaender, Jonathon Luiten, Philip HS Torr, and Bastian Leibe. Siam R-CNN: Visual tracking by re-detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6578–6588, 2020. 2

[30] Ning Wang, Wengang Zhou, Jie Wang, and Houqiang Li. Transformer meets tracker: Exploiting temporal context for robust visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1571–1580, 2021. 2

[31] Qiang Wang, Li Zhang, Luca Bertinetto, Weiming Hu, and Philip HS Torr. Fast online object tracking and segmentation: A unifying approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1328–1338, 2019. 2

[32] Xing Wei, Yifan Bai, Yongchao Zheng, Dahu Shi, and Yihong Gong. Autoregressive visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9697–9706, 2023. 4

[33] Fei Xie, Chunyu Wang, Guangting Wang, Yue Cao, Wankou Yang, and Wenjun Zeng. Correlation-aware deep tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8751–8760, 2022. 2

[34] Yinda Xu, Zeyu Wang, Zuoxin Li, Ye Yuan, and Gang Yu. SiamFC++: Towards robust and accurate visual tracking with target estimation guidelines. In *AAAI Conference on Artificial Intelligence*, volume 34, pages 12549–12556, 2020. 2

[35] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *IEEE International Conference on Computer Vision*, pages 10448–10457, 2021. 2

[36] Tianyu Yang, Pengfei Xu, Runbo Hu, Hua Chai, and Antoni B Chan. Roam: Recurrently optimizing tracking model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6718–6727, 2020. 2

[37] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision*, pages 341–357, 2022. 1, 2

[38] Yuechen Yu, Yilei Xiong, Weilin Huang, and Matthew R Scott. Deformable siamese attention networks for visual object tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6728–6737, 2020. 2

[39] Zhipeng Zhang, Yihao Liu, Xiao Wang, Bing Li, and Weiming Hu. Learn to match: Automatic matching network design for visual tracking. In *IEEE International Conference on Computer Vision*, pages 13339–13348, 2021. 2

[40] Zhipeng Zhang and Houwen Peng. Deeper and wider siamese networks for real-time visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4591–4600, 2019. 2

[41] Linyu Zheng, Ming Tang, Yingying Chen, Jinqiao Wang, and Hanqing Lu. Learning feature embeddings for discriminant model based tracking. In *European Conference on Computer Vision*, pages 759–775, 2020. 2

[42] Jianlong Fu Bing Li Weiming Hu Zhipeng Zhang, Houwen Peng. Ocean: Object-aware anchor-free tracking. In *European Conference on Computer Vision*, pages 771–787, 2020. 2

[43] Zikun Zhou, Wenjie Pei, Xin Li, Hongpeng Wang, Feng Zheng, and Zhenyu He. Saliency-associated object tracking. In *IEEE International Conference on Computer Vision*, pages 9866–9875, 2021. 2

[44] Zheng Zhu, Qiang Wang, Bo Li, Wei Wu, Junjie Yan, and Weiming Hu. Distractor-aware siamese networks for visual object tracking. In *European Conference on Computer Vision*, pages 101–117, 2018. 2