Learning Occlusion-Robust Vision Transformers for Real-Time UAV Tracking (Supplementary Material)

You Wu[1], Xucheng Wang[2], Xiangyang Yang[1], Mengyuan Liu[1], Dan Zeng[3],
Hengzhou Ye[1], Shuiwang Li[1]
[1]College of Computer Science and Engineering, Guilin University of Technology, China
[2]School of Computer Science, Fudan University, Shanghai, China
[3]School of Artificial Intelligence, Sun Yat-sen University, Zhuhai, China
wuyou@glut.edu.cn, xcwang317@glut.edu.cn, xyyang317@163.com, mengyuaner1122@foxmail.com
zengd8@mail.sysu.edu.cn, yehengzhou@glut.edu.cn, lishuiwang0721@163.com

1. Further experiments

Table 1. Comparison of precision (Prec.), success rate (Succ.), and inference speed on NVIDIA Jetson AGX Xavier edge device (AGX.FPS) between ORTrack-DeiT and the ten lightweight SOTA trackers on BioDrone [50] dataset.

Trcker	Source	BioDrone		AGX.FPS
		Prec.	Succ.	
ORTrack-DeiT	**Ours**	**35.6**	**30.8**	**38.1**
AVTrack-DeiT[30]	ICML 24	33.8	29.1	**40.7**
LightFC[31]	KBS 24	31.8	29.1	34.5
SMAT[21]	WACV 24	28.3	24.9	33.2
PRL-Track[17]	IROS 24	23.0	19.4	33.7
Aba-ViTrack[27]	ICCV 23	**35.9**	**31.3**	34.3
HiT[24]	ICCV 23	**34.0**	**29.9**	**36.8**
SGDViT[44]	ICRA 23	20.2	17.2	31.7
TCTrack++[6]	TPAMI 23	25.0	20.9	28.2
TCTrack[5]	CVPR 22	23.8	20.3	34.1
HiFT[4]	ICCV 21	21.4	17.8	35.2

1.1. Comparison on BioDrone

We have conducted an additional comparison of our ORTrack-DeiT with 10 state-of-the-art lightweight trackers on the BioDrone [50] benchmark. The results are illustrated in Table 1. In terms of tracking performance, our ORTrack-DeiT demonstrates precision and success rates comparable to the best tracker, Aba-ViTrack, with only minor differences of 0.3% and 0.5%, respectively. Regarding efficiency, ORTrack-DeiT ranks as the second-fastest with a speed of 38.1 AGX.FPS, which achieves speed improvements of 11% over Aba-ViTrack. Although it is slightly behind AVTrack-DeiT, which achieves 40.7 AGX.FPS, ORTrack-DeiT outperforms AVTrack-DeiT in precision and success rates by 1.8% and 1.7%, respectively. These results further underscore the effectiveness of our approach.

Table 2. Attribute-based comparison on the occlusion challenge subsets of LaSOT [16] and OTB100 [42].

Trcker	Source	LaSOT		OTB100	
		Prec.	Succ.	Prec.	Succ.
ORTrack-DeiT	**Ours**	**60.4**	**54.2**	**75.9**	**64.9**
AVTrack-DeiT[30]	ICML 24	**57.8**	**52.6**	**75.6**	**64.4**
LightFC[31]	KBS 24	54.8	49.7	**75.4**	**64.6**
SMAT[21]	WACV 24	56.7	51.7	74.6	63.9
PRL-Track[17]	IROS 24	38.1	37.3	64.8	58.8
Aba-ViTrack[27]	ICCV 23	**60.0**	**53.9**	74.1	63.6
HiT[24]	ICCV 23	53.5	50.5	64.6	57.2
SGDViT[44]	ICRA 23	38.4	37.0	64.7	57.8
TCTrack++[6]	TPAMI 23	41.3	39.1	69.6	61.1
TCTrack[5]	CVPR 22	39.8	35.3	65.5	56.9
HiFT[4]	ICCV 21	33.2	33.7	62.1	56.1

1.2. Extension on general object tracking

We have performed additional experiments on the subsets of LaSOT [16] and OTB100 [42] that involve occlusion challenges, to evaluate the generalization applicability of our method. As shown in Table 2, ten SOTA lightweight trackers are compared with our ORTrack-DeiT tracker. Obviously, our ORTrack-DeiT outperforms all trackers in both precision and success rate, further highlighting the effectiveness of our approach.

2. Supplementary comprehensive experimental results to the main text

2.1. Comparison with Deep Trackers

Due to page length constraints in the main paper, we limited our comparison to our approach and state-of-the-art (SOTA) deep trackers using the VisDrone2018 [52] dataset. In this section, we compare our ORTrack-DeiT with SOTA deep trackers using more UAV tracking datasets featured in our main paper. These additional datasets include DTB70 [28], UAVDT [15], and UAV123 [34]. The evaluation re-

Table 3. The comparison of the precision (Prec.), success rate (Succ.), and speed (FPS) of deep-based trackers on DTB70 [28], UAVDT [15], VisDrone2018 [52], and UAV123 [34] with ORTrack-DeiT. The top three results are displayed in **red**, **blue** and **green** fonts.

Trcker	Source	DTB70		UAVDT		VisDrone2018		UAV123		Avg.		Avg.FPS
		Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	
ORTrack-DeiT	**Ours**	86.2	66.4	83.4	60.1	**88.6**	**66.8**	84.3	66.4	85.6	65.0	**226.4**
AQATrack[43]	CVPR 24	**86.3**	66.2	**84.7**	63.7	**87.2**	**66.9**	**89.5**	**70.6**	**86.9**	**66.9**	60.5
HIPTrack[2]	CVPR 24	**88.4**	**68.6**	79.6	60.9	**86.7**	**67.1**	**89.2**	**70.5**	**86.0**	**66.8**	35.6
EVPTrack[36]	AAAI 24	85.8	66.5	80.6	61.2	84.5	65.8	**88.9**	**70.2**	85.0	65.9	25.3
ROMTrack[3]	ICCV 23	**87.2**	**67.4**	81.9	61.6	86.4	66.7	87.4	69.2	**85.7**	**66.2**	53.6
ZoomTrack[26]	NIPS 23	82.0	63.2	77.1	57.9	81.4	63.4	88.4	69.6	82.2	63.5	64.3
SeqTrack[8]	CVPR 23	85.7	65.6	79.0	59.8	85.3	65.8	86.8	68.6	84.2	65.0	17.6
MAT[49]	CVPR 23	83.2	64.5	72.9	54.8	81.6	62.2	86.7	68.3	81.1	62.5	72.3
SparseTT[18]	IJCAI 22	82.3	65.8	82.8	65.4	81.4	62.1	85.4	68.8	83.0	65.5	32.8
OSTrack[45]	ECCV 22	82.7	65.1	**85.0**	**67.2**	84.2	64.8	87.2	68.9	84.8	66.5	65.8
SimTrack[7]	ECCV 22	83.2	64.6	76.5	57.2	80.0	60.9	88.2	69.2	81.9	62.9	**73.1**
ToMP[33]	CVPR 22	85.6	**67.1**	**85.4**	**64.1**	84.1	64.4	82.6	65.9	84.4	65.4	24.3
AutoMatch[48]	ICCV 21	82.5	63.4	82.1	60.8	78.1	59.6	78.5	60.7	80.3	61.1	64.6
KeepTrack[32]	ICCV 21	83.6	64.3	83.8	60.5	84.0	63.5	85.9	67.3	84.3	63.9	22.1
SAOT[51]	ICCV 21	83.1	64.6	82.1	60.7	76.9	59.1	82.7	64.9	81.2	62.3	37.2
TranT[9]	CVPR 21	83.6	65.8	82.6	63.2	85.9	65.2	85.0	67.1	84.3	65.3	53.2
TrDiMP[38]	CVPR 21	82.4	63.9	**88.2**	**64.5**	84.1	63.1	84.0	66.3	84.7	64.5	31.6
PrDiMP50[11]	CVPR 20	76.4	59.5	82.7	60.1	79.4	59.7	85.7	67.7	81.1	61.8	45.5
DiMP50[1]	ICCV 19	79.2	61.3	78.3	57.4	83.5	63.0	83.1	65.2	81.0	61.7	**71.9**

Table 4. Effect of ORR and AFKD on the performance of the baseline trackers.

Method	ORR	AFKD	DTB70		UAVDT		VisDrone2018		UAV123		Avg.		Avg.FPS
			Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	
ORTrack-ViT			79.3	62.4	77.0	55.6	83.0	62.7	83.2	66.5	80.6	61.8	212.2
	✓		**83.2**	**64.5**	**80.3**	**58.2**	**84.6**	**63.6**	**84.1**	**65.9**	**83.1**$_{\uparrow 2.5}$	**63.1**$_{\uparrow 1.5}$	-
	✓	✓	81.8	63.8	79.1	57.5	84.1	63.3	82.5	65.3	81.9$_{\uparrow 1.3}$	62.5$_{\uparrow 0.7}$	**270.6**$_{\uparrow 27\%}$
ORTrack-Eva			80.5	62.8	78.1	56.6	79.7	60.7	80.4	63.2	79.7	60.8	238.1
	✓		**81.6**	**63.3**	**80.8**	**58.7**	**84.5**	**63.6**	**82.3**	**64.7**	**82.3**$_{\uparrow 2.6}$	**62.5**$_{\uparrow 1.7}$	-
	✓	✓	81.1	62.8	79.5	57.8	81.8	62.3	81.5	64.4	80.8$_{\uparrow 1.1}$	61.6$_{\uparrow 0.8}$	**301.2** $_{\uparrow 26\%}$
ORTrack-DeiT			84.2	65.1	78.6	56.7	81.6	62.2	83.7	66.1	82.0	62.5	226.4
	✓		**86.2**	**66.4**	**83.4**	**60.1**	**88.6**	**66.8**	**84.3**	**66.4**	**85.6** $_{\uparrow 3.6}$	**65.0**$_{\uparrow 2.5}$	-
	✓	✓	83.7	65.1	82.5	59.7	84.6	63.9	84.0	66.1	83.7$_{\uparrow 1.7}$	63.7$_{\uparrow 1.2}$	**292.3** $_{\uparrow 29\%}$

sults for ORTrack-DeiT and the competing deep trackers are presented in Table 3. As demonstrated, our ORTrack-DeiT is the fastest tracker and it is above three times the speed of the second fastest tracker DiMP50 [1]. It can be seen that there is no tracker can achieve the highest Prec. and Succ. on all datasets. Remarkably, our ORTrack-DeiT achieves the first place of Prec. and the third place of Succ. on VisDrone2018. Although the proposed method was not targeted at generic visual tracking, its performance on these UAV tracking datasets is able to beat many deep trackers. It is very impressive considering the significantly high efficiency our method achieves, i.e., 226.4 FPS, while all other deep trackers are under 100.0 FPS. These results suggest that our method is able to strike a better balance for UAV tracking.

2.2. Effect of Occlusion-Robust Representations (ORR) and Adaptive Feature-Based Knowledge Distillation (AFKD)

Table 4 presents comprehensive results on the impact of the ORR and AFKD, evaluated across four datasets. To avoid potential variations due to randomness, we only present the speed of the baseline, since the GPU speeds of the baseline and its ORR-enhanced version are theoretically identical. As can be seen, the incorporation of ORR significantly enhances both Prec. and Succ. for all baseline trackers. Specifically, the Avg.Prec. increases for ORTrack-ViT, ORTrack-Eva, and ORTrack-DeiT are 2.5%, 2.6%, and 3.6%, respectively, while the Avg.Succ. increases are 1.5%, 1.7%, and 2.5%, respectively. These significant enhancements highlight the effectiveness of ORR in improving tracking precision. The further integration of AFKD results in consistent improvements in GPU speeds, with only slight reductions in Prec. and Succ. Specifically, all baseline trackers experience GPU speed enhancements of over 26.0%, with ORTrack-DeiT showing an impressive 29.0% improvement. These results affirm the effectiveness of AFKD in optimizing tracking efficiency while maintaining high tracking performance.

2.3. Impact of Masking Operators

Table 5 provides comprehensive results of ORTrack-DeiT with various implementations of masking operators (i.e., $\mathfrak{m}_{\mathrm{U}}$, $\mathfrak{m}_{\mathrm{C}}$, and SAM [25]) alongside data mixing augmentation methods (i.e., AdAutoMix [35] and CutMix [46]) on the performance, evaluated across four datasets. As

Table 5. Effect of Masking Operators on the performance on DTB70, UAVDT, VisDrone2018, and UAV123.

Trcker	$\mathfrak{m}_U$	$\mathfrak{m}_C$	SAM[25]	AdAutoMix[35]	CutMix[46]	DTB70		UAVDT		VisDrone2018		UAV123		Avg.	
						Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
ORTrack-DeiT						84.2	65.1	78.6	56.7	81.6	62.2	83.7	66.1	82.0	62.5
	✓					84.3	65.1	**83.9**	**60.6**	86.7	65.4	83.2	65.3	$84.5_{\uparrow 2.5}$	$64.1_{\uparrow 1.6}$
		✓				**86.2**	**66.4**	83.4	60.1	**88.6**	**66.8**	84.3	66.4	$\mathbf{85.6}_{\uparrow 3.6}$	$\mathbf{65.0}_{\uparrow 2.5}$
			✓			85.0	65.5	82.8	59.6	86.8	65.6	82.7	64.8	$84.3_{\uparrow 2.3}$	$63.8_{\uparrow 1.3}$
				✓		83.6	64.7	81.7	58.5	84.3	63.8	83.2	65.8	$83.2_{\uparrow 1.2}$	$63.2_{\uparrow 0.7}$
					✓	81.7	63.1	79.5	57.7	85.7	64.2	**84.6**	**66.5**	$82.8_{\uparrow 0.8}$	$62.9_{\uparrow 0.4}$

shown, on average, while using SAM, AdAutoMix, and CutMix improve performance, the best result achieved with SAM is only comparable to our $\mathfrak{m}_U$ masking operator, both achieving gains exceeding 2.0% in Avg.Prec. and 1.0% in Avg.Succ., respectively. When $\mathfrak{m}_C$ is applied, the improvements are even more substantial, with increases of 3.6% and 2.5%, respectively. These results validate the effectiveness of the proposed ORR component and particularly demonstrate the superiority of the masking operator based on spatial Cox processes.

Table 6. Impact of the masking ratio σ in constructing masked templates with the masking operator based on spatial Cox processes on ORTrack-DeiT.

σ	DTB70		UAVDT		VisDrone2018		UAV123		Avg.	
	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
0.1	83.2	64.4	82.9	59.6	84.5	64.2	82.2	64.7	83.2	63.2
0.2	**84.5**	**65.7**	**84.1**	**60.5**	**86.1**	**65.2**	83.1	65.5	**84.5**	**64.2**
0.3	**86.2**	**66.4**	**83.4**	**60.1**	**88.6**	**66.8**	**84.3**	**66.4**	**85.6**	**65.0**
0.4	**83.6**	**64.6**	81.6	59.1	85.1	64.6	**85.3**	**67.1**	83.9	63.8
0.5	82.9	64.3	83.3	60.3	**87.2**	**65.5**	83.3	65.5	**84.2**	**63.9**
0.6	83.4	64.8	**83.8**	**60.7**	85.1	64.6	81.8	64.3	83.5	63.6
0.7	82.3	63.6	82.6	59.4	83.0	63.0	**85.0**	**66.8**	83.2	63.2
0.8	82.7	64.1	81.2	58.9	84.1	63.9	83.6	65.7	82.9	63.2

2.4. Impact of Masking Ratio in Constructing Masked Templates

To understand the impact of the masking ratio σ in constructing masked templates with the masking operator based on spatial Cox processes on learning occlusion-robust representations, we train ORTrack-DeiT with different settings of σ, ranging from 0.1 to 0.8 with the step of 0.1, and evaluate on four UAV tracking benchmarks. The evaluation results are shown in Table 6. From table, ORTrack-DeiT demonstrates optimal performance with the σ value of 0.3, achieving an Avg.Prec of 85.6% and an Avg.Succ of 65.0%. Experimental results show that the choice of masking ratios significantly impacts tracking performance, with both smaller and larger ratios hindering optimal results.

2.5. Impact of Patch Size in Constructing Masked Templates

To understand the impact of patch size $p \times p$ on learning occlusion-robust ViTs with the proposed masking operator $\mathfrak{m}_C$, we trained ORTrack-DeiT with different p settings, ranging from 8 to 64, doubling each time, and evaluated on four UAV tracking benchmarks. The results are shown in Table 7. As observed, ORTrack-DeiT achieved the best performance when $p = 16$, achieving an average Prec. of 85.6% and Succ. of 65.0%, respectively. However, the optimal patch size varies by dataset: $p = 8$ is best for UAV123, while $p = 32$ is optimal for UAVDT, reflecting that occlusion challenges differ across datasets.

Table 7. Impact of the patch size $p \times p$ in constructing masked templates on ORTrack-DeiT.

$p \times p$	DTB70		UAVDT		VisDrone2018		UAV123		Avg.	
	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
8	85.6	65.9	82.3	59.3	87.5	66.1	**84.7**	**66.5**	85.0	64.4
16	**86.2**	**66.4**	83.4	60.1	**88.6**	**66.8**	84.3	66.4	**85.6**	**65.0**
32	84.8	65.4	**84.3**	**60.9**	85.6	64.3	83.8	66.2	84.6	64.2
64	82.4	63.9	83.8	60.3	84.8	63.8	84.2	66.2	83.8	63.6

Table 8. Ablation study on loss $\mathcal{L}_{orr}$ weighting on DTB70, UAVDT, VisDrone2018, and UAV123 by varying γ from 0.5×10^{-4} to 5×10^{-4}.

γ	DTB70		UAVDT		VisDrone2018		UAV123		Avg.	
	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
0.5	82.2	64.0	82.1	58.7	85.2	64.5	83.7	66.0	83.3	63.3
0.6	**84.8**	65.4	81.9	58.6	85.4	64.8	**84.7**	66.2	84.2	63.8
0.7	**85.1**	**65.8**	**84.2**	**60.5**	86.5	**65.9**	**84.6**	**66.5**	**85.1**	**64.7**
0.8	84.5	65.2	83.8	60.2	85.6	64.9	83.9	65.8	84.5	64.0
0.9	83.1	64.6	82.7	59.3	84.1	63.8	84.5	66.3	83.6	63.5
1.0	84.5	**65.9**	82.9	60.0	**86.6**	65.3	84.2	66.0	84.5	64.3
2.0	**86.2**	**66.4**	83.4	**60.1**	**88.6**	**66.8**	84.3	**66.4**	**85.6**	**65.0**
3.0	84.3	65.0	**84.7**	**60.8**	**87.9**	**66.5**	82.6	65.4	**84.9**	**64.4**
4.0	82.3	63.8	82.4	58.9	85.4	64.7	**85.2**	**66.9**	83.8	63.6
5.0	82.6	64.1	**83.6**	**60.2**	83.8	63.6	83.9	66.1	83.5	63.5

2.6. Impact of Weighting the Loss for Learning Occlusion-Robust Representations Based on Spatial Cox Processes

To obtain the optimal weight γ for the proposed loss that learns occlusion-robust representations based on spatial cox processes, we trained ORTrack-DeiT using varied values of γ ranging from 0.5×10^{-4} to 5×10^{-4} with increments of 0.1×10^{-4}. The evaluation results are presented in Table 8. As shown, our tracker achieves optimal performance when the loss weight (γ) is set to 2.0×10^{-4}. Additionally, we have observed that the second and third-best performances across these datasets are scattered both above and below the

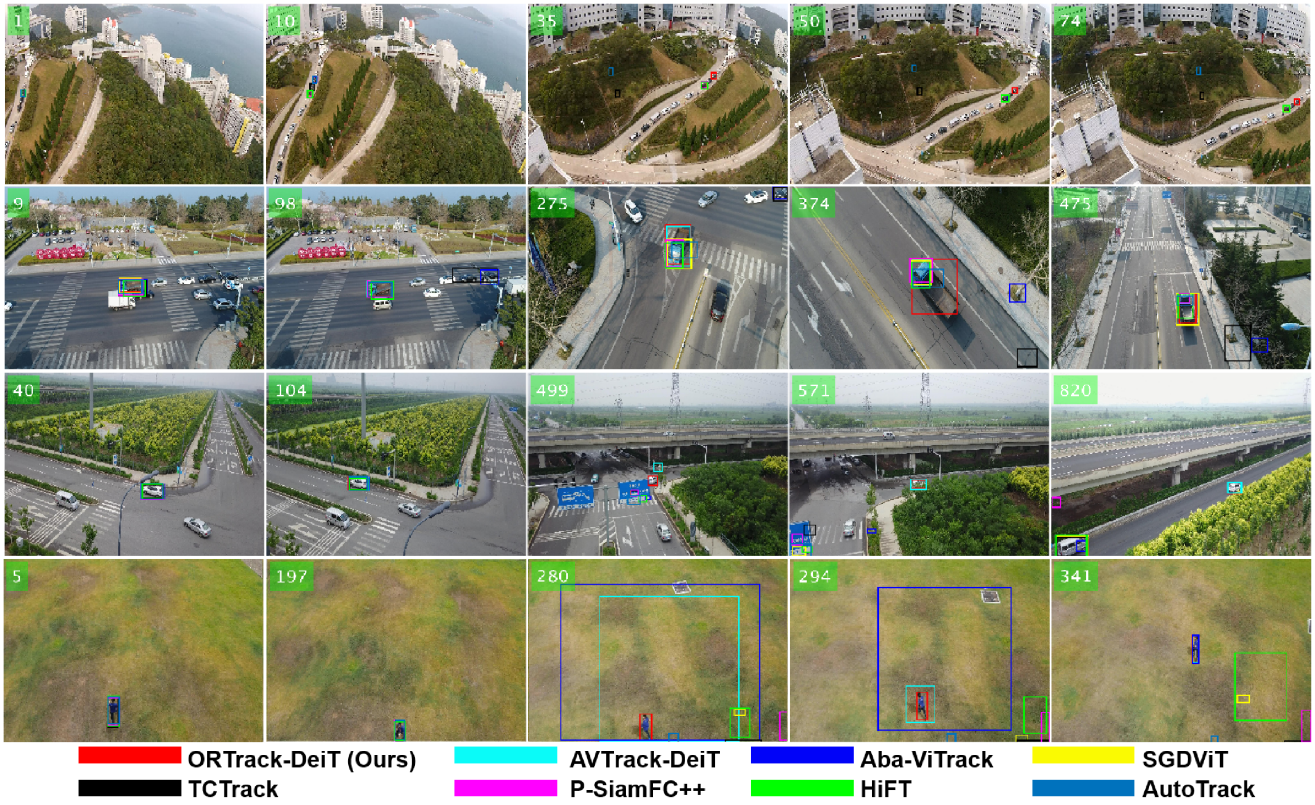

Figure 1. Qualitative evaluation on 4 video sequences from, respectively, DTB70 [28], VisDrone2018 [52], UAVDT [15], and UAV123 [34] (i.e. Car2, S1607, uav0000180_00050_s, and person10).

value of 2.0×10^{-4}, without any apparent patterns. This variation may be attributed to the inherent differences between these datasets. Setting a value of 0.5×10^{-4} for the loss weight (γ) results in an average maximal difference of Prec. of 2.3% and a maximal difference of Succ. of 1.7%. These significant margins clearly demonstrate that the choice of weight a has a considerable impact on the tracking performance. To be more specific, when the proposed loss is appropriately weighted, it can enhance the tracking performance. However, if not properly weighted, it may have detrimental effects on the training of the tracking task.

Table 9. Application of our ORR component to three SOTA trackers: ARTrack [40], GRM [20], and DropTrack[41].

Method	ORR	DTB70		UAVDT		VisDrone2018		UAV123		Avg.	
		Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.	Prec.	Succ.
ARTrack		78.1	59.8	77.1	54.6	77.7	59.5	79.4	60.8	78.3	58.7
	✓	**79.8**	**61.4**	**78.5**	**55.8**	**79.5**	**60.8**	**80.6**	**61.7**	**79.6**	**59.9**
GRM		82.9	64.3	79.0	57.7	82.7	63.4	84.6	66.2	82.3	62.9
	✓	**85.1**	**65.4**	**81.7**	**59.3**	**84.8**	**64.6**	**85.1**	**66.6**	**84.1**	**64.0**
DropTrack		80.7	63.3	76.9	55.9	81.5	62.7	83.3	65.6	80.6	61.9
	✓	**84.3**	**65.1**	**78.7**	**57.4**	**82.8**	**64.2**	**83.6**	**65.9**	**82.3**	**63.1**

2.7. Application to SOTA trackers

To show the wide applicability of our proposed method, we incorporate the proposed ORR into three existing SOTA trackers: ARTrack [40], GRM [20], and DropTrack [41]. Please note that we replace the model's original backbones with ViT-tiny [14] to reduce training time. The evaluation results on four UAV tracking benchmarks are shown in Table 9. As mentioned previously, the GPU speeds of the baseline and its ORR-enhanced version are theoretically identical to eliminate potential nuances arising from randomness. As observed, incorporating ORR results in significant improvements in both precision and success rates for the three baseline trackers. Specifically, ARTrack, GRM, and DropTrack show increases of 1.3%, 1.8%, and 1.7% in Avg.Prec., respectively, while their Avg.Succ. improve by 1.2%, 1.1%, and 1.2%, respectively. These experimental results demonstrate that the proposed ORR component can be seamlessly integrated into existing tracking frameworks, improving tracking accuracy without adding extra computational overhead.

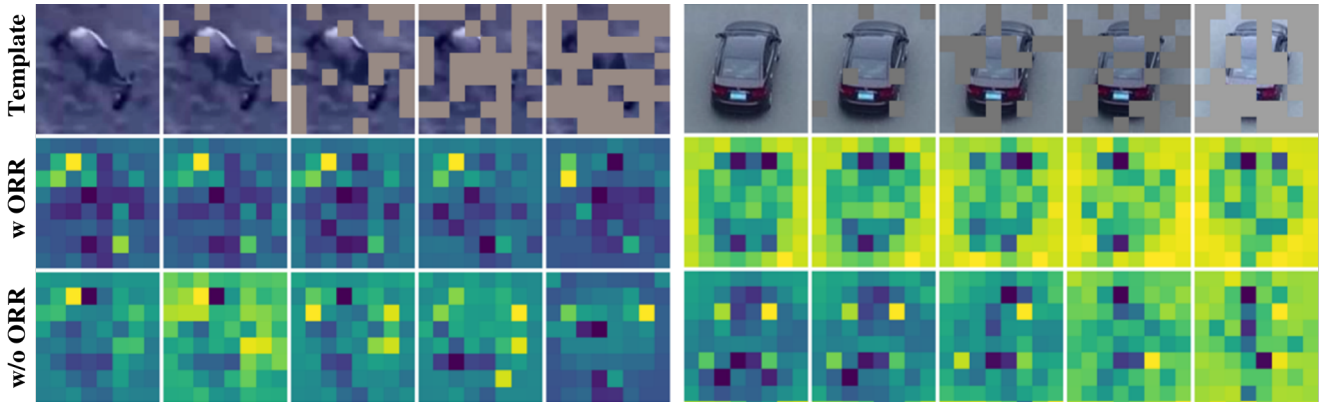

Figure 2. Each group shows the masked image (top), feature map by ORTrack-DeiT with (middle) and without (bottom) ORR. The masking ratios are 0%, 10%, 30%, 50%, and 70%, from left to right.

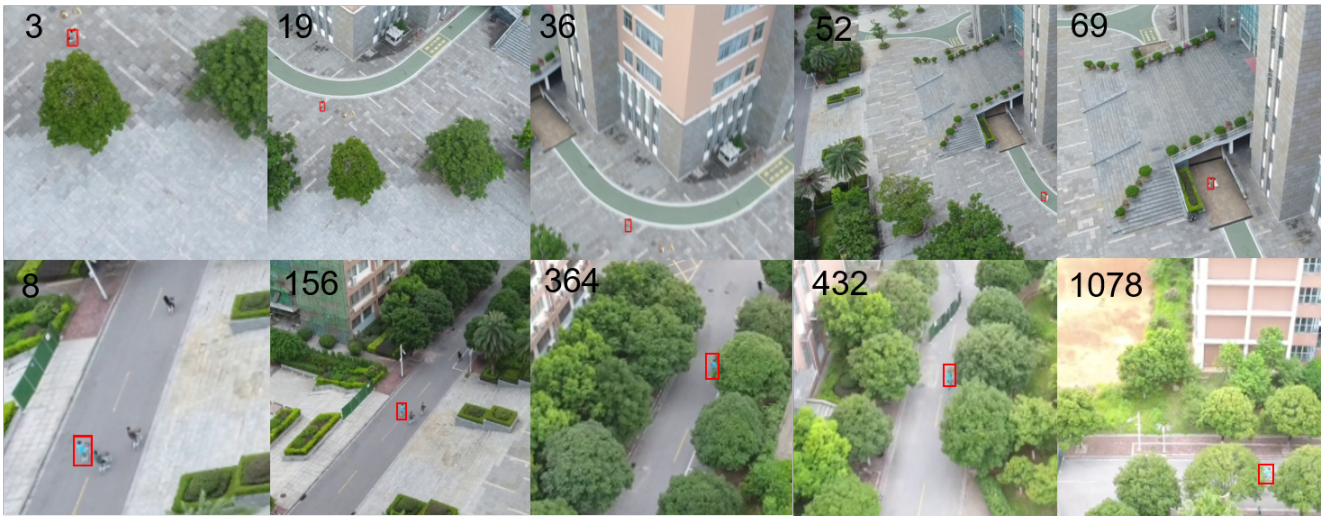

Figure 3. The real data visualization recorded on the UAV platform is visualized, the tracking target has been marked with a red box. Different line representatives perform target tracking in different environments, the frame has been marked in the upper left corner.

2.8. Qualitative Results

Several qualitative tracking results of ORTrack-DeiT and seven SOTA UAV trackers are shown in Fig. 1. As show, only our tracker successfully tracks the targets in all challenging examples, where pose variations (i.e., S1607, uav0000180_00050_s, and person10), background clusters (i.e., all sequences), and scale variations (i.e., uav0000180_00050_s) are presented. In these cases, our method performs significantly better and is more visually appealing, bolstering the effectiveness of the proposed method for UAV tracking.

Fig. 2 visualizes more feature maps produced by ORTrack-DeiT on two samples from UAV123 [34] with and without occlusion-robustness implemented. We can see that feature maps generated by ORTrack-DeiT with occlusion-robustness implemented are more consistent as the masking ratio changes, whereas those generated by ORTrack-DeiT without occlusion-robustness change dramatically, especially at higher masking ratios. These qualitative results provide visual evidence for the effectiveness of our method in learning occlusion robust feature representations with ViTs.

2.9. Real-world Tests

In order to test our method on a real drone, we integrated an embedded onboard processor, the NVIDIA Jetson AGX Xavier 32GB, into a typical UAV platform. During real-world UAV testing, our ORTracker-DeiT and ORTracker-D-DeiT maintained average speeds of 37.5 FPS and 45.3 FPS, respectively, with GPU utilization rates of 42.8% and 37.6%. Two example tracking results are shown in Fig. 3. The first row show a small object with drastic changes in

scale and rapid movement. In the second line of the ultra-long sequence video, the object is tracked with blurred vision and obstructed by trees under sunlight exposure. Real-world testing on embedded systems directly verifies it can still maintain robustness during frequent occlusion and excellent performance and efficiency in various UAV specific challenges.

2.10. Attribute-based Evaluation

To deeply understand the superiority of our trackers over SOTA UAV trackers, we conduct performance evaluations using ORTrack-DeiT against 17 trackers, including KCF [22], STRCF [13], fDSST [12], BACF [19], MCCT_H [37], ECO_HC [10], AutoTrack [29], ARCF [23], HiFT [4], TCTrack [5], P-SiamFC++ [39], SGDViT [44], DRCI [47], ABDNet[53], Aba-ViTrack [27], PRL-Track [17], and AVTrack-DeiT [30], evaluated on the VisDrone2018 nine different attribute subsets. The ORTrack-DeiT exhibits exceptional performance in terms of Prec. and Succ. across most of these attributes. It is worth mentioning that we also evaluate ORTrack-DeiT without employing the proposed components, which we refer to as ORTrack-DeiT* for reference. In Fig. 4, we present the precision plots and success plots for the VisDrone2018 [52] dataset on these attributes.

As shown, with respect to both precision and success plots, ORTrack-DeiT exhibits optimal performance in terms of 'Partial Occlusion', 'Full Occlusion', 'Fast Motion', 'Camera Motion', 'Out-of-View', 'Aspect Ratio Change', and 'Background Cluster'. Remarkably, across these nine attributes, and the integration of our proposed components leads to significant improvements when compared to ORTrack-DeiT*, achieving enhancements of 6.9%, 8.1%, 3.9%, 7.7%, 7.7%, 5.7%, 16.3%, 4.0%, and 20.7% in precision, and 4.4%, 5.6%, 2.1%, 5.0%, 5.2%, 6.7%, 10.6%, 2.4%, and 13.3% in success rate, respectively. These results provide strong evidence for the effectiveness of our method in improving tracking performance.

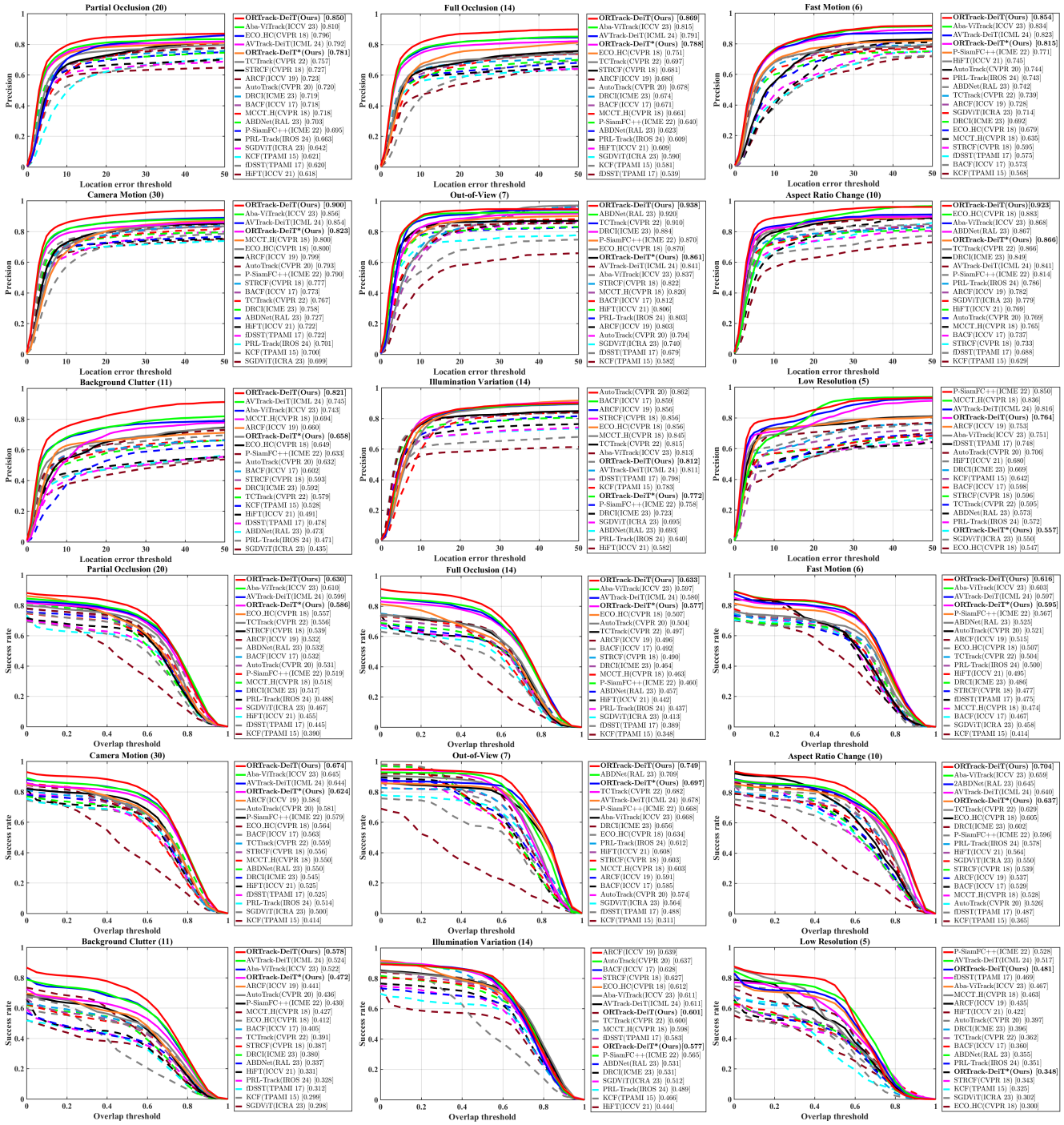

Figure 4. Precision and success rate plots for attribute-based comparisons are shown for the attribute subsets of VisDrone2018. Note that **ORTrack-DeiT*** denotes **ORTrack-DeiT** without the application of the proposed components.

References

[1] 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 (ICCV)*, pages 6182–6191, 2019. 2

[2] Wenrui Cai, Qingjie Liu, and Yunhong Wang. Hiptrack: Visual tracking with historical prompts. In *CVPR*, pages 19258–19267, 2024. 2

[3] Yidong Cai, Jie Liu, Jie Tang, and Gangshan Wu. Robust object modeling for visual tracking. In *ICCV*, pages 9589–9600, 2023. 2

[4] Ziang Cao, Changhong Fu, Junjie Ye, Bowen Li, and Yiming Li. Hift: Hierarchical feature transformer for aerial tracking. In *ICCV*, pages 15457–15466, 2021. 1, 6

[5] Ziang Cao, Ziyuan Huang, Liang Pan, Shiwei Zhang, Ziwei Liu, and Changhong Fu. Tctrack: Temporal contexts for aerial tracking. In *CVPR*, pages 14798–14808, 2022. 1, 6

[6] Ziang Cao, Ziyuan Huang, Liang Pan, Shiwei Zhang, Ziwei Liu, and Changhong Fu. Towards real-world visual tracking with temporal contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[7] Boyu Chen, Peixia Li, Lei Bai, Lei Qiao, and et al. Backbone is all your need: a simplified architecture for visual object tracking. In *ECCV*, pages 375–392, 2022. 2

[8] Xin Chen, Houwen Peng, Dong Wang, Huchuan Lu, and Han Hu. Seqtrack: Sequence to sequence learning for visual object tracking. In *CVPR*, pages 14572–14581, 2023. 2

[9] 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*, pages 8126–8135, 2021. 2

[10] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Eco: Efficient convolution operators for tracking. In *CVPR*, pages 6638–6646, 2017. 6

[11] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *CVPR*, pages 7181–7190, 2020. 2

[12] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and et al. Discriminative scale space tracking. *IEEE TPAMI*, 39(8):1561–1575, 2017. 6

[13] Martin Danelljan, Gustav Häger, Fahad Shahbaz Khan, and Michael Felsberg. Adaptive decontamination of the training set: A unified formulation for discriminative visual tracking. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1430–1438, 2016. 6

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, and et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020. 4

[15] Dawei Du, Yuankai Qi, Hongyang Yu, and et al. The unmanned aerial vehicle benchmark: Object detection and tracking. In *ECCV*, pages 375–391, 2018. 1, 2, 4

[16] Heng Fan, Liting Lin, Fan Yang, and et al. Lasot: A high-quality benchmark for large-scale single object tracking. In *CVPR*, pages 5369–5378, 2018. 1

[17] Changhong Fu, Xiang Lei, and et al. Progressive representation learning for real-time uav tracking. In *IROS*, pages 5072–5079, 2024. 1, 6

[18] Zhihong Fu, Zehua Fu, Qingjie Liu, Wenrui Cai, and Yunhong Wang. Sparsett: Visual tracking with sparse transformers. *arXiv e-prints*, 2022. 2

[19] Hamed Kiani Galoogahi, Ashton Fagg, and Simon Lucey. Learning background-aware correlation filters for visual tracking. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1144–1152, 2017. 6

[20] Shenyuan Gao, Chunluan Zhou, and Jun Zhang. Generalized relation modeling for transformer tracking. In *CVPR*, pages 18686–18695, 2023. 4

[21] Goutam Yelluru Gopal and Maria A Amer. Separable self and mixed attention transformers for efficient object tracking. In *WACV*, pages 6708–6717, 2024. 1

[22] João F. Henriques, Rui Caseiro, Pedro Martins, and et al. High-speed tracking with kernelized correlation filters. *IEEE TPAMI*, 37:583–596, 2015. 6

[23] Ziyuan Huang, Changhong Fu, and et al. Learning aberrance repressed correlation filters for real-time uav tracking. In *ICCV*, pages 2891–2900, 2019. 6

[24] Ben Kang, Xin Chen, D. Wang, Houwen Peng, and Huchuan Lu. Exploring lightweight hierarchical vision transformers for efficient visual tracking. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9578–9587, 2023. 1

[25] Alexander Kirillov, Eric Mintun, and et al. Segment anything. In *ICCV*, pages 4015–4026, 2023. 2, 3

[26] Yutong Kou, Jin Gao, Bing Li, and et al. Zoomtrack: Target-aware non-uniform resizing for efficient visual tracking. *NIPS*, 36:50959–50977, 2023. 2

[27] Shuiwang Li, Yangxiang Yang, Dan Zeng, and Xucheng Wang. Adaptive and background-aware vision transformer for real-time uav tracking. In *ICCV*, pages 13943–13954, 2023. 1, 6

[28] Siyi Li and D. Y. Yeung. Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models. In *AAAI*, 2017. 1, 2, 4

[29] Yiming Li, Changhong Fu, Fangqiang Ding, and et al. Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization. In *CVPR*, pages 11920–11929, 2020. 6

[30] Yongxin Li, Mengyuan Liu, You Wu, and et al. Learning adaptive and view-invariant vision transformer for real-time uav tracking. In *ICML*, 2024. 1, 6

[31] Yunfeng Li, Bo Wang, Xueyi Wu, Zhuoyan Liu, and Ye Li. Lightweight full-convolutional siamese tracker. *Knowledge-Based Systems*, 286:111439, 2024. 1

[32] Christoph Mayer, Martin Danelljan, and et al. Learning target candidate association to keep track of what not to track. In *ICCV*, pages 13424–13434, 2021. 2

[33] Christoph Mayer, Martin Danelljan, and et al. Transforming model prediction for tracking. In *CVPR*, pages 8721–8730, 2022. 2

[34] Matthias Mueller, Neil G. Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *ECCV*, 2016. 1, 2, 4, 5

[35] Huafeng Qin, Xin Jin, Yun Jiang, Mounim A El-Yacoubi, and Xinbo Gao. Adversarial automixup. *arXiv preprint arXiv:2312.11954*, 2023. 2, 3

[36] Liangtao Shi, Bineng Zhong, Qihua Liang, Ning Li, Shengping Zhang, and Xianxian Li. Explicit visual prompts for visual object tracking. In *AAAI*, 2024. 2

[37] Ning Wang, Wengang Zhou, and et al. Multi-cue correlation filters for robust visual tracking. In *CVPR*, pages 4844–4853, 2018. 6

[38] 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*, pages 1571–1580, 2021. 2

[39] Xucheng Wang, Dan Zeng, Qijun Zhao, and Shuiwang Li. Rank-based filter pruning for real-time uav tracking. In *ICME*, pages 01–06. IEEE, 2022. 6

[40] Xing Wei, Yifan Bai, and et al. Autoregressive visual tracking. In *CVPR*, pages 9697–9706, 2023. 4

[41] Qiangqiang Wu, Tianyu Yang, and et al. Dropmae: Masked autoencoders with spatial-attention dropout for tracking tasks. In *CVPR*, pages 14561–14571, 2023. 4

[42] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1834–1848, 2015. 1

[43] Jinxia Xie and et al. Autoregressive queries for adaptive tracking with spatio-temporal transformers. In *CVPR*, pages 19300–19309, 2024. 2

[44] Liangliang Yao, Changhong Fu, and et al. Sgdvit: Saliency-guided dynamic vision transformer for uav tracking. *arXiv preprint arXiv:2303.04378*, 2023. 1, 6

[45] Botao Ye, Hong Chang, and et al. Joint feature learning and relation modeling for tracking: A one-stream framework. In *ECCV*, pages 341–357, 2022. 2

[46] Sangdoo Yun, Dongyoon Han, and et al. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, pages 6023–6032, 2019. 2, 3

[47] Dan Zeng, Mingliang Zou, Xucheng Wang, and Shuiwang Li. Towards discriminative representations with contrastive instances for real-time uav tracking. In *ICME*, pages 1349–1354, 2023. 6

[48] Zhipeng Zhang, Yihao Liu, Xiao Wang, Bing Li, and Weiming Hu. Learn to match: Automatic matching network design for visual tracking. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13319–13328, 2021. 2

[49] Haojie Zhao, Dong Wang, and Huchuan Lu. Representation learning for visual object tracking by masked appearance transfer. In *CVPR*, pages 18696–18705, 2023. 2

[50] Xin Zhao, Shiyu Hu, Yipei Wang, Jing Zhang, Yimin Hu, Rongshuai Liu, Haibin Ling, Yin Li, Renshu Li, Kun Liu, et al. Biodrone: A bionic drone-based single object tracking benchmark for robust vision. *International Journal of Computer Vision*, 132(5):1659–1684, 2024. 1

[51] Zikun Zhou, Wenjie Pei, Xin Li, and et al. Saliency-associated object tracking. In *ICCV*, pages 9846–9855, 2021. 2

[52] Pengfei Zhu, Longyin Wen, and et al. Visdrone-vdt2018: The vision meets drone video detection and tracking challenge results. In *ECCV Workshops*, 2018. 1, 2, 4, 6

[53] Haobo Zuo, Changhong Fu, and et al. Adversarial blur-deblur network for robust uav tracking. *RAL*, 8(2):1101–1108, 2023. 6