TrackPGD: Efficient Adversarial Attack using Object Binary Masks against Robust Transformer Trackers

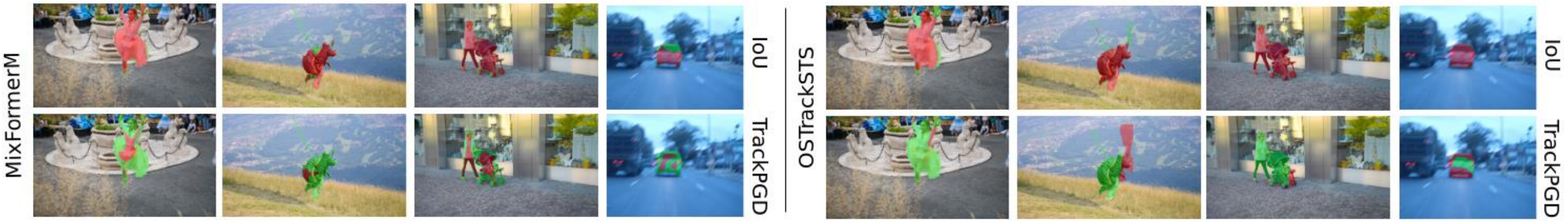
Fatemeh Nourilenjan Nokabadi[1,2,3], Yann Batiste Pequignot[1,2], Jean-François Lalonde[1,2], Christian Gagné[1,2,3,4]

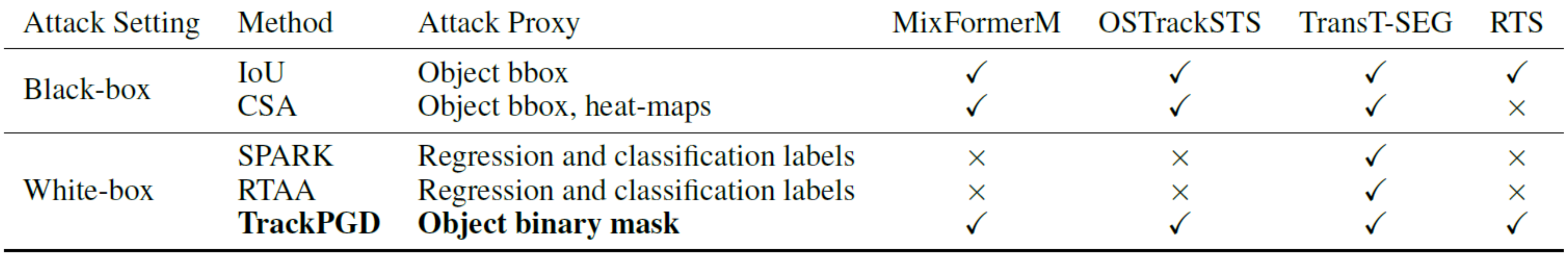
[1]IID, [2]Université Laval, [3]Mila, [4]Canada CIFAR AI Chair

The 3rd New Frontiers in Adversarial Machine Learning
AdvML Frontiers @NeurIPS2024, Vancouver, Canada

Adversarial Robustness

Adversarial attacks can significantly harm neural networks performances by adding carefully crafted imperceptible noise to the input.





Attack Setting	Method	Attack Proxy	MixFormerM	OSTrackSTS	TransT-SEG	RTS
Black-box	IoU	Object bbox	✓	✓	✓	✓
	CSA	Object bbox, heat-maps	✓	✓	✓	×
White-box	SPARK	Regression and classification labels	×	×	✓	×
	RTAA	Regression and classification labels	×	×	✓	×
	TrackPGD	**Object binary mask**	✓	✓	✓	✓

Research Contributions

1. TrackPGD builds the adversarial perturbations from the binary mask.

2. Difference loss is proposed to misleading trackers in providing an accurate binary mask.

3. MixFormerM suffers substantial accuracy (−75%), average overlap (−97%) and robustness (−91%) reductions after applying TrackPGD, evaluated on the VOT2022STS dataset.

4. Experimental results also demonstrate that the perturbations generated by TrackPGD have a substantial influence on bounding box predictions in tracking benchmarks.

Preliminaries: SegPGD

Y: mask annotation

$f_{seg}(X_{adv})$: Adversarial segmentation map

X_{adv}: Adversarial image

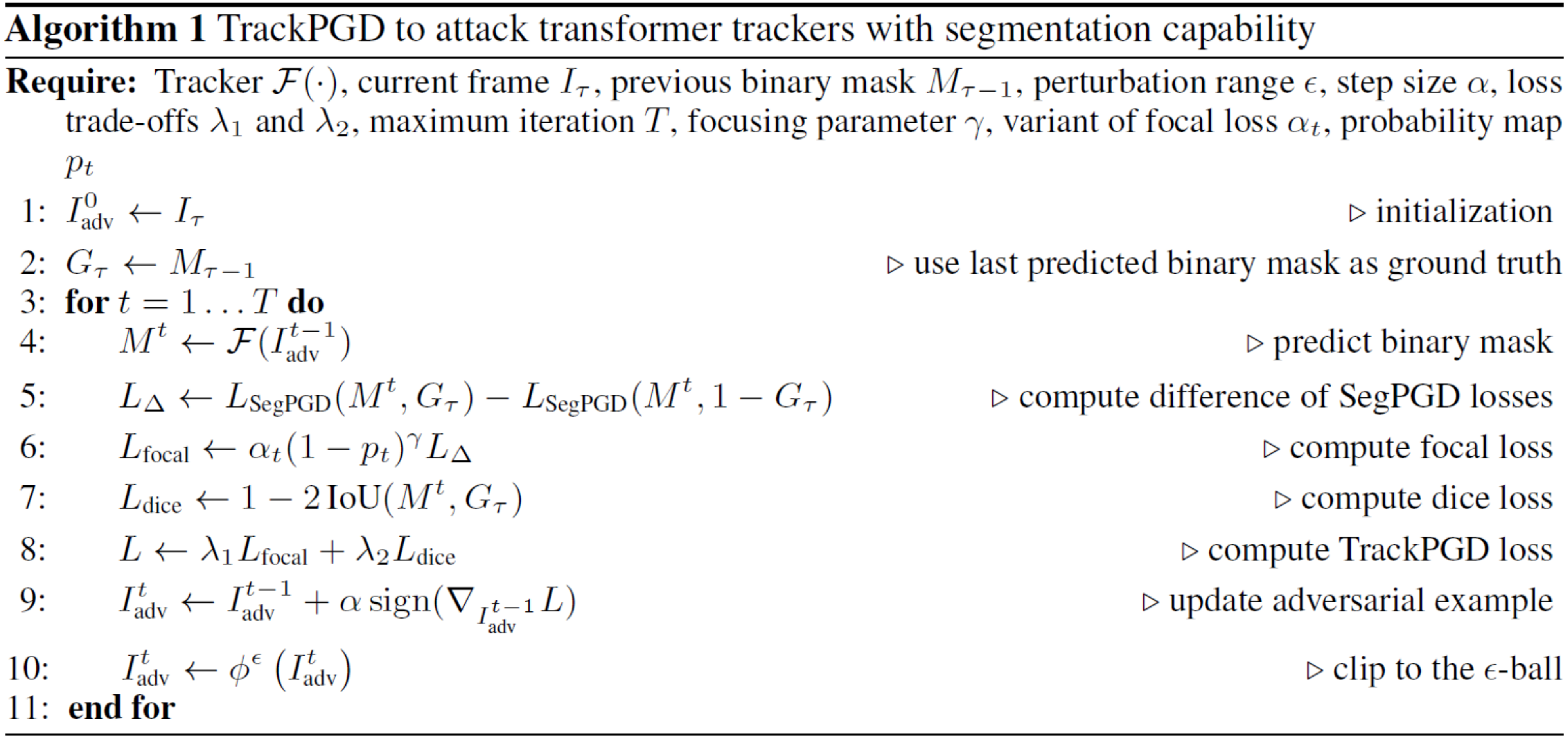
$$L_{\mathrm{SegPGD}}(f_{\mathrm{seg}}(X^{\mathrm{adv}}), Y) = \frac{1-\lambda}{HW} L_{\mathrm{pos}} + \frac{\lambda}{HW} L_{\mathrm{neg}} \, ,$$

We must overcome two key challenges to build TrackPGD from SegPGD:

1. While SegPGD operates on multi-class segmentation networks, binary masks only have two classes, necessitating the switching of classification labels between pixels.

2. Secondly, the labels in the majority of samples are imbalanced, with the number of object pixels being significantly fewer than background pixels.

Proposed method: TrackPGD



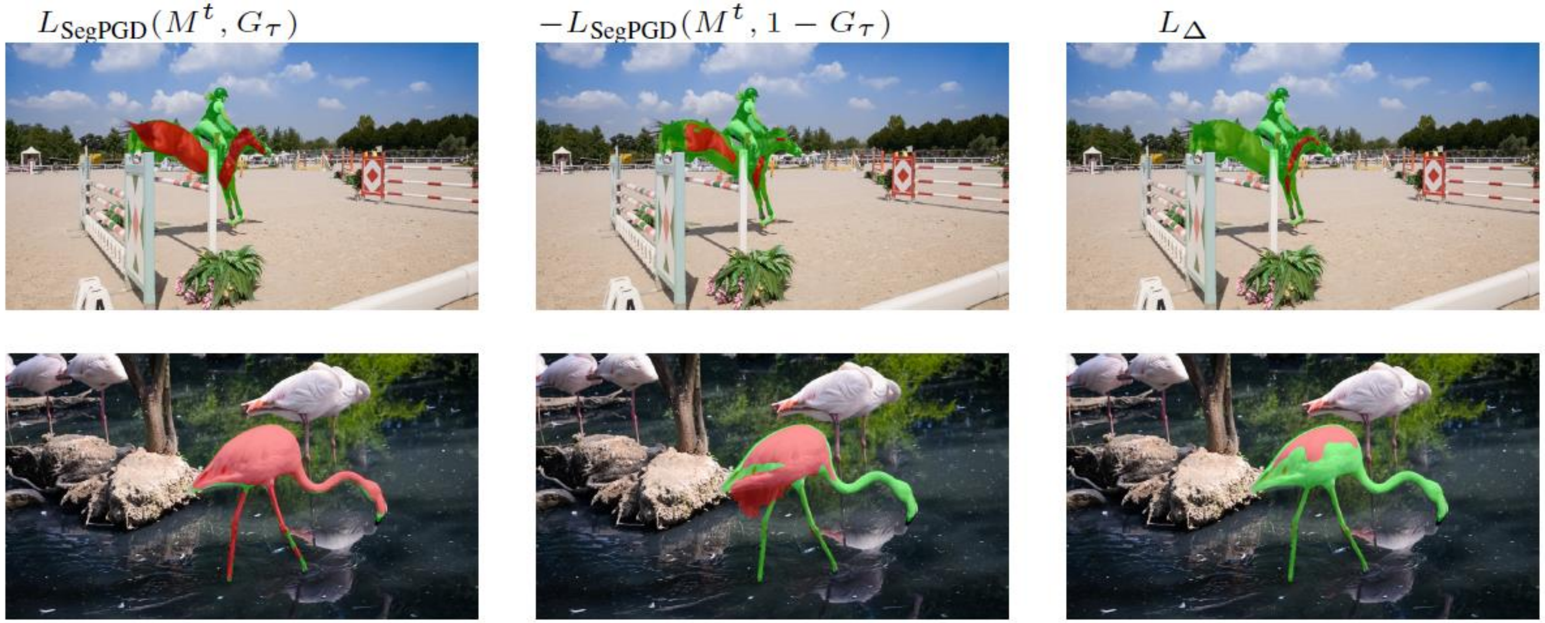
Algorithm 1 TrackPGD to attack transformer trackers with segmentation capability

```
Require: Tracker $\mathcal{F}(\cdot)$, current frame $I_\tau$, previous binary mask $M_{\tau-1}$, perturbation range $\epsilon$, step size $\alpha$, loss trade-offs $\lambda_1$ and $\lambda_2$, maximum iteration $T$, focusing parameter $\gamma$, variant of focal loss $\alpha_t$, probability map $p_t$
$I^0_{\text{adv}} \leftarrow I_\tau$                                          ▷ initialization
$G_\tau \leftarrow M_{\tau-1}$                                         ▷ use last predicted binary mask as ground truth
for $t = 1 \ldots T$ do
    $M^t \leftarrow \mathcal{F}(I^{t-1}_{\text{adv}})$                                  ▷ predict binary mask
    $L_\Delta \leftarrow L_{\text{SegPGD}}(M^t, G_\tau) - L_{\text{SegPGD}}(M^t, 1 - G_\tau)$     ▷ compute difference of SegPGD losses
    $L_{\text{focal}} \leftarrow \alpha_t(1 - p_t)^\gamma L_\Delta$                     ▷ compute focal loss
    $L_{\text{dice}} \leftarrow 1 - 2\,\text{IoU}(M^t, G_\tau)$                      ▷ compute dice loss
    $L \leftarrow \lambda_1 L_{\text{focal}} + \lambda_2 L_{\text{dice}}$                     ▷ compute TrackPGD loss
    $I^t_{\text{adv}} \leftarrow I^{t-1}_{\text{adv}} + \alpha\,\text{sign}(\nabla_{I^{t-1}_{\text{adv}}} L)$          ▷ update adversarial example
    $I^t_{\text{adv}} \leftarrow \phi^\epsilon\left(I^t_{\text{adv}}\right)$                          ▷ clip to the $\epsilon$-ball
end for
```

Role of L_Δ in TrackPGD

The vanilla SegPGD losses vs. the difference loss in the TrackPGD against MixFormerM on DAVIS2016

Original	$L_{\text{SegPGD}}(M^t, G_\tau)$	$-L_{\text{SegPGD}}(M^t, 1-G_\tau)$	L_Δ
85.82	52.86	37.28	**30.30**

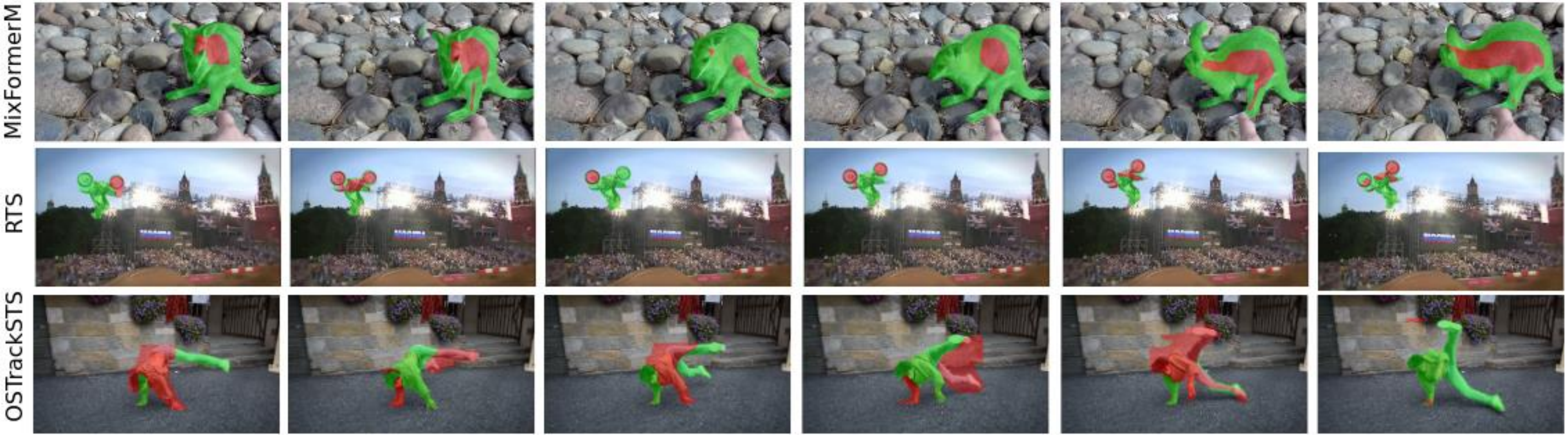


$L_{\text{SegPGD}}(M^t, G_\tau)$

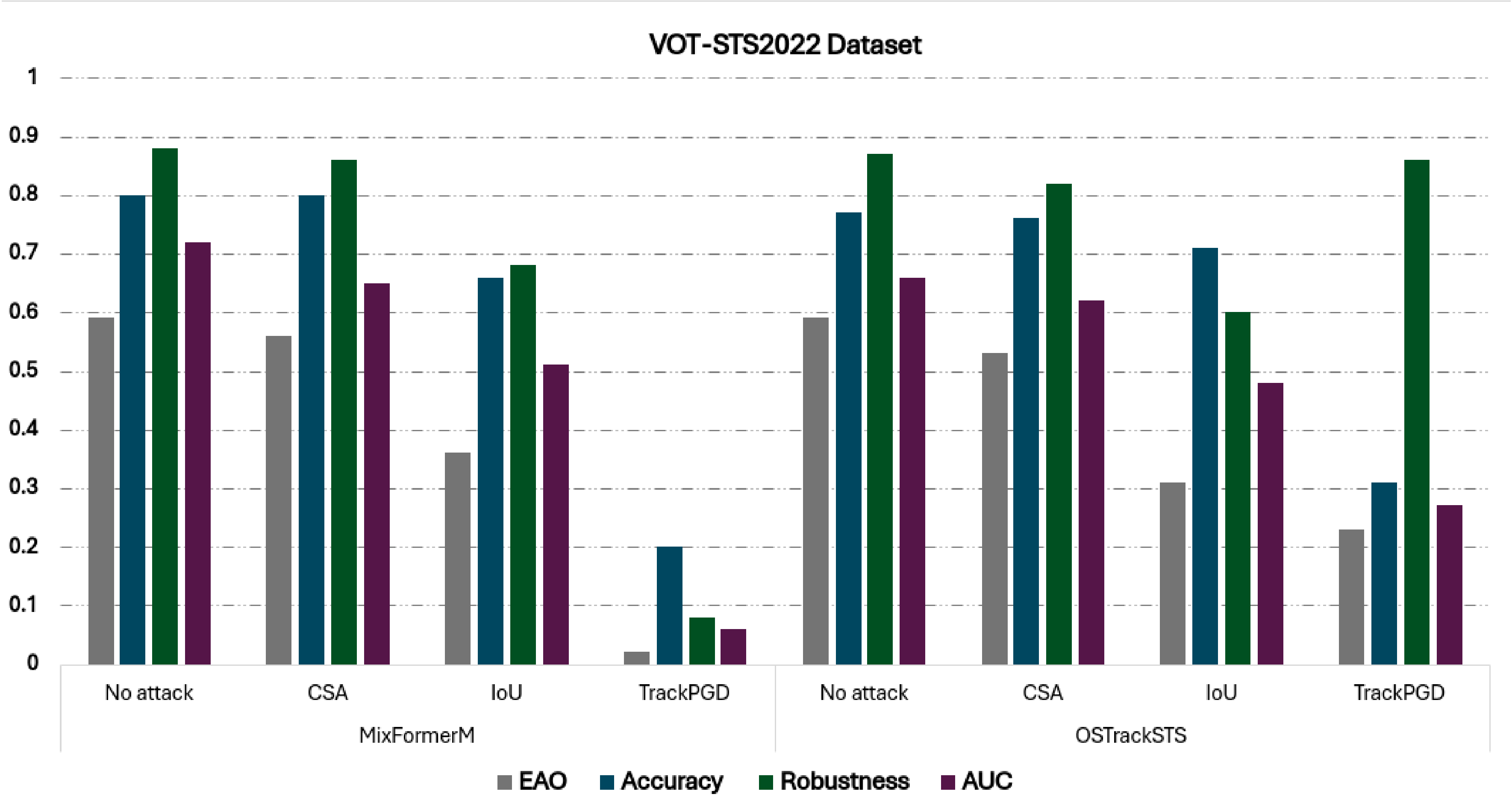
$-L_{\text{SegPGD}}(M^t, 1-G_\tau)$

L_Δ

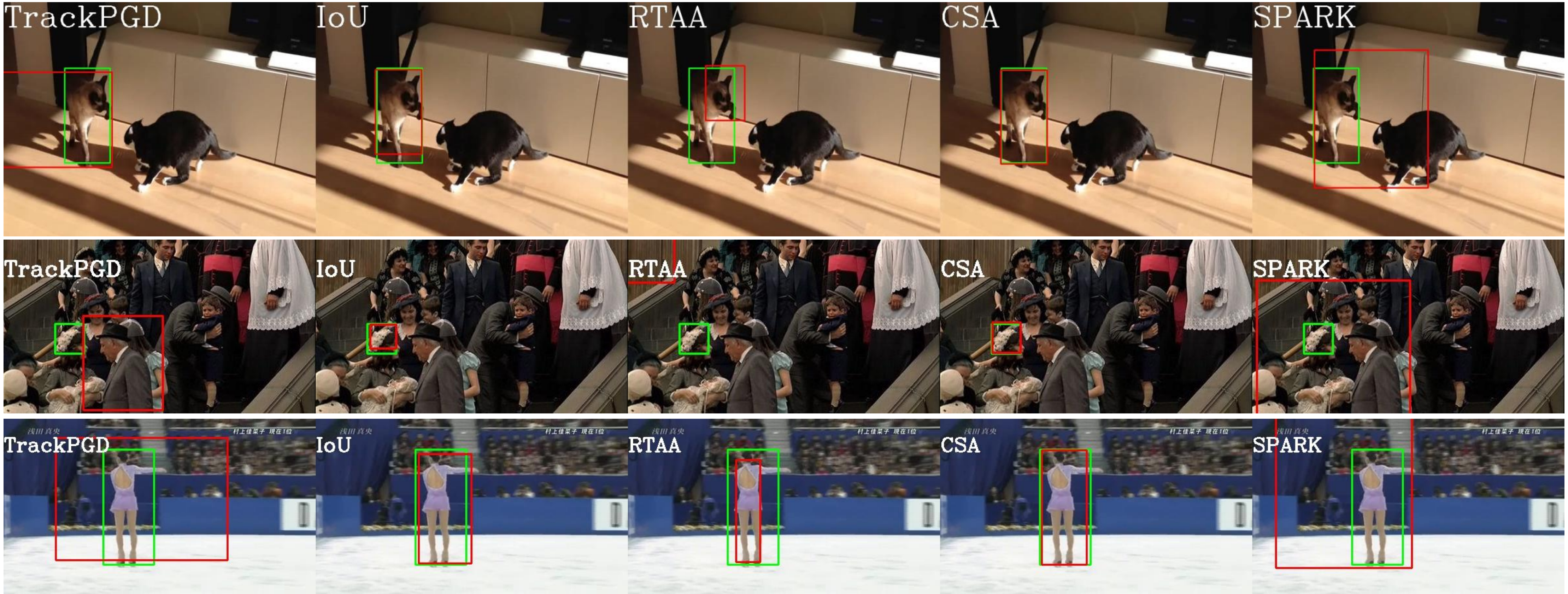
Object Binary Mask Evaluation



Object Binary Mask Evaluation



Object Bounding Box Evaluation



Conclusion

- We proposed TrackPGD, a novel white-box attack that leverages object binary masks to assess the adversarial robustness of transformer trackers.

- We highlighted the effectiveness of our proposed difference loss in impacting tracker performance compared to standard segmentation losses such as SegPGD.

- The efficacy of TrackPGD is validated through comprehensive experiments on various transformer architecture networks and popular datasets.