NEURAL INFORMATION PROCESSING SYSTEMS

ZoomTrack: Target-aware Non-uniform Resizing for Efficient Visual Tracking

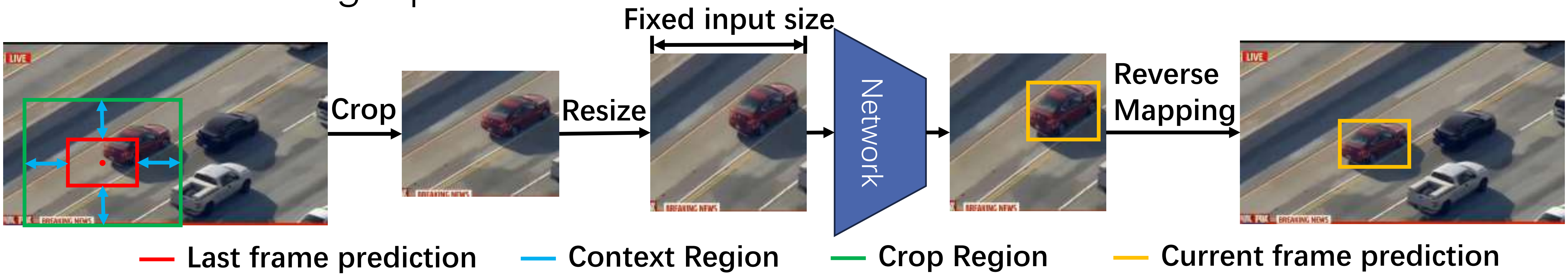
Yutong Kou · Jin Gao · Bing Li · Gang Wang · Weiming Hu · Yizheng Wang · Liang Li

Institute of Automation, Chinese Academy of Sciences

Beijing Institute of Basic Medical Sciences   People AI

Background and Motivations

- Visual Tracking Pipeline

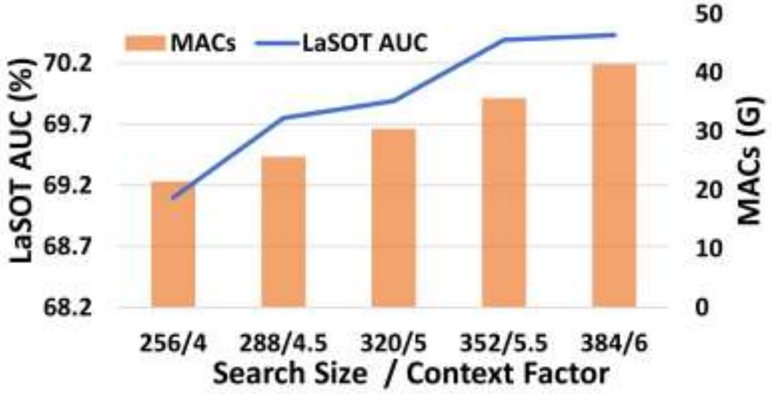


- Larger input size leads to better accuracy and longer latency

Tracker	Year	Input Size	LaSOT AUC	Δ AUC	FPS	Δ FPS
OSTrack-384 OSTrack-256	2022	384 256	71.1% 69.1%	+2.0%	58 105	-44.8%
SwinTrack-B SwinTrack-T	2022	384 256	67.2% 71.3%	+4.1%	45 98	-54.1%
SeqTrack-B384 SeqTrack-B256	2023	384 256	71.5% 69.9%	+1.6%	15 40	-62.5%

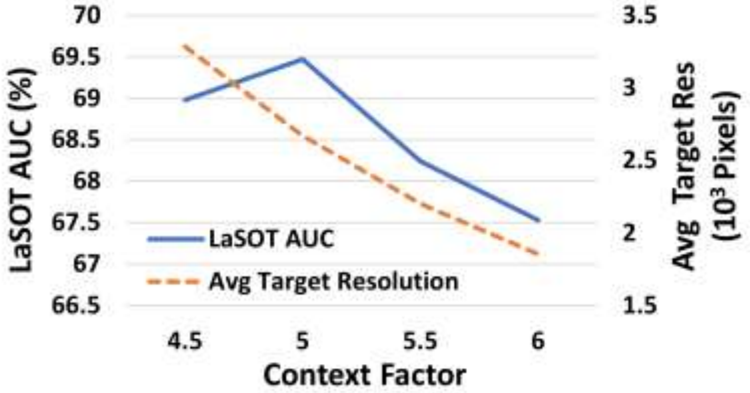
Can we narrow this accuracy gap without sacrificing too much speed?

Why larger input size improves accuracy?



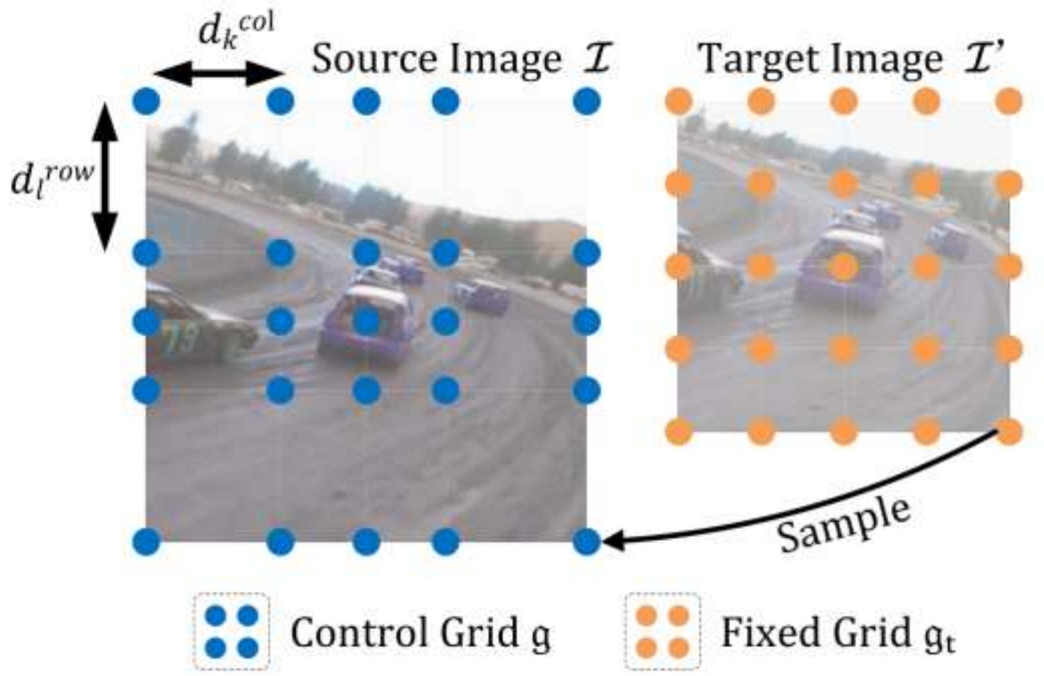
Larger visual field improves accuracy when target resolution is sufficiently large at the cost of larger input size

How to simultaneously have large visual field and large target resolution under a smaller input size?



The improvement brought by increased visual field is wiped out by decreased target resolution, when fixing a small input size

Non-uniform resizing for visual tracking



By sampling uniformly distributed pixels at non-uniformly distributed locations, the search area of current frame is non-uniformly resized. The non-uniformly resizing is controlled by a non-uniform control grid $\mathfrak{g}$.

- QP–based control grid $\mathfrak{g}$ generation

$$\underset{d_l^{row},\ d_k^{col}}{\text{minimize}} \quad E = E_{zoom} + \lambda E_{rigid}$$

$$\text{subject to} \quad \sum_{l=1}^{m} d_l^{row} = H, \ \sum_{k=1}^{n} d_k^{col} = W$$

- $S_{k,l}$: importance score, higher at location where the target is more likely to appear, G(x,y) is a Gaussian function

$$S_{k,l} = G\left(\left(k+\frac{1}{2}\right)\times\frac{W}{n}, \left(l+\frac{1}{2}\right)\times\frac{H}{m}\right) + \epsilon$$

- E_{zoom}: Zoom the target region by γ

$$E_{zoom} = \sum_{l=1}^{m}\sum_{k=1}^{n} S_{k,l}^2 \left(\left(d_l^{row} - \frac{1}{\gamma}\frac{H}{m}\right)^2 + \left(d_k^{col} - \frac{1}{\gamma}\frac{W}{n}\right)^2\right)$$

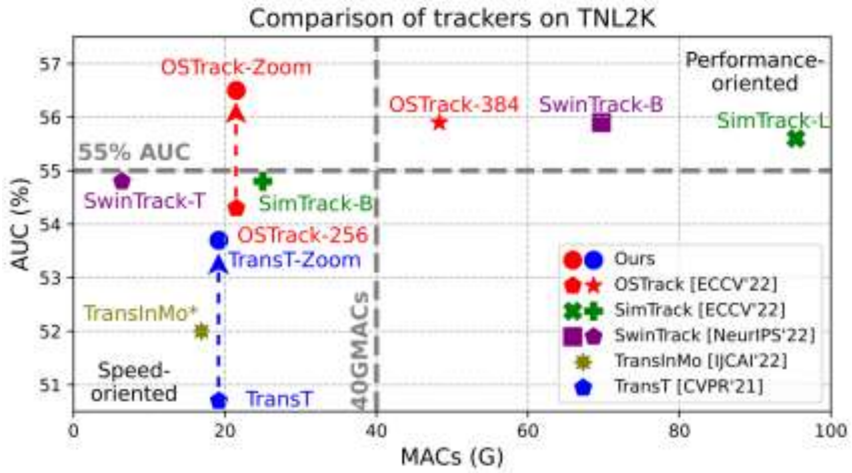
- E_{rigid}: Avoid extreme deformation

$$E_{rigid} = \sum_{l=1}^{m}\sum_{k=1}^{n} S_{k,l}^2 \left(\frac{m}{H} d_l^{row} - \frac{n}{W} d_k^{col}\right)^2$$

Results

Group	Tracker	Size	GOT-10k[13] AO	GOT-10k[13] $SR_{0.5}$	GOT-10k[13] $SR_{0.75}$	LaSOT[11] AUC	LaSOT[11] P	$LaSOT_{ext}$[10] AUC	$LaSOT_{ext}$[10] P	TNL2K[23] AUC	TNL2K[23] P	TrackingNet[18] SUC	TrackingNet[18] P	MACs (G)	FPS
Baseline &Ours	OSTrack-Zoom	256	**73.5**	**83.6**	**70.0**	**70.2**	**76.2**	**50.5**	**57.4**	**56.5**	**57.3**	**83.2**	**82.2**	21.5	100
	OSTrack-256[26]	256	71.0	80.4	68.2	69.1	75.2	47.4	53.3	54.3	-	83.1	82.0	21.5	119
	TransT-Zoom	255	67.5	77.6	61.3	67.1	71.6	46.8	52.9	53.7	62.3	81.8	80.2	19.2	45
	TransT[8]	255	67.1	76.8	60.9	64.9	69.0	44.8	52.5	50.7	51.7	81.4	80.3	19.2	48
Speed-oriented	SwinTrack-T[15]	224	71.3	81.9	64.5	67.2	70.8	47.6	53.9	53.0	53.2	81.1	78.4	6.4	98
	SimTrack-B[7]	224	68.6	78.9	62.4	69.3	-	-	-	54.8	53.8	82.3	-	25.0	40
	MixFormer-22k[9]	320	70.7	80.0	67.8	69.2	74.7	-	-	-	-	83.1	81.6	23.0	25
	ToMP-101[17]	288	-	-	-	68.5	73.5	45.9	-	-	-	81.5	78.9	-	20
	TransInMo*[12]	255	-	-	-	65.7	70.7	-	-	52.0	52.7	81.7	-	16.9	34
	Stark-ST101[24]	320	68.8	78.1	64.1	67.1	-	-	-	-	-	82.0	-	28.0	32
	AutoMatch[27]	255	65.2	76.6	54.3	58.3	59.9	37.6	43.0	47.2	43.5	76.0	72.6	-	50
	DiMP[4]	288	61.1	71.7	49.2	56.9	56.7	39.2	45.1	44.7	43.4	74.0	68.7	5.4	40
	SiamRPN++[14]	255	51.7	61.6	32.5	49.6	49.1	34.0	39.6	41.3	41.2	73.3	69.4	7.8	35
Performance -oriented	OSTrack-384[26]	384	73.7	83.2	70.8	71.1	77.6	50.5	57.6	55.9	-	83.9	83.2	48.3	61
	SwinTrack-B[15]	384	72.4	80.5	67.8	71.3	76.5	49.1	55.6	55.9	57.1	84.0	82.8	69.7	45
	SimTrack-L[7]	224	69.8	78.8	66.0	70.5	-	-	-	55.6	55.7	83.4	-	95.4	-
	MixFormer-L[9]	320	-	-	-	70.1	76.3	-	-	-	-	83.9	83.1	127.8	18

- Consistent improvements over two baselines on multiple large-scale tracking datasets
- Narrowed the accuracy gap between speed/ accuracy oriented trackers
- Causing little computational overhead (1.58 ms & 1.28M MACs)

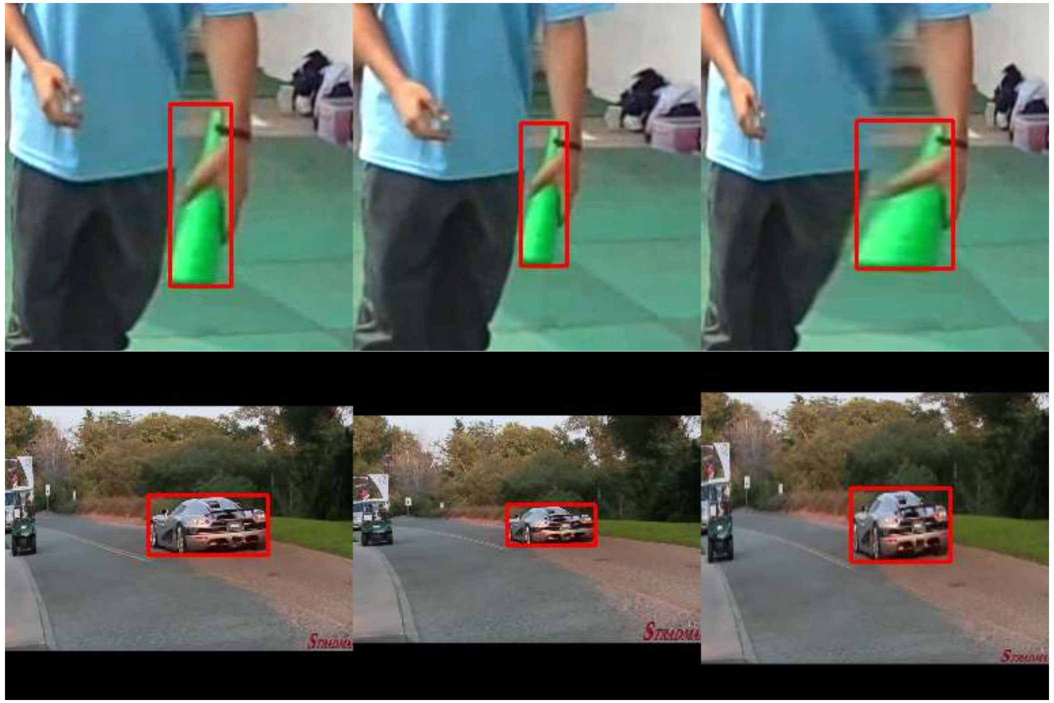


- Outperforms SOTA method on TNL2K while reduces more than 50% of computation

#	Non-uniform resizing Testing	Non-uniform resizing Training	OSTrack[33] LaSOT[11]	OSTrack[33] TNL2K[30]	TransT[8] LaSOT[11]	TransT[8] TNL2K[30]
①			69.1	54.3	64.9	50.7
②	✔		69.1	55.9	66.3	53.6
③	✔	✔	**70.2**	**56.5**	**67.1**	**53.7**

- Improve off-the-shelf models without training

Visualization



Ours | Uniform Resizing | FOVEA

- Compared with uniform resizing, we have larger target resolution
- Compared other non-uniform resizing method (FOVEA), we have less deformation and scale change

Faster re-detection | Fewer target lost

Ground Truth — OSTrack-Zoom — OSTrack

NEURAL INFORMATION PROCESSING SYSTEMS

Conclusion

Paper

Code

An efficient non-uniform resizing method for visual tracking

- Non-uniformly resize the cropped search area to have a smaller input size while the resolution of the area where the target is more likely to appear is higher, vice versa.
- Bridge the gap between speed-oriented and performance-oriented trackers with negligible computation
- Push forward the Pareto front of MACs and AUC trade-off

NEURAL INFORMATION PROCESSING SYSTEMS