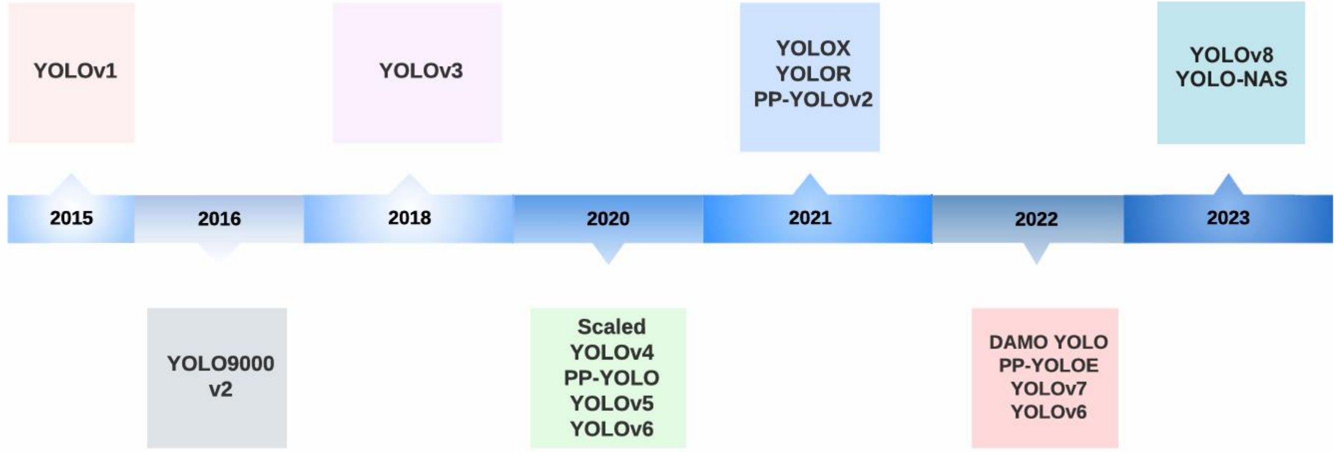
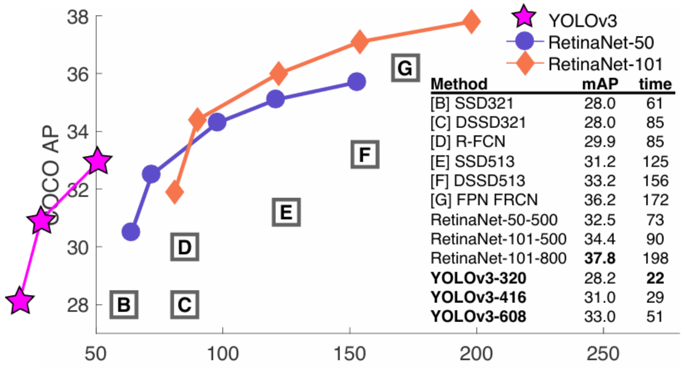
YOLOV10: REAL-TIME END-TO-END OBJECT DETECTION

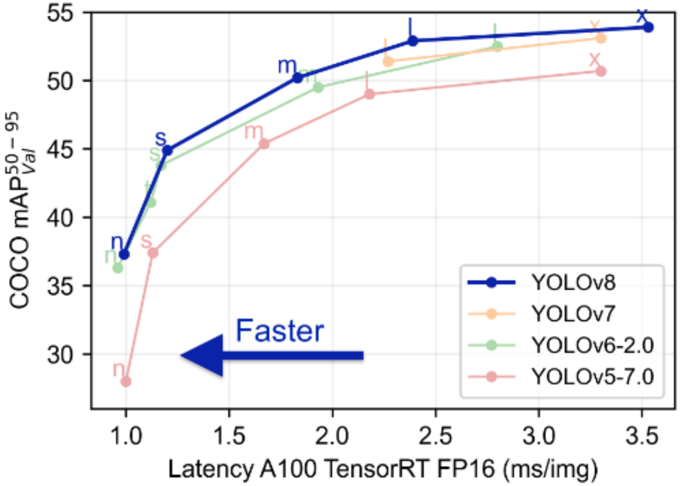
Ao Wang[1], Hui Chen[2,†], Lihao Liu[1], Kai Chen[1], Zijia Lin[1], Jungong Han[1], Guiguang Ding[1,†]
[1]Tsinghua University [2]BNRist

Background

- Since the initial release in 2015, YOLO (You Only Look Once) series of models has achieved significant advancements in the field of real-time object detection. It enjoys high performance and fast inference speed.
- The typical architecture of YOLO includes Backbone, FPN, and Head, for extracting multi-scale features, fusing these features, and outputting predictions, respectively.

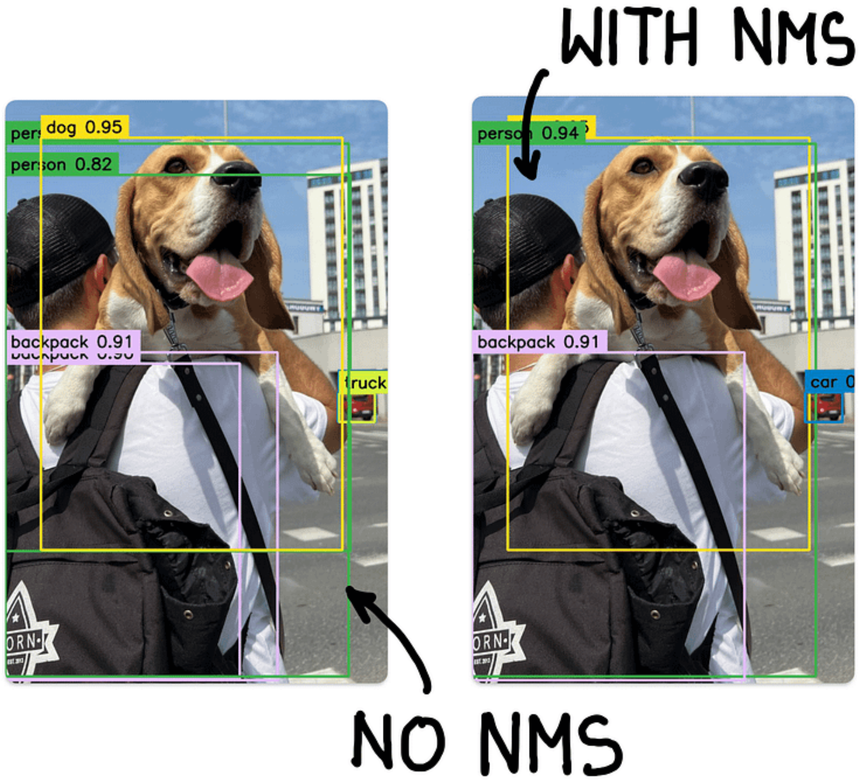






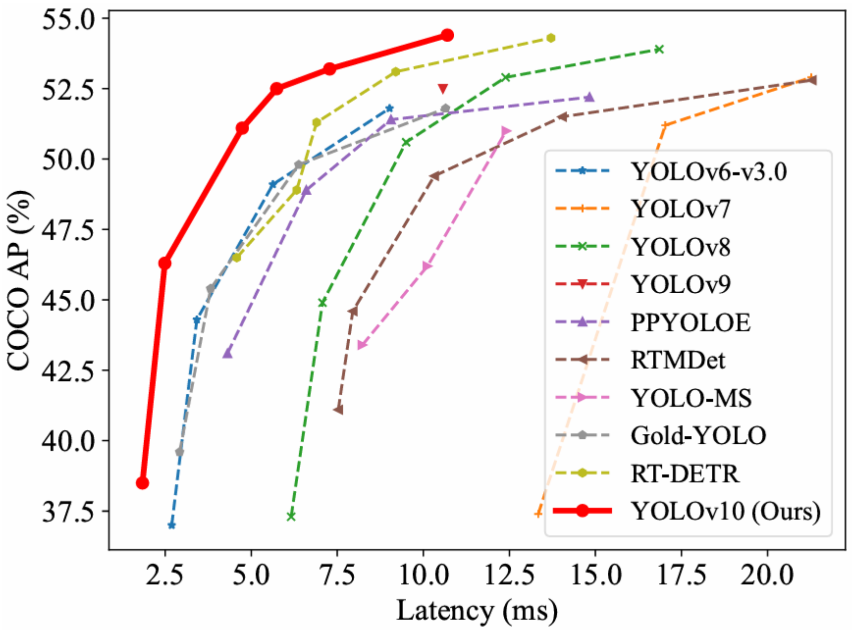
Motivation

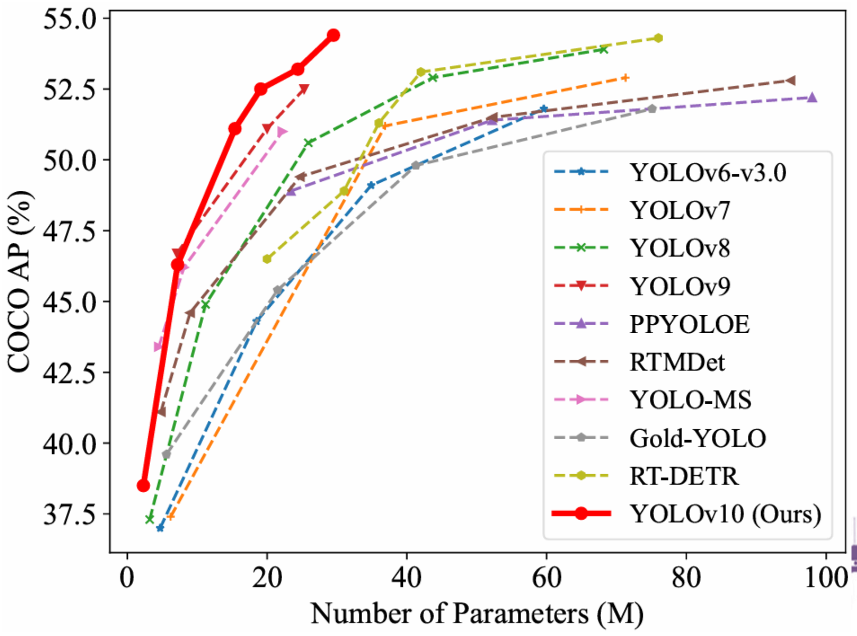
- YOLO relies on Non-Maximum Suppression (NMS) for post-processing, which hinders end-to-end deployment.
  - Adopt consistent dual assignments to eliminate the NMS.
  - Enjoy rich supervision and end-to-end inference simultaneously.
- YOLO variants exhibit computational redundancy and limited modeling capacity, indicating the room for improvement in both efficiency and accuracy.
  - Employ efficiency-driven model design simplifies components.
  - Utilize accuracy-driven model design enhances the performance.
- Improvements based on YOLOv8 lead to the YOLOv10.



Motivation

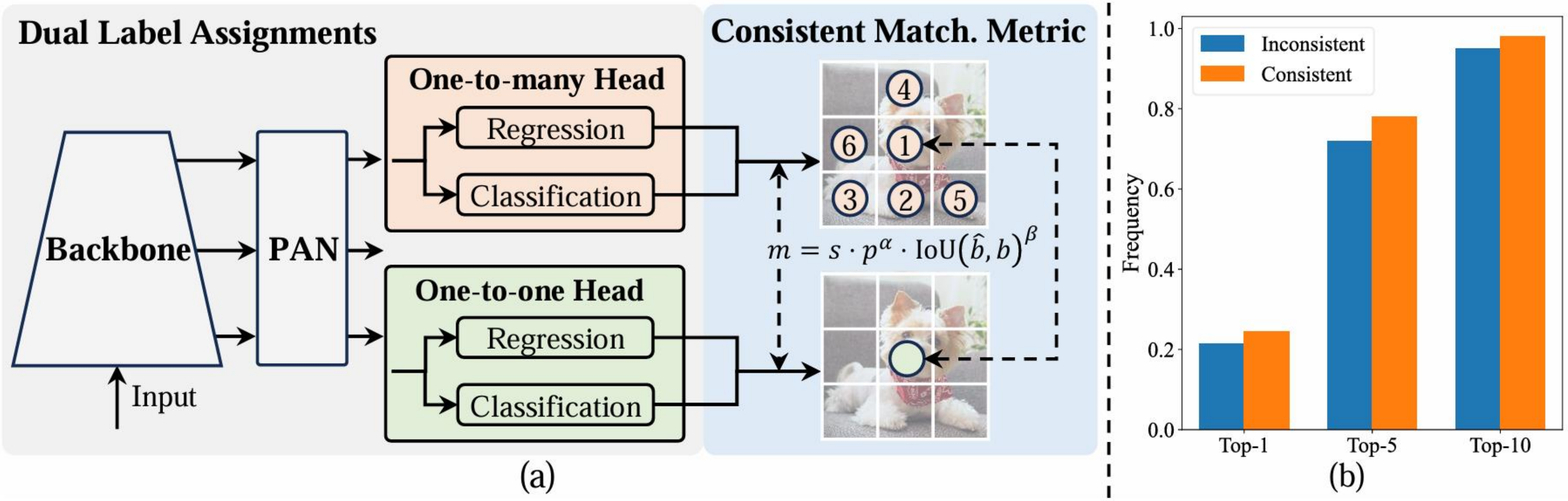
- YOLOv10 achieves state-of-the-art balance between performance and efficiency across various scales.
- YOLOv10-S is 1.8× faster than RT-DETR-R18, with 2.8× fewer parameters and FLOPs.
- Compared to YOLOv9-C, YOLOv10-B reduces latency by 46% and parameters by 25% while maintaining comparable performance.





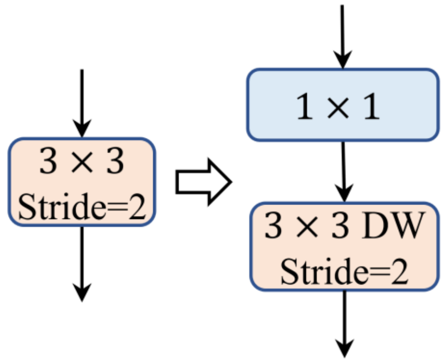
Methodology

- Consistent Dual Assignments for NMS-free Training
  - One-to-one matching eliminates the need for NMS but with limited supervisory information. In contrast, one-to-many strategy provides rich supervision signal. Therefore, dual label assignments are introduced for YOLO, as shown in the Figure (a), to fully leverage the advantages of both strategies.
  - Consistent matching metric is further employed to minimize the supervision gap between two heads, as shown in the Figure (b). Assuming the matching metric takes the form $m = p^{\alpha}\mathrm{IoU}^{\beta}$, then $\alpha_{o2o} = r \cdot \alpha_{o2m}$ and $\beta_{o2o} = r \cdot \beta_{o2m}$, which encourage the same optimal positive sample for two branches

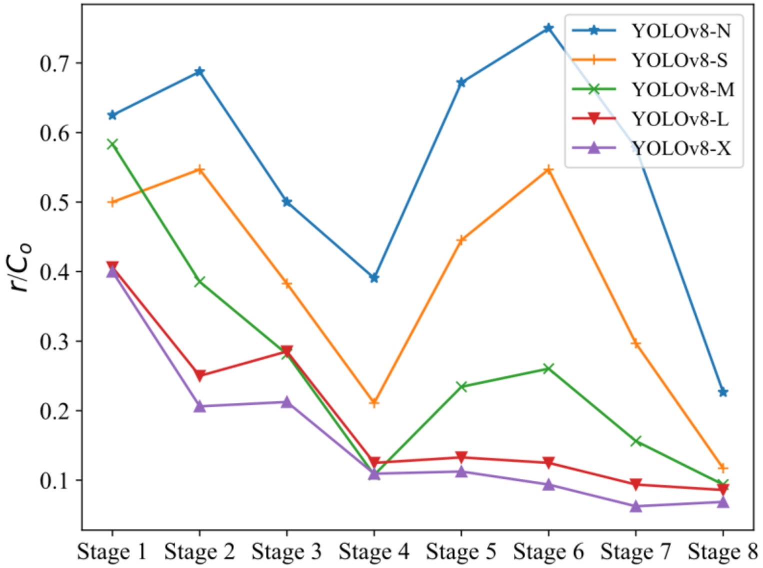


Methodology

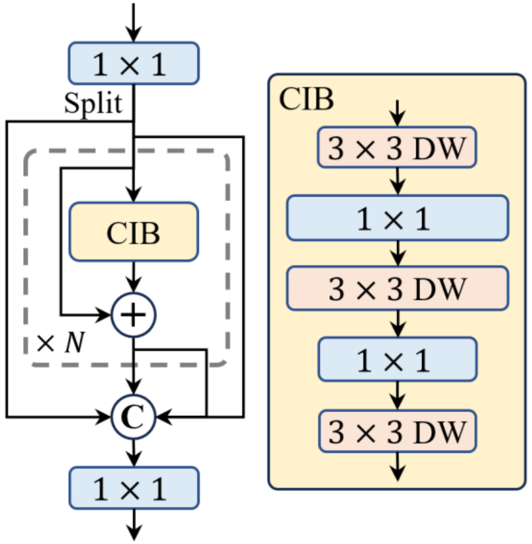
- Efficiency driven model design
  - Lightweight classification head: Employ the lightweight design of PW(DW) to reduce the redundancy in the classification task.
  - Spatial-channel decoupled downsampling: Decouple the spatial reduction by DW and the channel expansion by PW for efficiency. (a) & (b)



(a) (b)



(c)

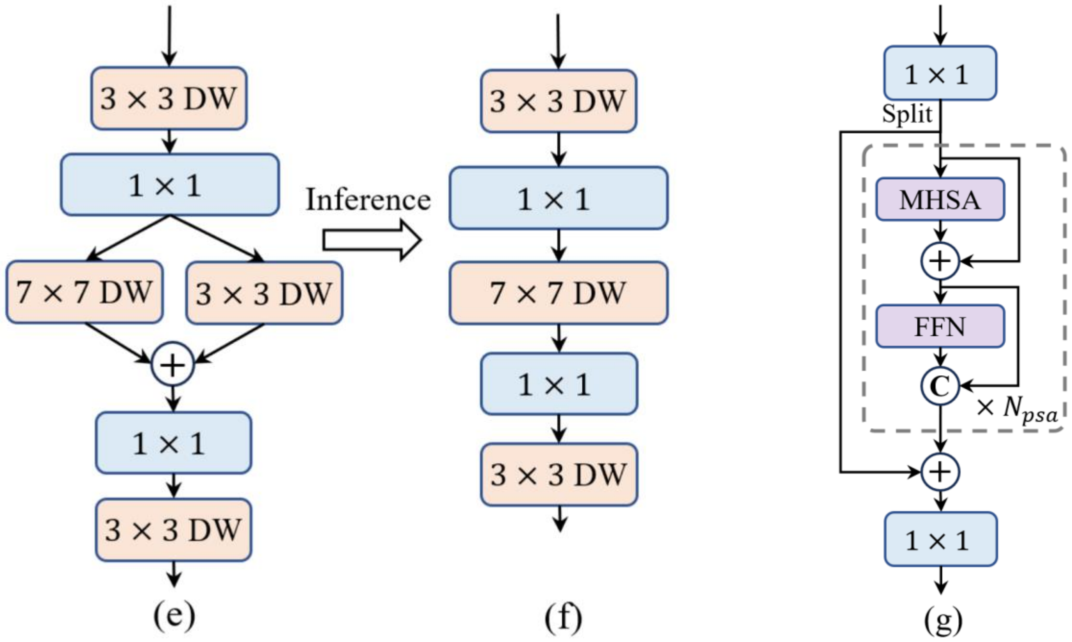


(d)

- Rank-guided block design: Adopt the compact inverted block for higher efficiency adaptively based on the intrinsic ranks which indicates the redundancy. (c) & (d)

Methodology

- Accuracy driven model design
  - Large kernel convolution: Employ large kernel DW to effectively compensates for the insufficient receptive field of small models. (e) & (f)
  - Partial self-attention: Introduce global representation learning by operating on partial channels to reduce the redundancy in attention heads. (g)



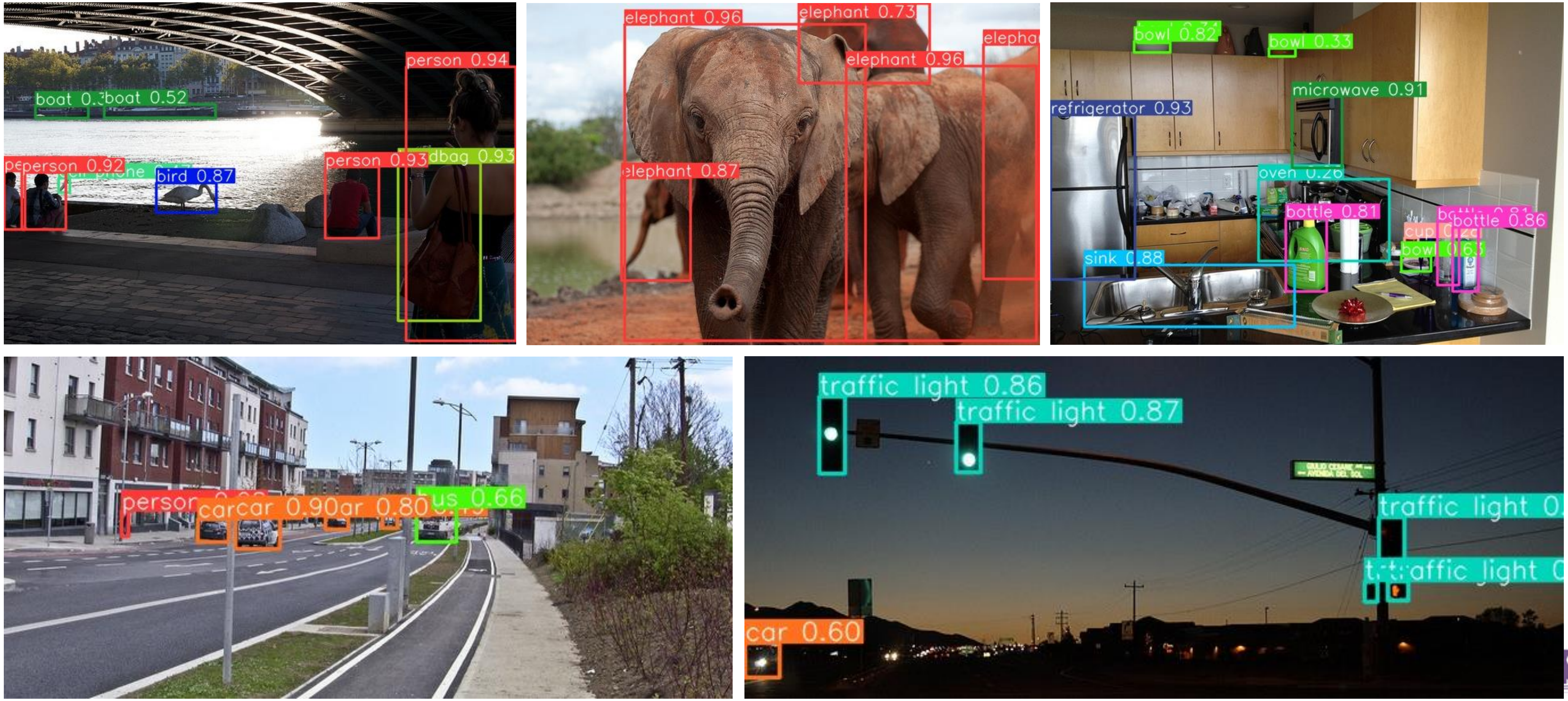
Experiments

Model	#Param.(M)	FLOPs(G)	AP^{val}(%)	Latency(ms)	$Latency^{f}$(ms)
YOLOv6-3.0-N [27]	4.7	11.4	37.0	2.69	1.76
Gold-YOLO-N [54]	5.6	12.1	39.6	2.92	1.82
YOLOv8-N [20]	3.2	8.7	37.3	6.16	1.77
YOLOv10-N (Ours)	**2.3**	**6.7**	**38.5 / 39.5†**	**1.84**	**1.79**
YOLOv6-3.0-S [27]	18.5	45.3	44.3	3.42	2.35
Gold-YOLO-S [54]	21.5	46.0	45.4	3.82	2.73
YOLO-MS-XS [7]	4.5	17.4	43.4	8.23	2.80
YOLO-MS-S [7]	8.1	31.2	46.2	10.12	4.83
YOLOv8-S [20]	11.2	28.6	44.9	7.07	2.33
YOLOv9-S [59]	7.1	26.4	46.7	-	-
RT-DETR-R18 [71]	20.0	60.0	46.5	4.58	4.49
YOLOv10-S (Ours)	**7.2**	**21.6**	**46.3 / 46.8†**	**2.49**	**2.39**
YOLOv6-3.0-M [27]	34.9	85.8	49.1	5.63	4.56
Gold-YOLO-M [54]	41.3	87.5	49.8	6.38	5.45
YOLO-MS [7]	22.2	80.2	51.0	12.41	7.30
YOLOv8-M [20]	25.9	78.9	50.6	9.50	5.09
YOLOv9-M [59]	20.0	76.3	51.1	-	-
RT-DETR-R34 [71]	31.0	92.0	48.9	6.32	6.21
RT-DETR-R50m [71]	36.0	100.0	51.3	6.90	6.84
YOLOv10-M (Ours)	**15.4**	**59.1**	**51.1 / 51.3†**	**4.74**	**4.63**
YOLOv6-3.0-L [27]	59.6	150.7	51.8	9.02	7.90
Gold-YOLO-L [54]	75.1	151.7	51.8	10.65	9.78
YOLOv9-C [59]	25.3	102.1	52.5	10.57	6.13
YOLOv10-B (Ours)	**19.1**	**92.0**	**52.5 / 52.7†**	**5.74**	**5.67**
YOLOv8-L [20]	43.7	165.2	52.9	12.39	8.06
RT-DETR-R50 [71]	42.0	136.0	53.1	9.20	9.07
YOLOv10-L (Ours)	**24.4**	**120.3**	**53.2 / 53.4†**	**7.28**	**7.21**
YOLOv8-X [20]	68.2	257.8	53.9	16.86	12.83
RT-DETR-R101 [71]	76.0	259.0	54.3	13.71	13.58
YOLOv10-X (Ours)	**29.5**	**160.4**	**54.4 / 54.4†**	**10.70**	**10.60**

- Compared with other YOLO variants, YOLOv10 demonstrates significant advantages in terms of accuracy, parameter count, computational complexity, and latency.
- Compared to the RT-DETR end-to-end model, YOLOv10 demonstrates superior performance in terms of latency. YOLOv10-S and YOLOv10-X are 1.8× and 1.3× faster than RT-DETR-R18 and RT-DETR-R101, respectively, with significantly fewer parameters and FLOPs.

Visualization

YOLOv10 performs well in complex and challenging scenarios.



THANKS！