YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information

Presenter : Hao-Tang Tsui

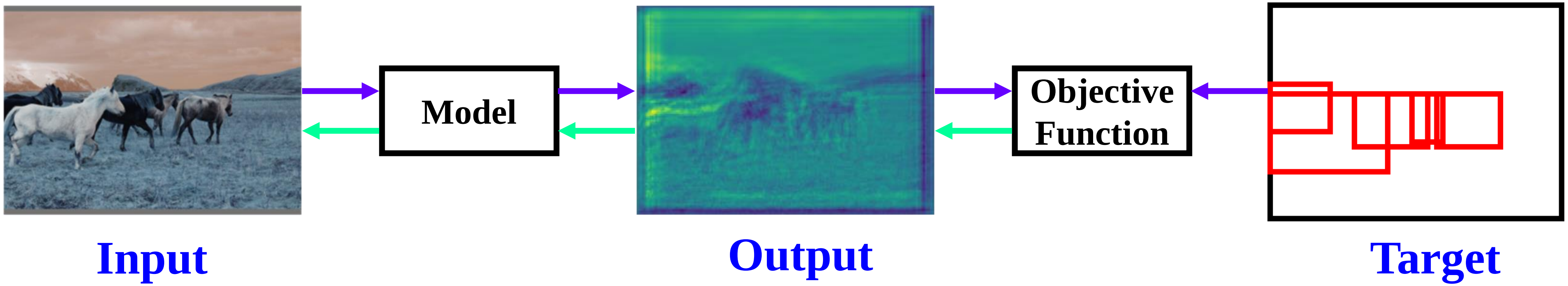
Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao

Institute of Information Science, Academia Sinica, Taiwan

Poster: TUE-AM-Session1

Motivation (1/3)

- YOLOv4 to YOLOv7 learn diverse and consistent features through back-propagated gradient flow.



YOLOv4 to YOLOv7: Optimize gradient path.

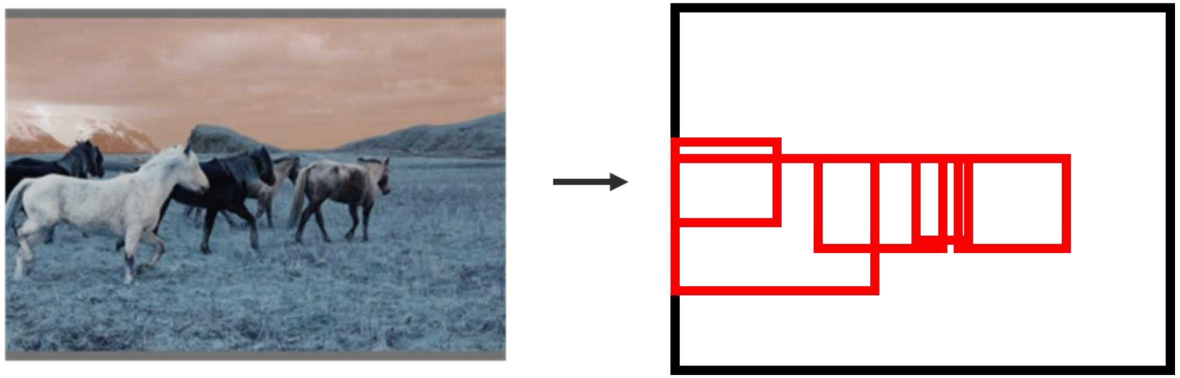
YOLOv9: Optimize both forward path and gradient path.

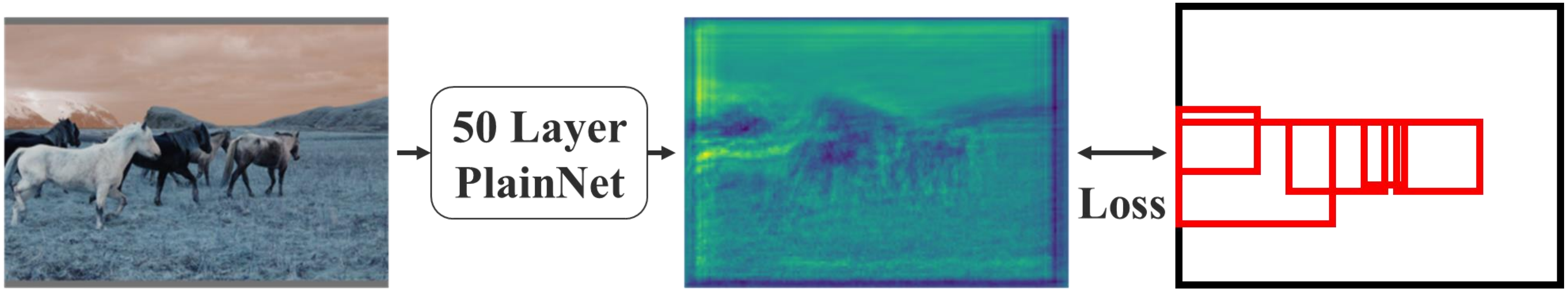
- In YOLOv9, deal with the information bottleneck problem.

Motivation (2/3)

Information Bottleneck of DNNs:

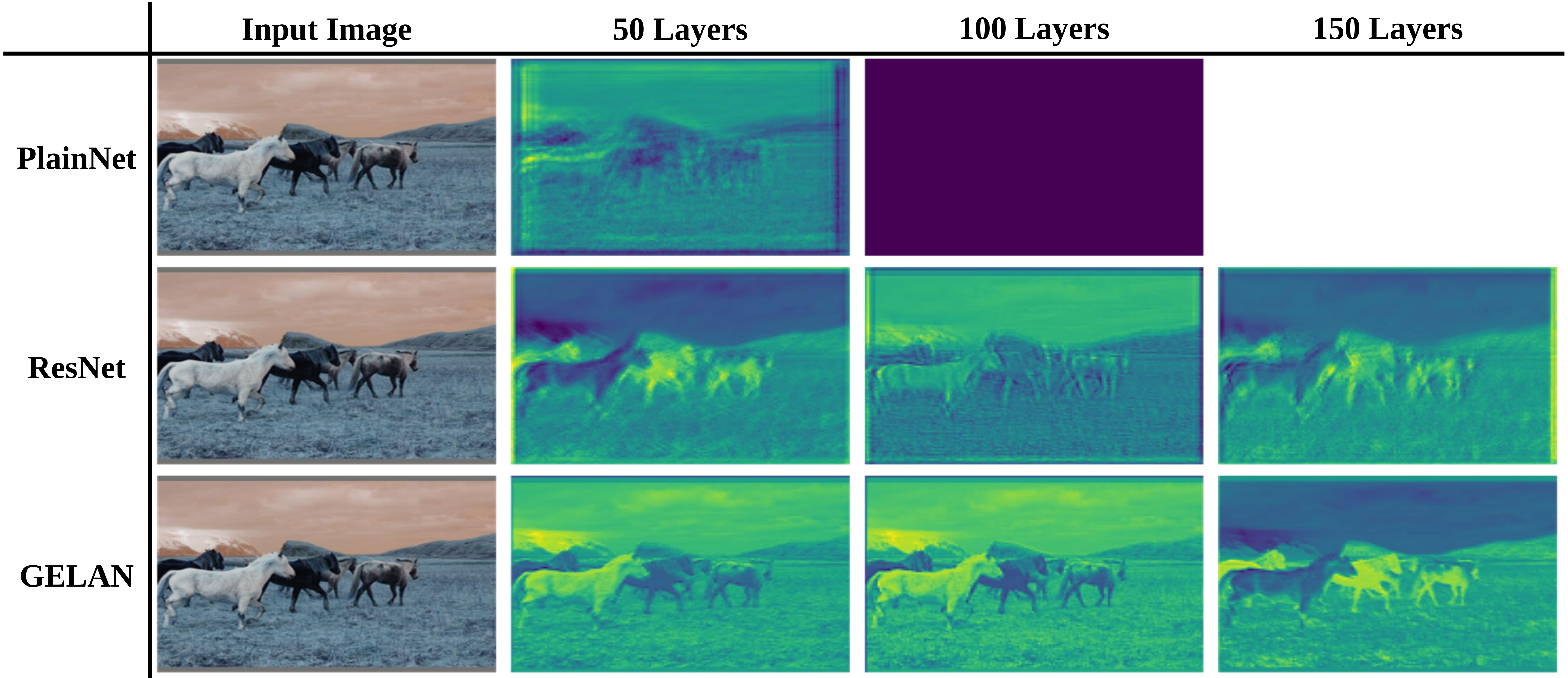
$$I(x; x) \geq I(x; f_0(x)) \geq I(x; f_1(f_0(x))) \geq \ldots \geq I(x; F_\theta(x))$$





Motivation (3/3)

Data loss in different object detection models at different depths



Mainstream Approaches to Solving Information Bottleneck Problems

- *Reversible architectures*

  pros: output data can be restored to input data through reverse calculations

  cons: need extra layers, thus increase inference cost

- *Masked modeling*

  pros: use reconstruction loss to preserve input information

  cons: reconstruction loss may contradict with target loss

- *Deep supervision*

  pros: add prediction heads in shallow layers

  cons: if shallow supervision loses information during training, it will cause considerable error accumulations

Our solutions for information bottleneck

- The design of PGI (Programmable Gradient Information)

  - reversible architecture + deep supervision

- The design of GELAN (Generalized Efficient Layer Aggregation Network)

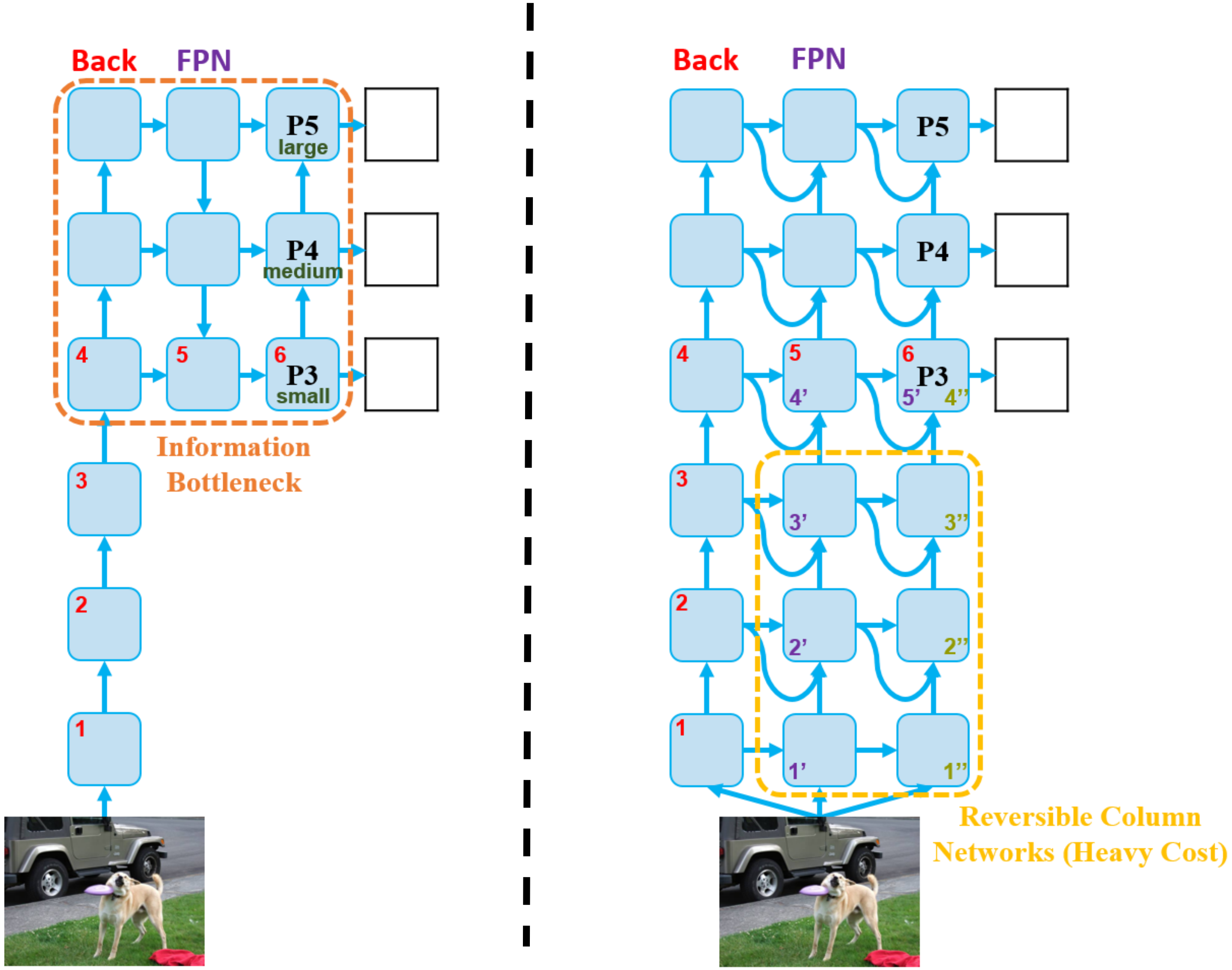
The Design of PGI (1/3)

- Use **auxiliary reversible branch** to make information remains intact when forwarded to the network end
- Use **multi-level auxiliary information** to help learn unbiased information
- Both techniques can be classified as **bag-of-freebies**

Bag-of-freebies
1. Improve accuracy
2. May increase training cost
3. No additional inference cost

The Design of PGI (2/3)

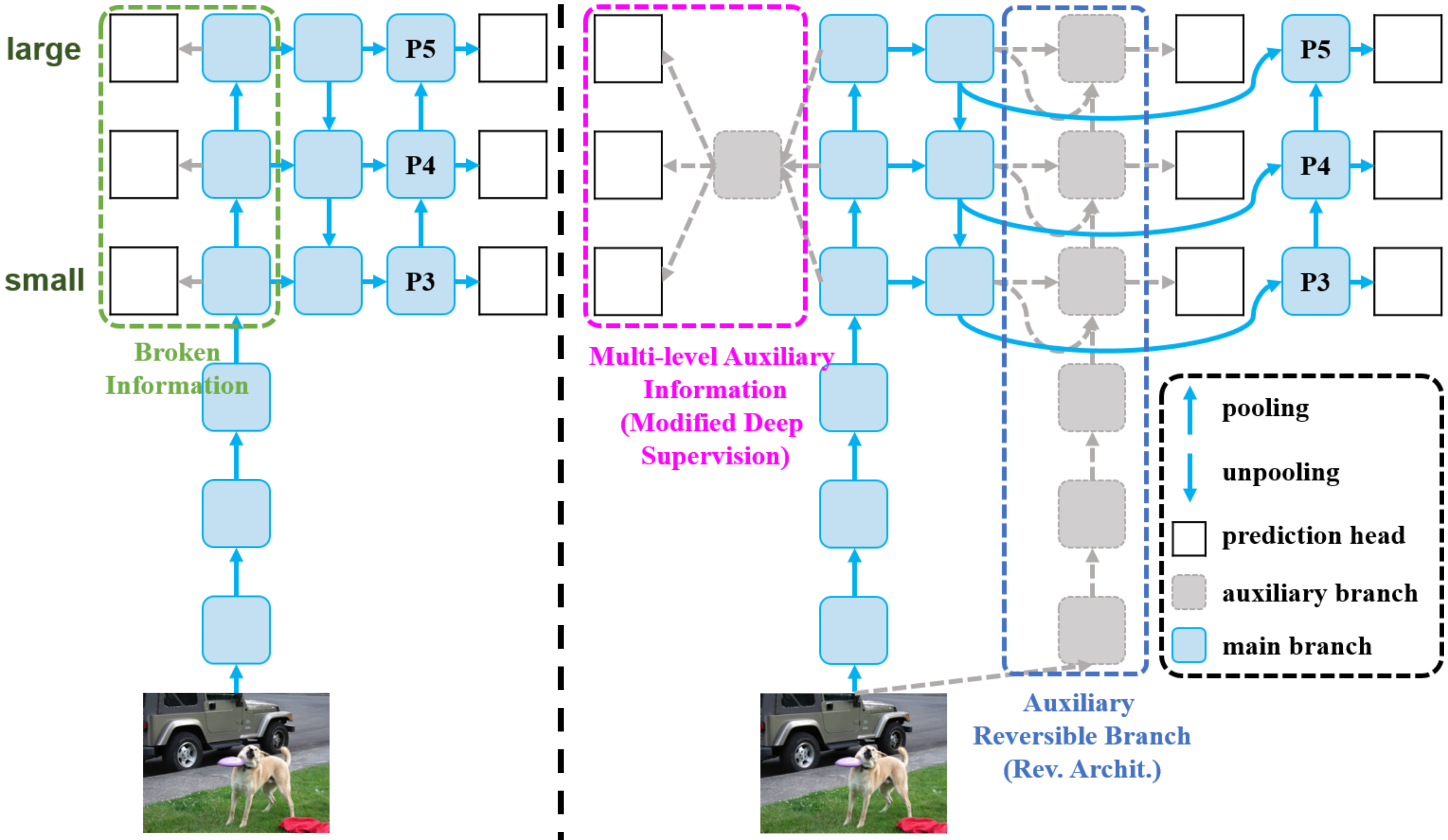
PGI: Reversible Architecture
+ Deep Supervision



(a) PAN [37]

(b) RevCol [3]

The Design of PGI (3/3)



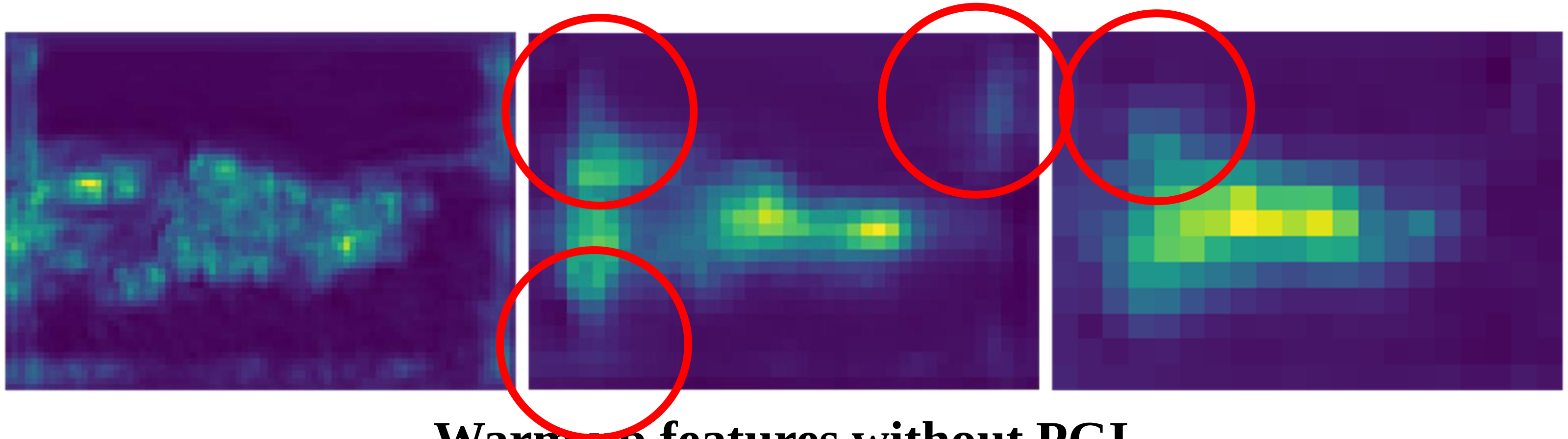
(c) Deep Supervision

(d) Programmable Gradient Information

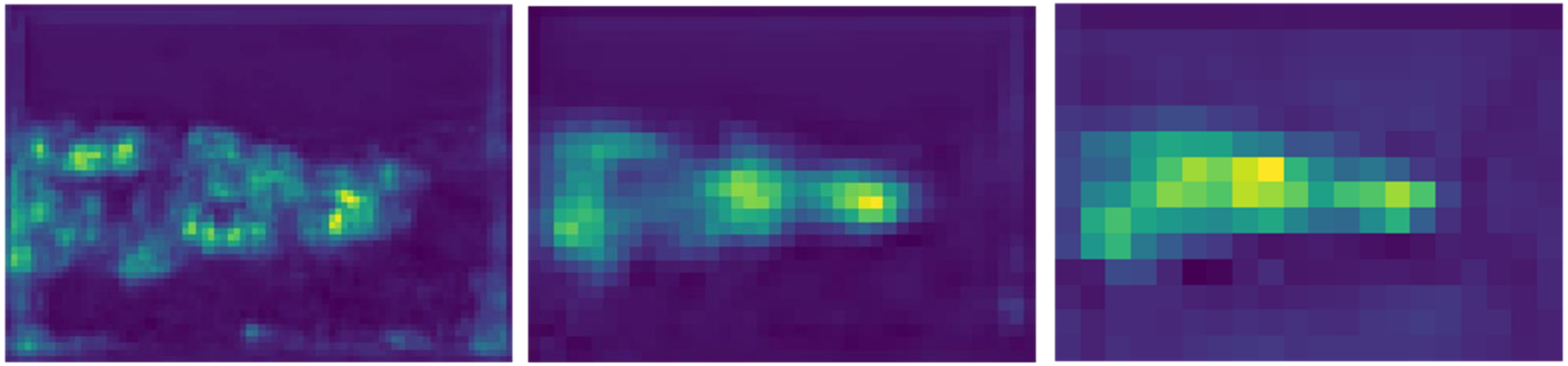
The Power of PGI

Power of PGI (1/3)

Visualize features of warm-up model on object detection task



Warm up features without PGI



Warm up features with PGI

Power of PGI (2/3)

PGI solves the problem of Deep Supervision (DS).

Model	Param. (M)	FLOPs (G)	AP (%)
GELAN-S	7.1	26.4	46.7
+ DS	-	-	46.5 (-0.2)
+ PGI	-	-	**46.8 (+0.1)**
GELAN-C	25.3	102.1	52.5
+ DS	-	-	52.5 (=)
+ PGI	-	-	**53.0 (+0.5)**
GELAN-E	57.3	189.0	55.0
+ DS	-	-	55.3 (+0.3)
+ PGI	-	-	**55.6 (+0.6)**

Power of PGI (3/3)

PGI can be generalized to various models, tasks, and training scheme

Generalize to model scales

COCO det	**YOLOv9-S**	**YOLOv9-M**	**YOLOv9-L**	**YOLOv9-E**
#parameter	7.1M	20.0M	25.3M	57.3M
without PGI	46.7	51.1	52.5	55.0
with PGI	**46.8**	**51.4**	**53.0**	**55.6**

Generalize to various tasks

Multi-task	**Detection**	**Segment**	**Panoptic**	**Caption**
metric	AP^{box}	AP^{box}/AP^{seg}	mIoU/PQ	BLEU4
without PGI	52.5	52.3/42.4	39.0/39.4	38.8
with PGI	**53.0**	**52.9/43.2**	**39.8/40.5**	**39.1**

Generalize to small dataset

VOC det	**YOLOv9-S**	**YOLOv9-S**	**YOLOv8-S**	**YOLOv8-L**
pretrain	-	COCO (PGI)	COCO	COCO
without PGI	64.4	73.5	67.1	73.8
with PGI	**65.1**	**74.4**	**-**	**-**

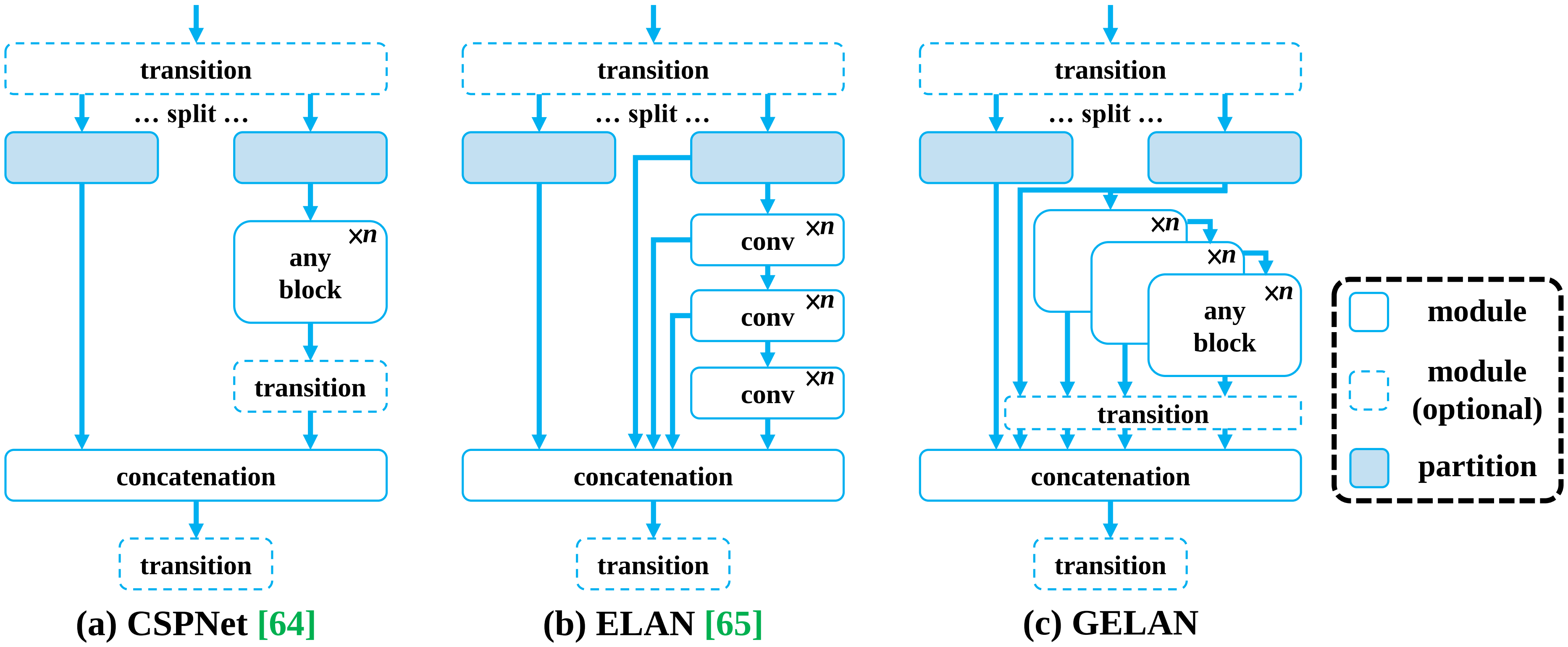
Design of GELAN: Generalized Efficient Layer Aggregation Network (1/2)

GELAN	=	CSPNet	[Wang et al. 2019]
	+	ELAN	[Wang et al. 2022]

Characteristics of GELAN

1. compatible to various existing modern networks
2. has extremely high parameter utilization
3. has excellent inference speed on various devices

Design of GELAN:Generalized Efficient Layer Aggregation Network (2/2)



Power of GELAN

Power of GELAN

GELAN can be generalized to various models and has high inference speed

	MG YOLOv9	LH YOLOv9	YOLOv9 TR	YOLOv9 Lite	YOLOv9 Light
#Parameter	25.3M	21.1M	14.1M	13.3M	2.5M
FLOPs	102.1G	82.5G	67.5G	66.7G	11.0G
mAP	**53.3%**	**52.9%**	**53.1%**	**52.7%**	**44.1%**

Generalize to various models: mask-guided YOLOv9 (MG YOLOv9), light head YOLOv9 (LH YOLOv9), YOLOv9 with Transformer (YOLOv9 TR), YOLOv9 using hybrid convolution (YOLOv9 Lite), and YOLOv9 using depth-wise convolution (YOLOv9 Light).

	YOLOv6-L 3.0	YOLOv7 AF	YOLOv8-L	YOLOv9-C	YOLOv9-C-TR
Latency	7.9ms	6.7ms	8.1ms	**6.1ms**	**5.9ms**
mAP	51.8/52.8^{distill}	**53.0**	52.9	**53.0**	**53.1**

GELAN has very high inference speed, it about 25% faster than YOLOv8.

Results

Results

YOLOv9 has strong ability on multi-task applications

PGI makes YOLOv9 outperforms SOTA methods in all aspects.

ECCV’24 (Oral)

Model	#Param.	AP^{box}	AP^{mask}	$mIoU^{sem}$	$mIoU^{stuff}$	PQ^{pan}	$BLEU4^{cap}$	Acc^{gnd}
GiT-B	131M	46.7	31.9	47.8^{*}	-	-	35.4^{+}	85.8
GiT-L	387M	51.3	35.1	50.6^{*}	-	-	35.7^{+}	88.4
GiT-H	756M	**52.9**	35.8	**52.4^{*}**	-	-	36.2^{+}	89.2
YOLOv9	**45.5M**	52.2	**42.9**	49.4	**56.8**	**42.2**	**39.4**	-

17× lighter

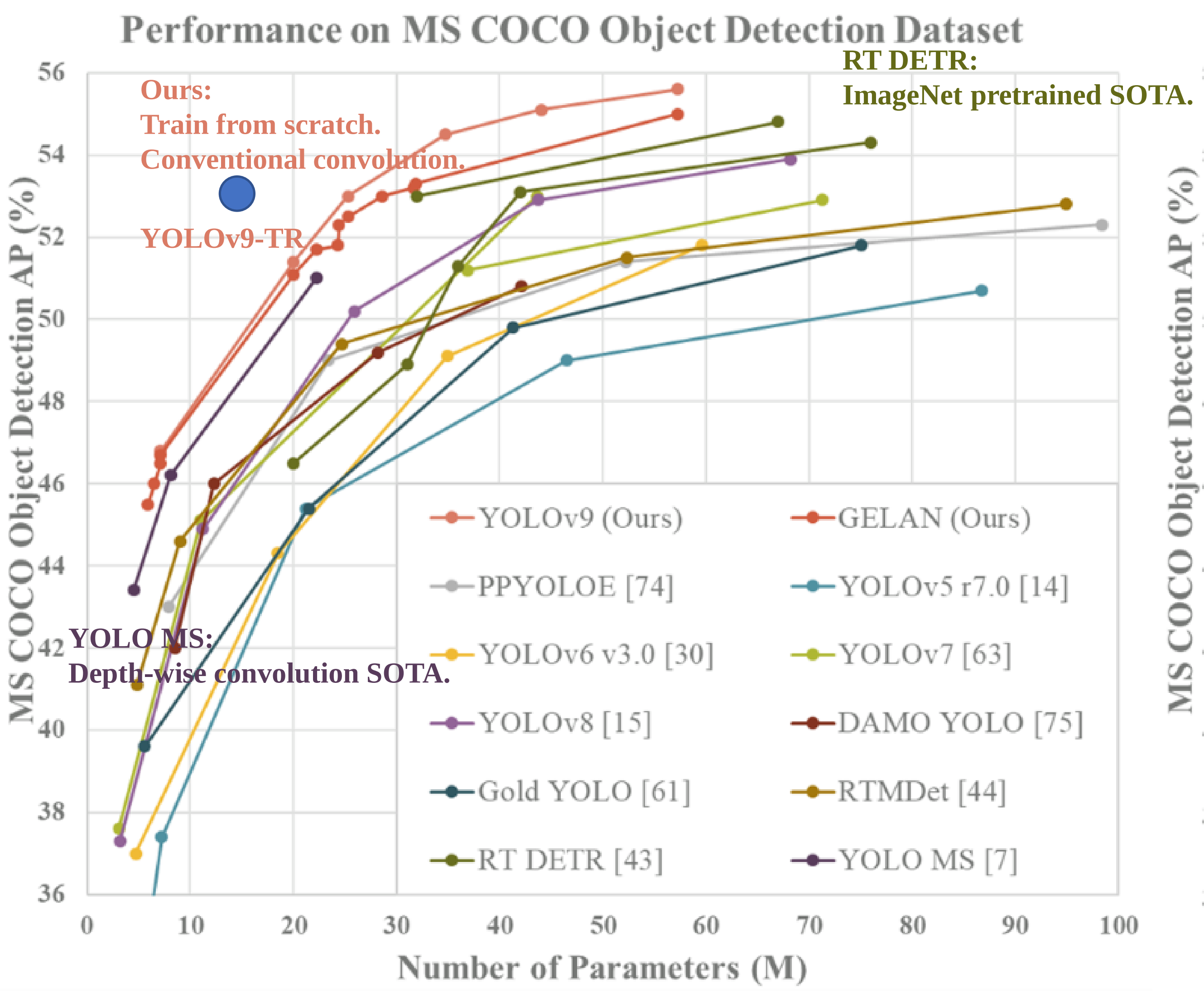
ECCV’24 (Oral)

Model	$Time^{total}$	$Time^{box}$	$Time^{mask}$	$Time^{sem}$	$Time^{stuff}$	$Time^{pan}$	$Time^{cap}$	$Time^{gnd}$
GiT-B	2589ms	359	1149	717	-	-	272	92
GiT-L	5320ms	689	2451	1617	-	-	424	139
GiT-H	8321ms	1053	3838	2703	-	-	550	177
YOLOv9	**61.5ms**	**61.5**						-

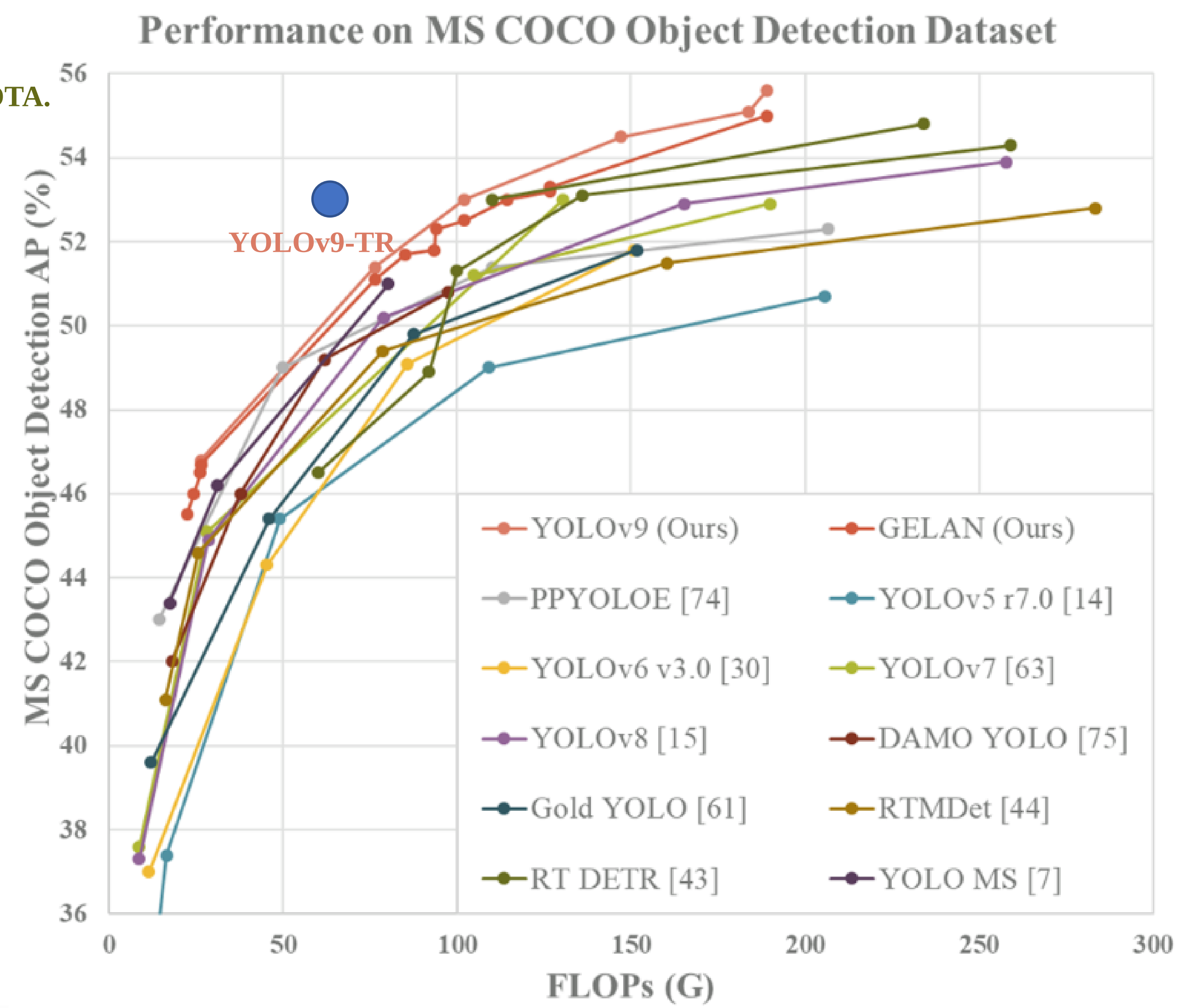
135× faster

YOLOv9 gets all predictions in one shot inference

Results

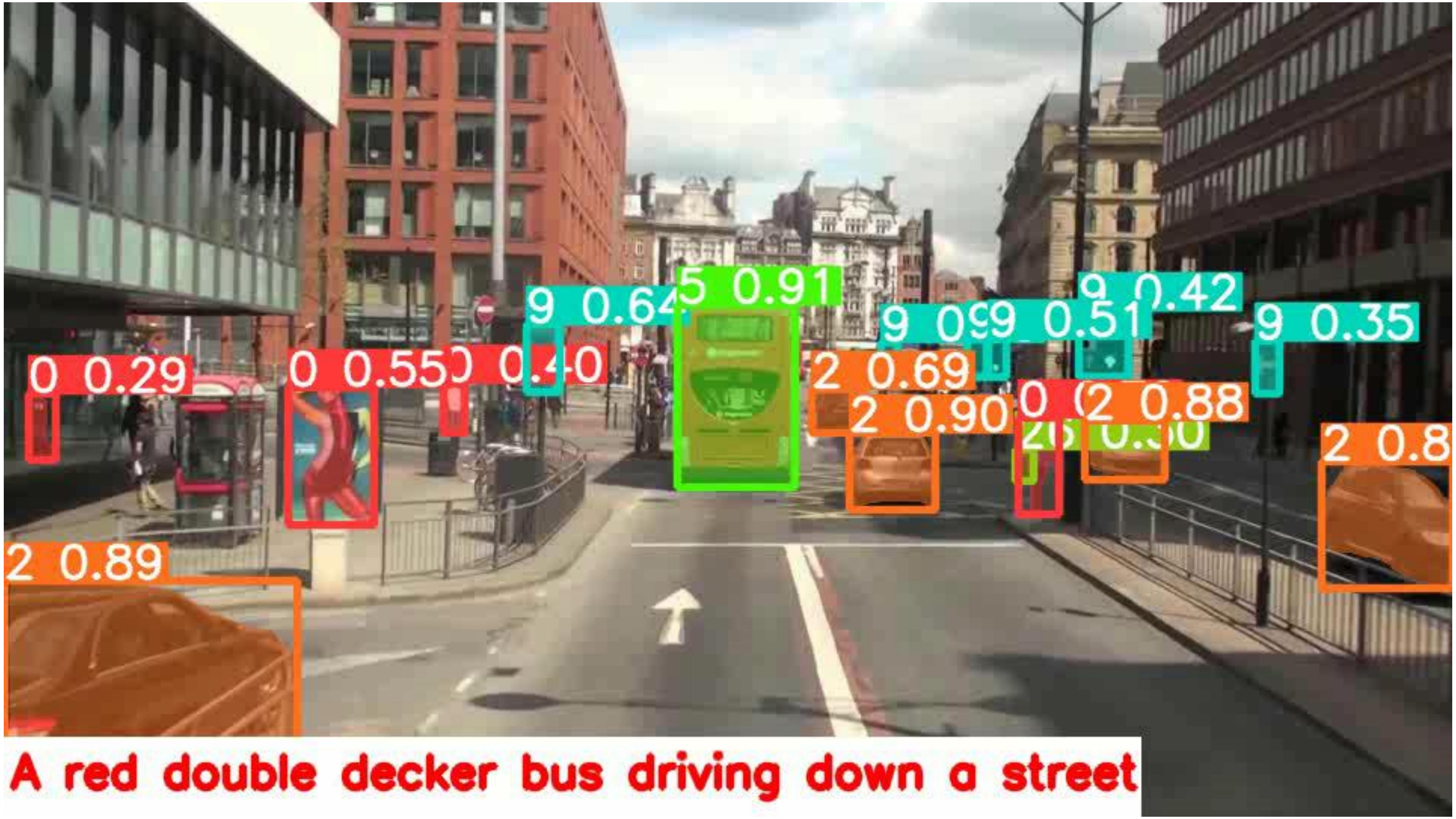


Performance on MS COCO Object Detection Dataset
Ours:
Train from scratch.
Conventional convolution.
YOLOv9-TR
RT DETR:
ImageNet pretrained SOTA.
YOLO MS:
Depth-wise convolution SOTA.
MS COCO Object Detection AP (%)
Number of Parameters (M)
YOLOv9 (Ours)
GELAN (Ours)
PPYOLOE [74]
YOLOv5 r7.0 [14]
YOLOv6 v3.0 [30]
YOLOv7 [63]
YOLOv8 [15]
DAMO YOLO [75]
Gold YOLO [61]
RTMDet [44]
RT DETR [43]
YOLO MS [7]



Performance on MS COCO Object Detection Dataset
YOLOv9-TR
MS COCO Object Detection AP (%)
FLOPs (G)
YOLOv9 (Ours)
GELAN (Ours)
PPYOLOE [74]
YOLOv5 r7.0 [14]
YOLOv6 v3.0 [30]
YOLOv7 [63]
YOLOv8 [15]
DAMO YOLO [75]
Gold YOLO [61]
RTMDet [44]
RT DETR [43]
YOLO MS [7]

Generalist YOLO



Conclusions

- propose a trustworthy AI technology to make a model generate and learn from reliable gradient information
- design an efficient networks which is generalized various architectures and has low inference latency.
- 3. show the proposed trustworthy AI technology is generalized to various models, tasks, and training scheme.
- 4. show the proposed framework will bring real-time computer vision systems to a new achievement.

Thanks

Q&A