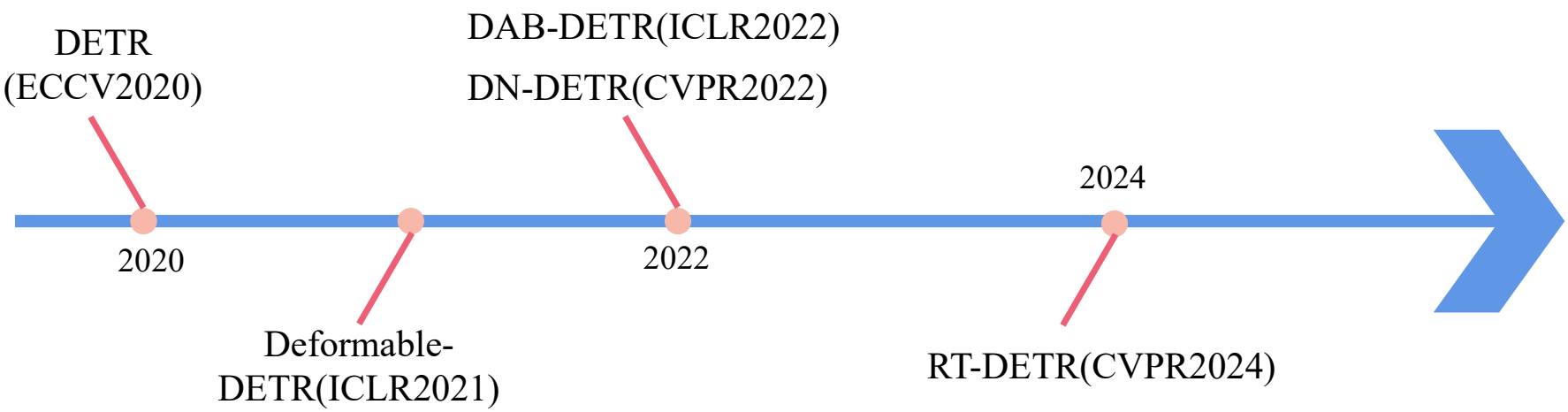
模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

In-depth Introduction to the DETRs

欣子豪
2024.11.11

Contents



- Conditional DETR
- Anchor DETR
- DINO
- Efficient DETR
- Sparse DETR
- Lite DETR
- DINO
- . . .

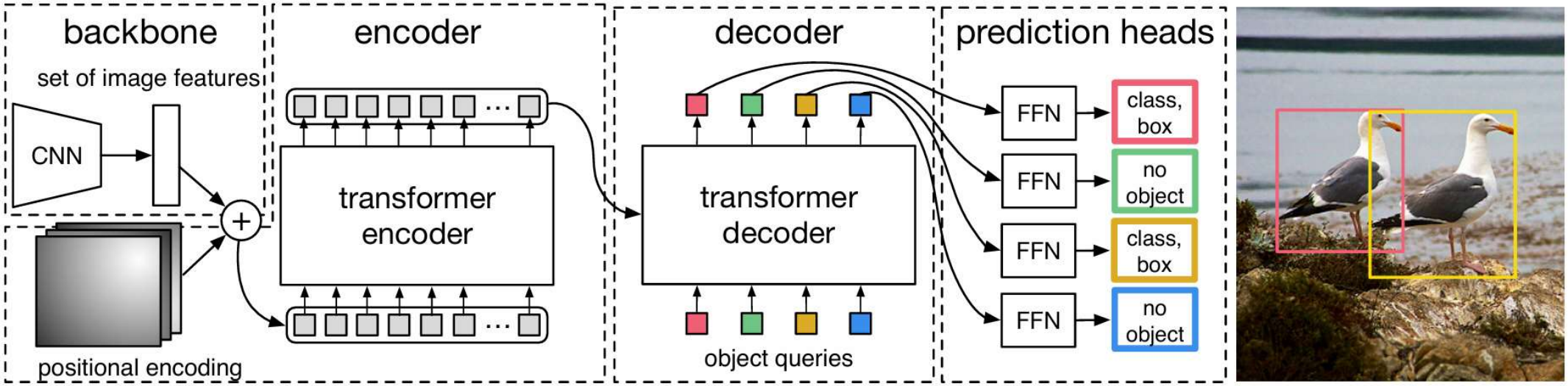
End-to-End Object Detection with Transformers

Nicolas Carion*, Francisco Massa*, Gabriel Synnaeve, Nicolas Usunier,
Alexander Kirillov, and Sergey Zagoruyko

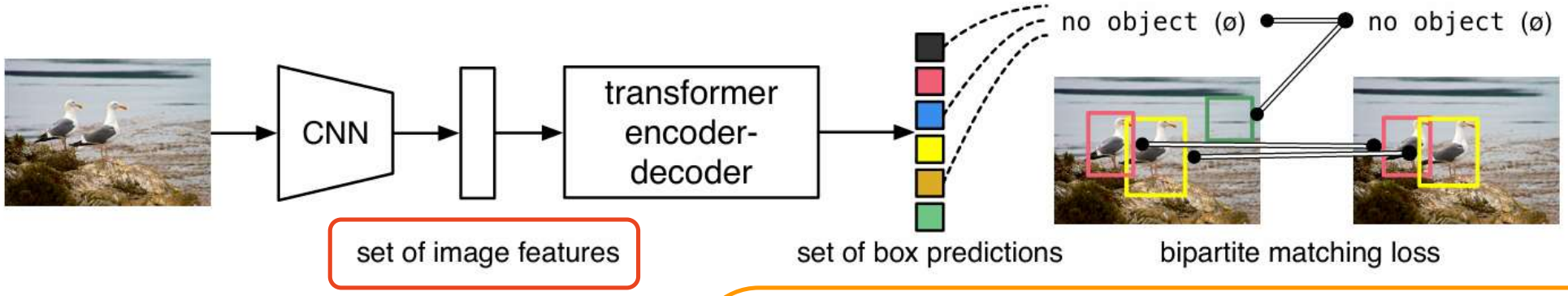
Facebook AI

ECCV 2020

DETR


backbone
set of image features
CNN
positional encoding
encoder
transformer encoder
decoder
transformer decoder
object queries
prediction heads
FFN
FFN
FFN
FFN
class, box
no object
class, box
no object

DETR

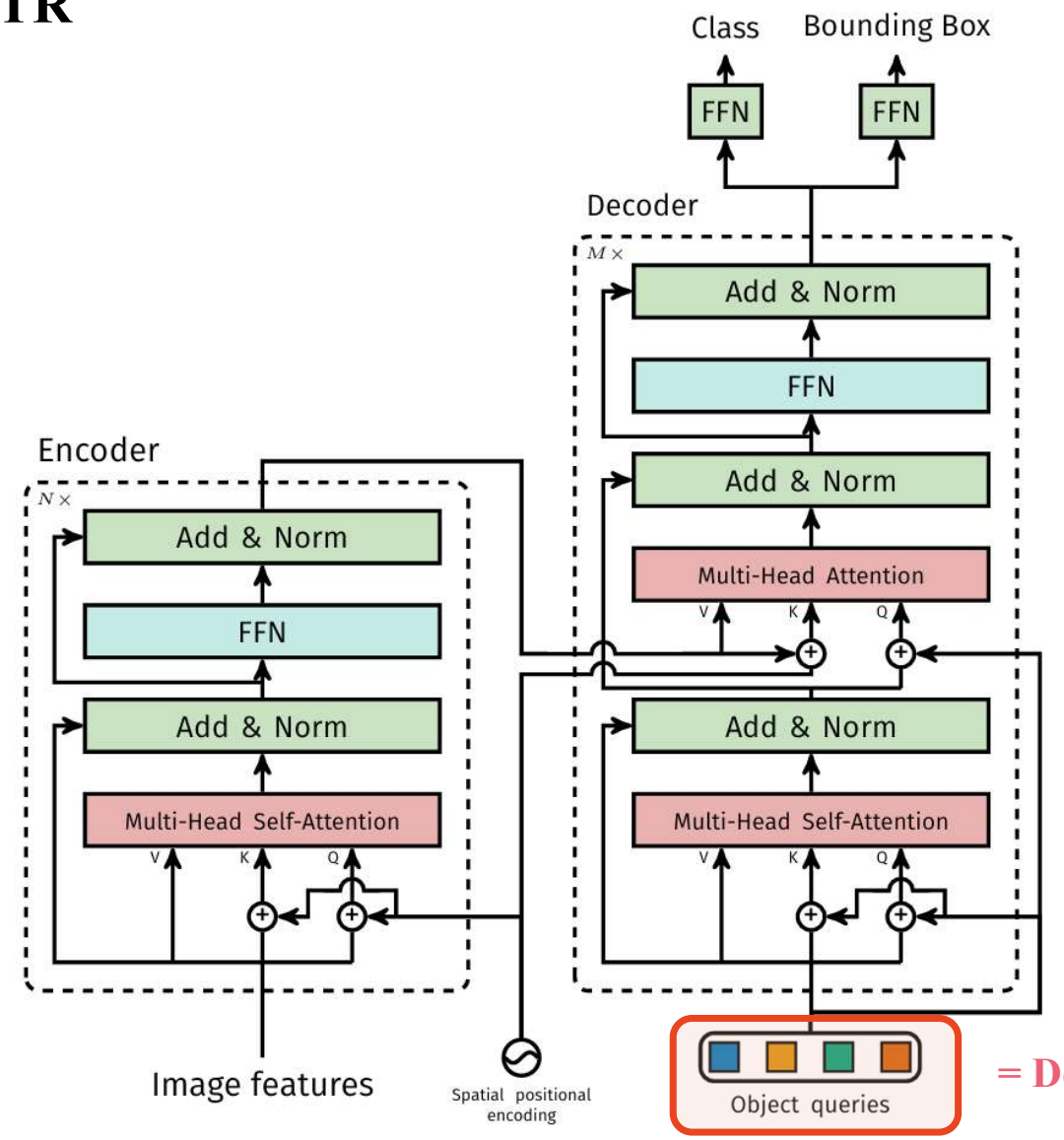


Set Prediction

$$\hat{\sigma} = \arg\min_{\sigma \in \mathfrak{S}_N} \sum_{i}^{N} \mathcal{L}_{\text{match}}(y_i, \hat{y}_{\sigma(i)})$$

$$\mathcal{L}_{\text{Hungarian}}(y, \hat{y}) = \sum_{i=1}^{N} \left[-\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}}(i)) \right]$$

DETR

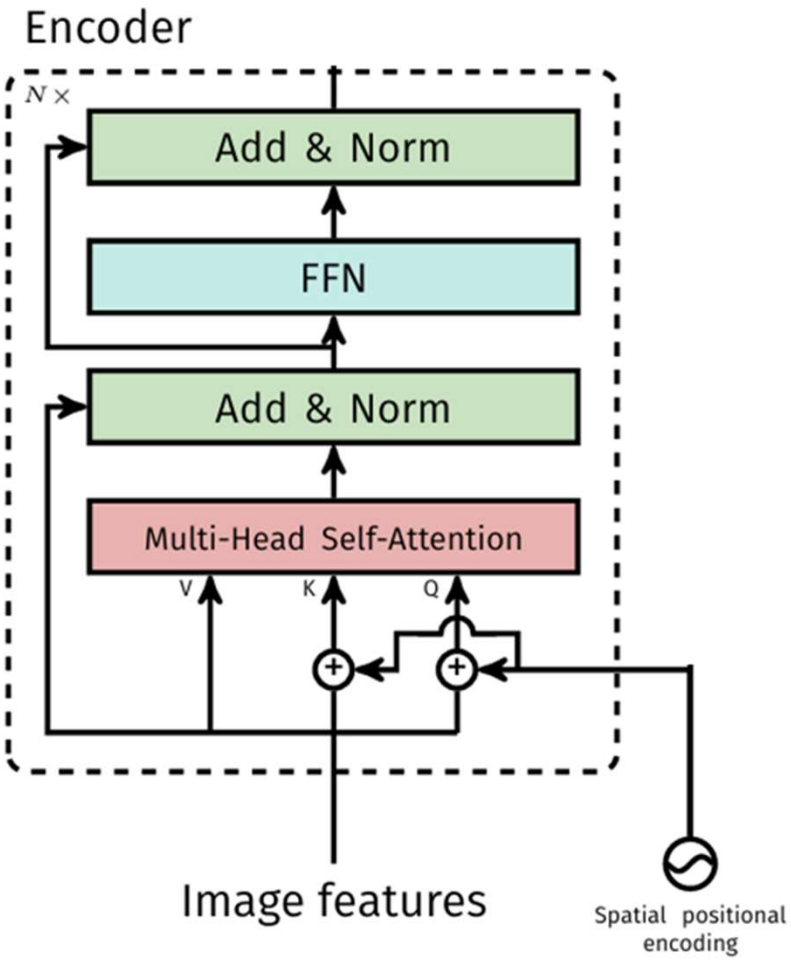


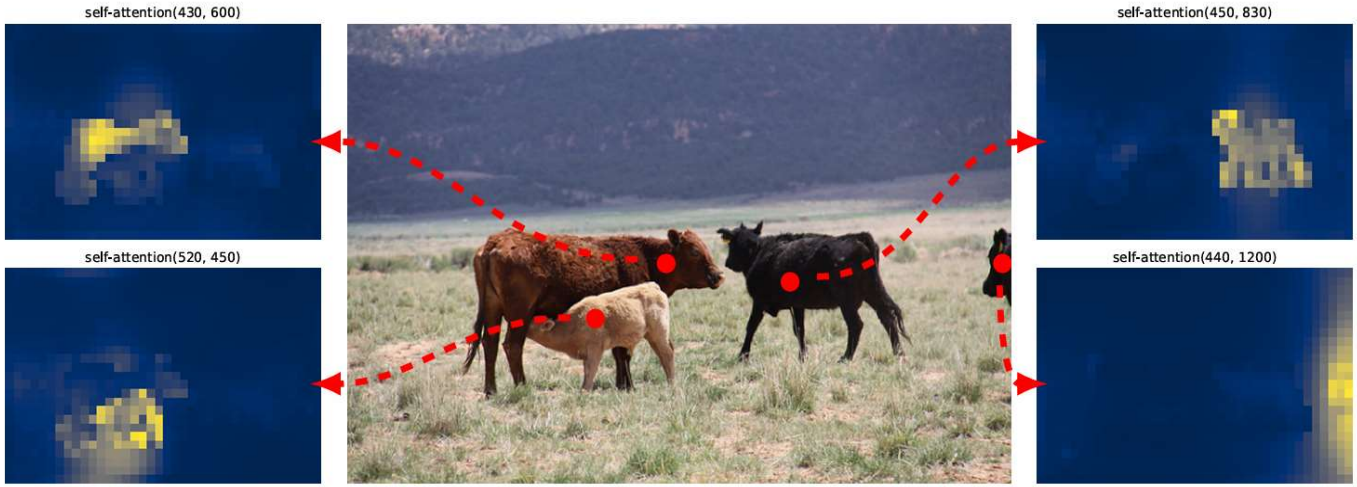
FFN

FC + Act + Dropout + FC + Dropout

= Decoder Embedding + Learnable queries

DETR

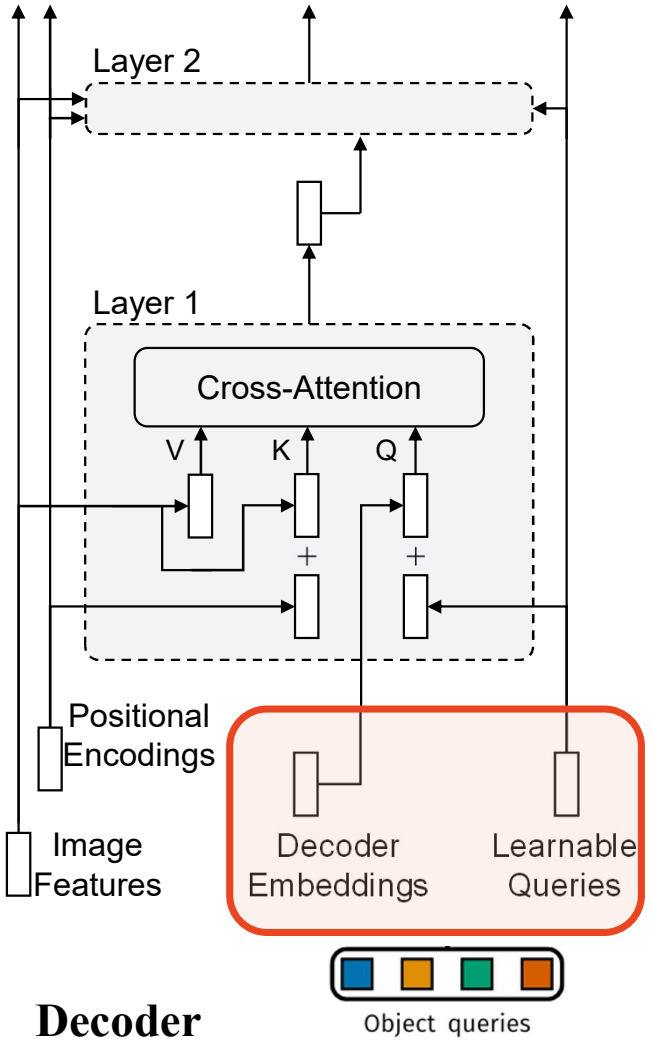




Encoder self-attention for a set of reference points.

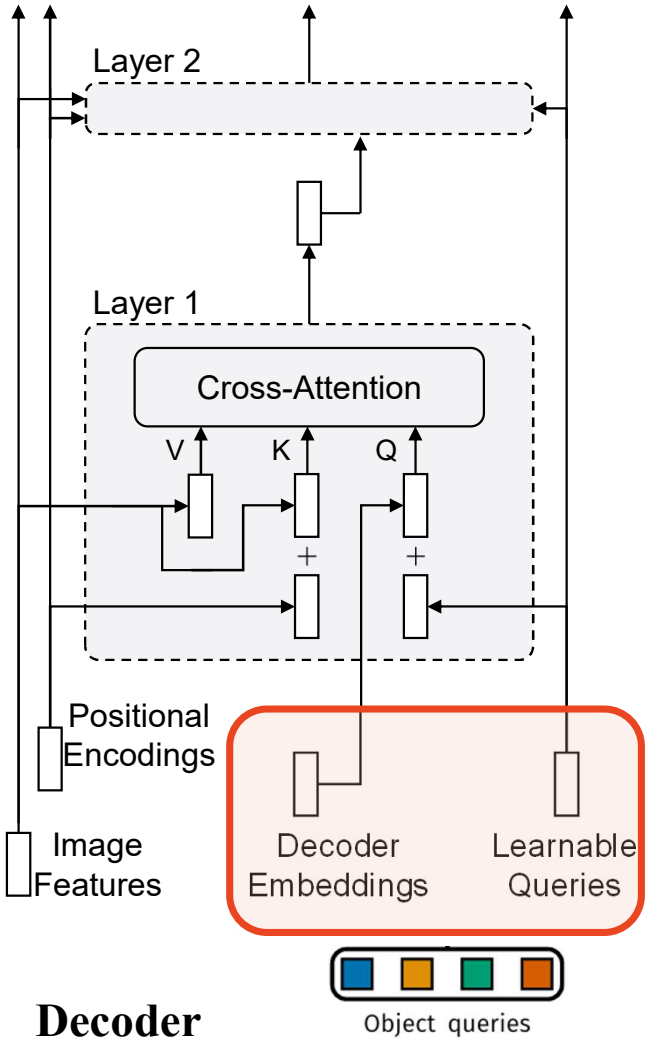
The encoder is able to separate individual instances.

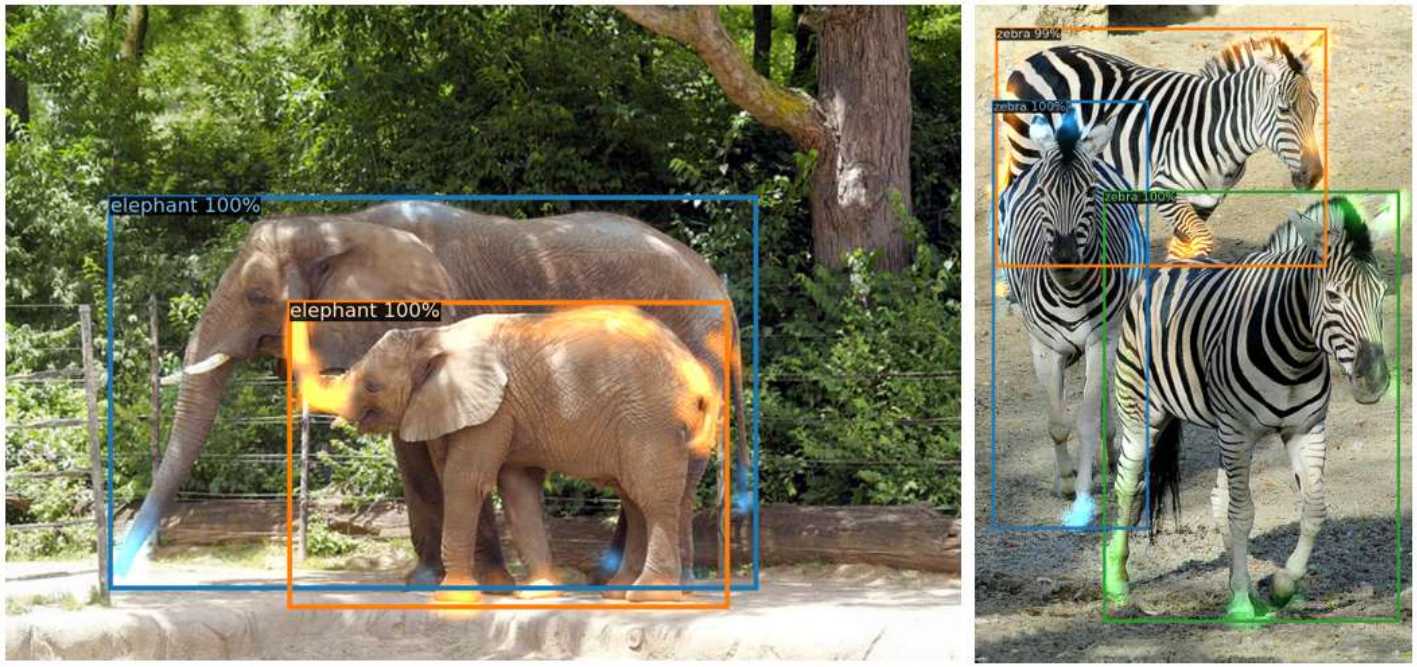
DETR



spatial pos. enc. encoder	spatial pos. enc. decoder	output pos. enc. decoder	AP	Δ	AP_{50}	Δ
none	none	learned at input	32.8	-7.8	55.2	-6.5
sine at input	sine at input	learned at input	39.2	-1.4	60.0	-1.6
learned at attn.	learned at attn.	learned at attn.	39.6	-1.0	60.7	-0.9
none	sine at attn.	learned at attn.	39.3	-1.3	60.3	-1.4
sine at attn.	sine at attn.	learned at attn.	**40.6**	-	**61.6**	-

DETR

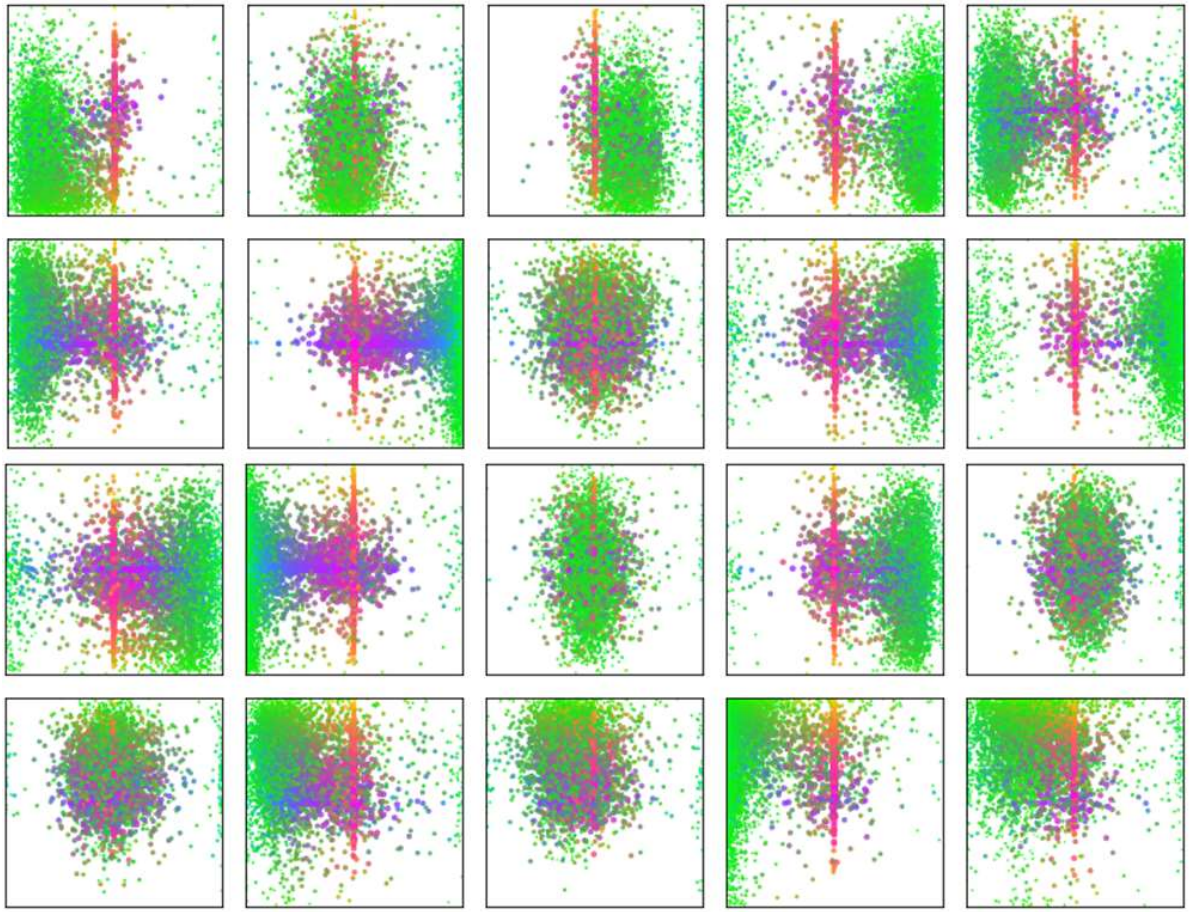




Visualizing decoder attention for every predicted object.

Decoder typically attends to object extremities!

DETR



Visualization of all box predictions on all images from COCO 2017 val set for 20 out of total N = 100 prediction slots in DETR decoder.

Small boxes

Large horizontal boxes

Large vertical boxes

Each slot (query) learns to specialize on certain areas and box sizes with several operating modes!

DETR

Model	GFLOPS/FPS	#params	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster RCNN-DC5	320/16	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster RCNN-FPN	180/26	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster RCNN-R101-FPN	246/20	60M	42.0	62.5	45.9	25.2	45.6	54.6
Faster RCNN-DC5+	320/16	166M	41.1	61.4	44.3	22.9	45.9	55.0
Faster RCNN-FPN+	180/26	42M	42.0	62.1	45.5	26.6	45.4	53.4
Faster RCNN-R101-FPN+	246/20	60M	44.0	63.9	**47.8**	**27.2**	48.1	56.0
DETR	86/28	41M	42.0	62.4	44.2	20.5	45.8	61.1
DETR-DC5	187/12	41M	43.3	63.1	45.9	22.5	47.3	61.1
DETR-R101	152/20	60M	43.5	63.8	46.4	21.9	48.0	61.8
DETR-DC5-R101	253/10	60M	**44.9**	**64.7**	47.7	23.7	**49.5**	**62.3**

Drawbacks:

1. Slow convergence (500 epoch)
2. Limited feature spatial resolution

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

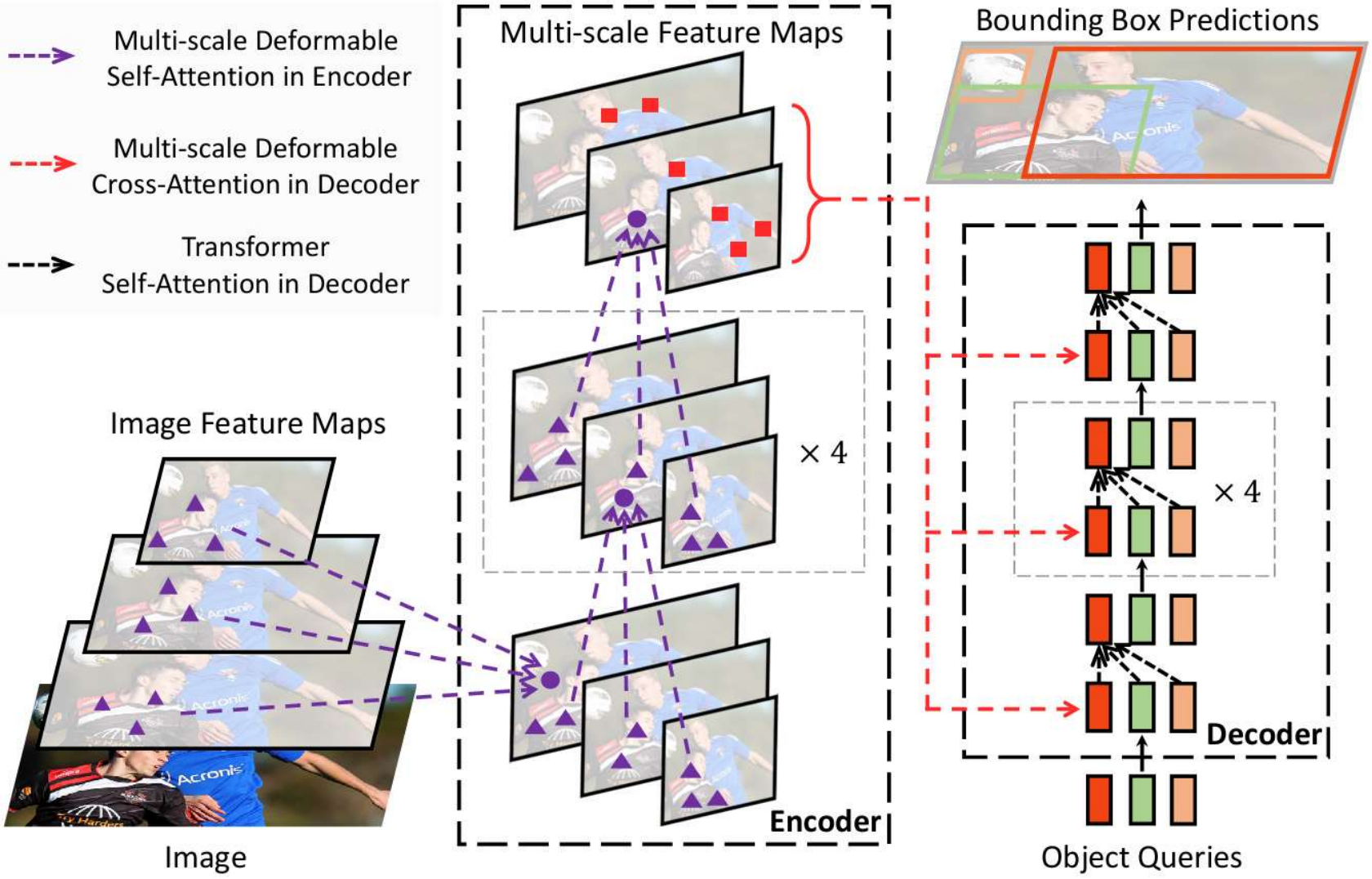
ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

Deformable DETR: Deformable Transformers For End-to-End Object Detection

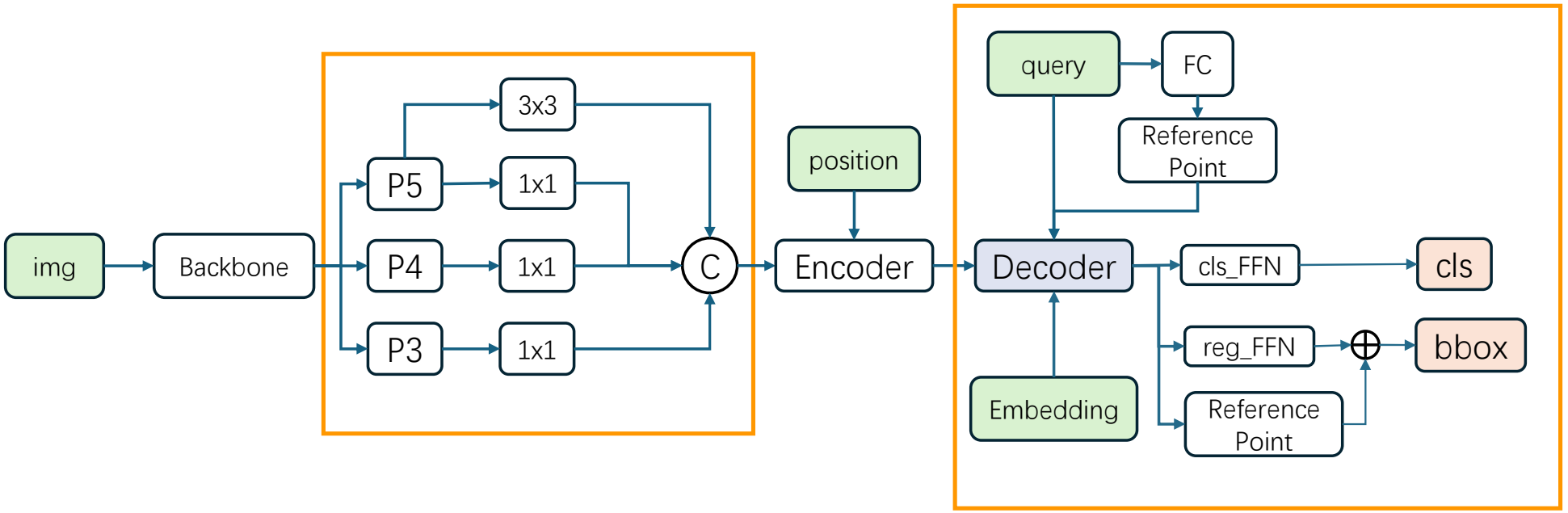
Xizhou Zhu[1*], **Weijie Su**[2*‡], **Lewei Lu**[1], **Bin Li**[2], **Xiaogang Wang**[1,3], **Jifeng Dai**[1†]
[1]SenseTime Research
[2]University of Science and Technology of China
[3]The Chinese University of Hong Kong

ICLR 2021

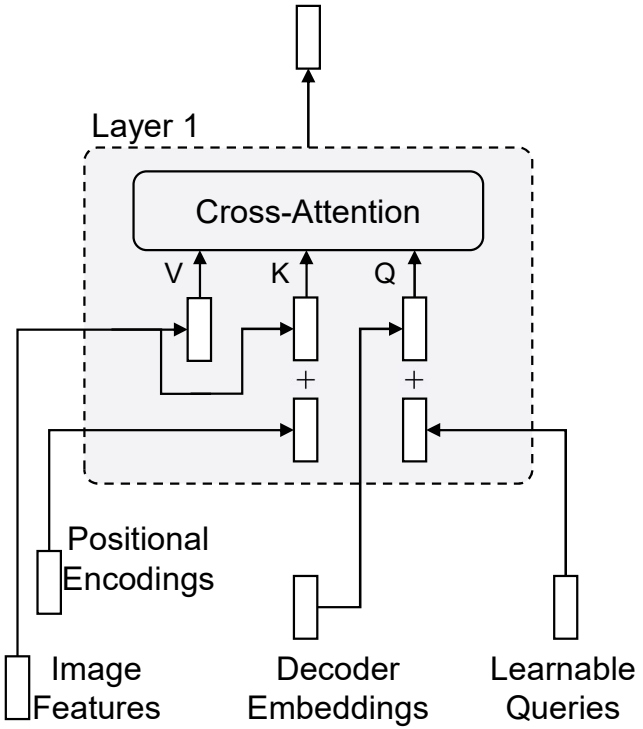
Deformable-DETR



Deformable-DETR



Deformable-DETR



DETR Decoder

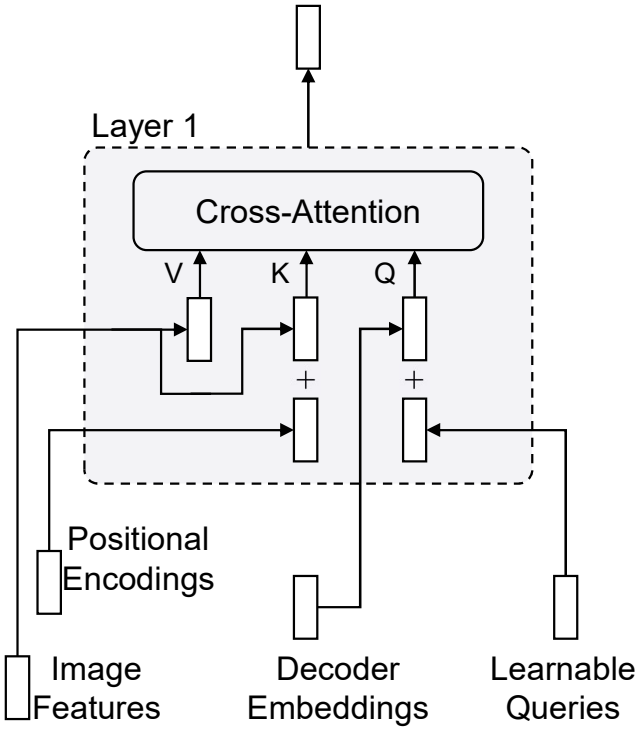
query element　key elements　Learnable weights

$$\mathrm{MultiHeadAttn}(\boldsymbol{z}_q, \boldsymbol{x}) = \sum_{m=1}^{M} \boldsymbol{W}_m \Big[\sum_{k \in \Omega_k} A_{mqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}_k \Big]$$

Sum over attention heads　Attention weights　Key feature

subjective to $\sum_{k \in \Omega_k} A_{mqk} = 1$

Deformable-DETR



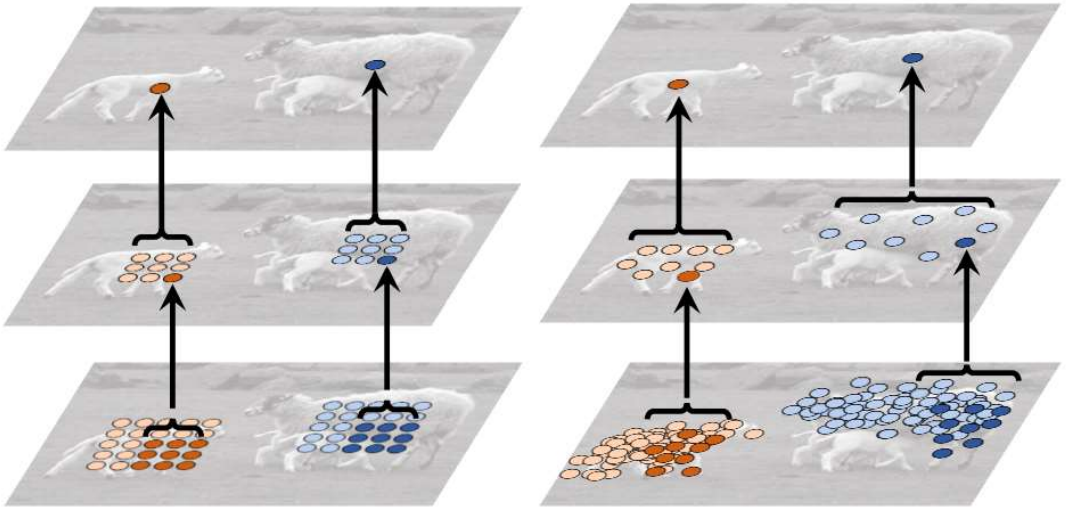
DETR Decoder

$$\text{MultiHeadAttn}(\boldsymbol{z}_q, \boldsymbol{x}) = \sum_{m=1}^{M} \boldsymbol{W}_m \Big[\sum_{k \in \Omega_k} A_{mqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}_k \Big]$$

$A_{mqk} \approx \frac{1}{N_k}$ at initialization, which leads to ambiguous gradients for inputs.

The core issue is Transformer attention would look overall possible spatial locations

Deformable-DETR



(a) standard convolution (b) deformable convolution

$$\text{MultiHeadAttn}(\boldsymbol{z}_q, \boldsymbol{x}) = \sum_{m=1}^{M} \boldsymbol{W}_m \big[\sum_{k \in \Omega_k} A_{mqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}_k \big]$$

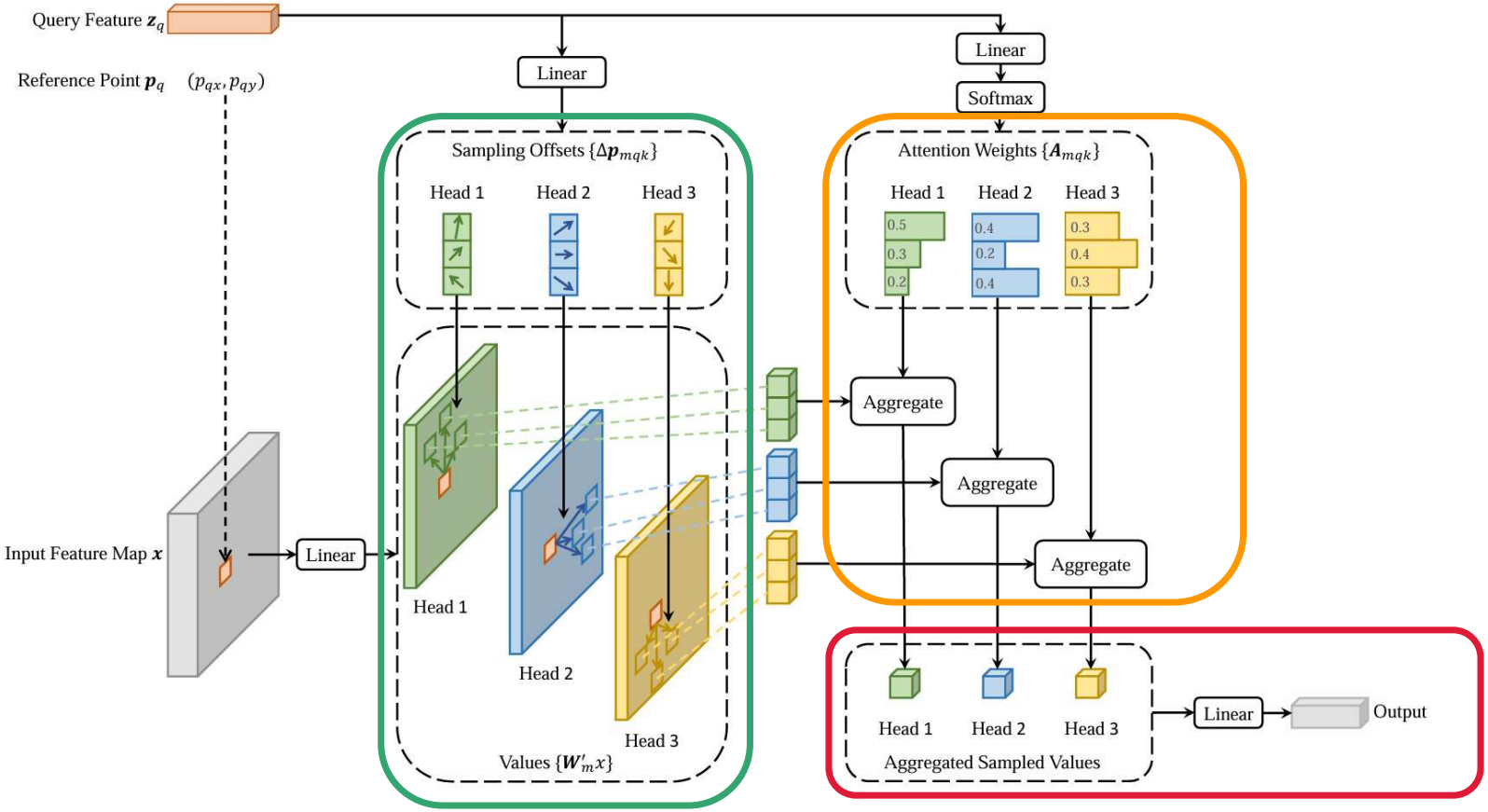
$$\text{DeformAttn}(\boldsymbol{z}_q, \boldsymbol{p}_q, \boldsymbol{x}) = \sum_{m=1}^{M} \boldsymbol{W}_m \big[\sum_{k=1}^{K} A_{mqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}(\boldsymbol{p}_q + \Delta \boldsymbol{p}_{mqk}) \big]$$

reference point

Sparsely sampled key feature

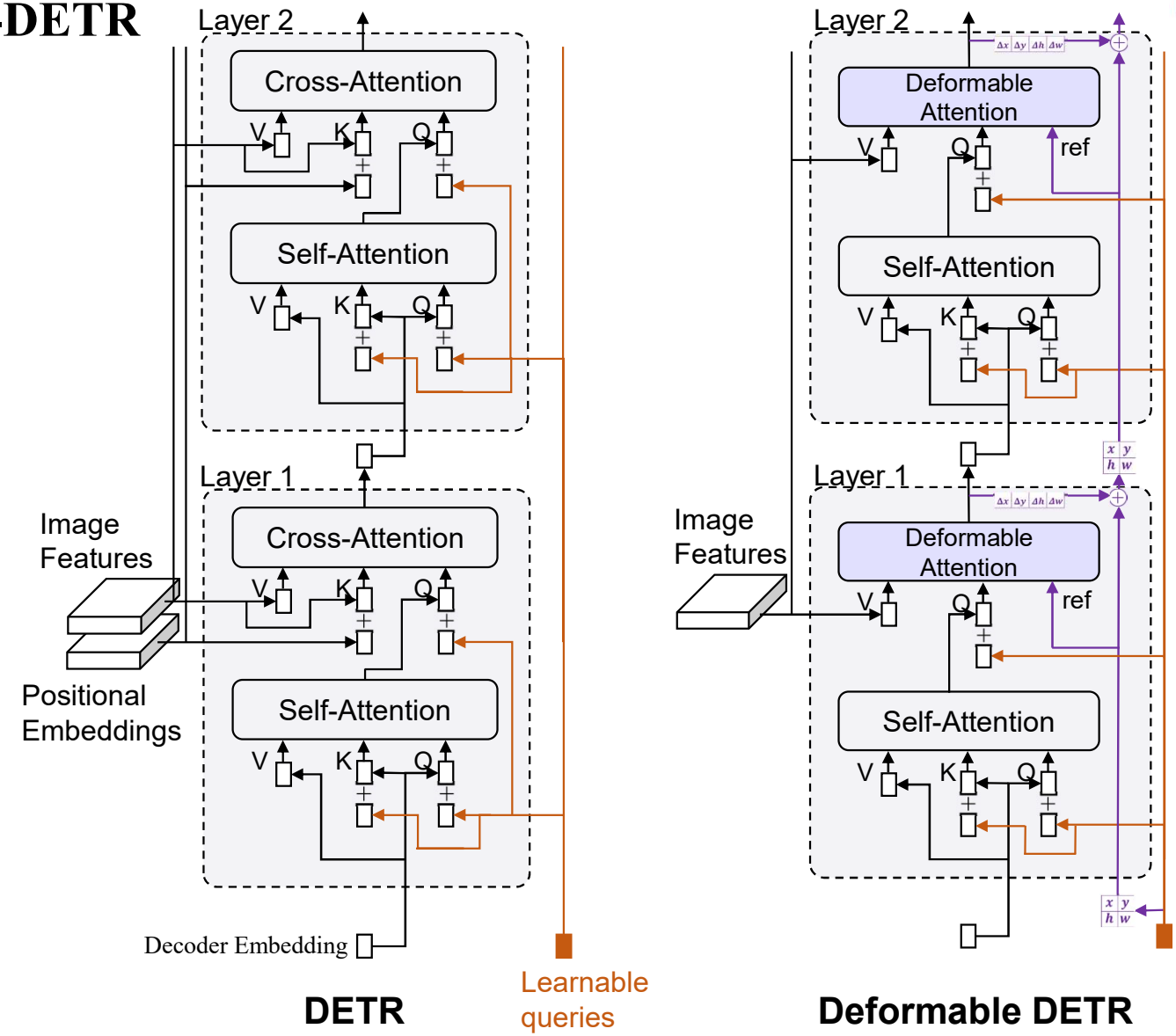
$$\text{MSDeformAttn}(\boldsymbol{z}_q, \hat{\boldsymbol{p}}_q, \{\boldsymbol{x}^l\}_{l=1}^{L}) = \sum_{m=1}^{M} \boldsymbol{W}_m \big[\sum_{l=1}^{L} \sum_{k=1}^{K} A_{mlqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}^l(\phi_l(\hat{\boldsymbol{p}}_q) + \Delta \boldsymbol{p}_{mlqk}) \big]$$

Deformable-DETR

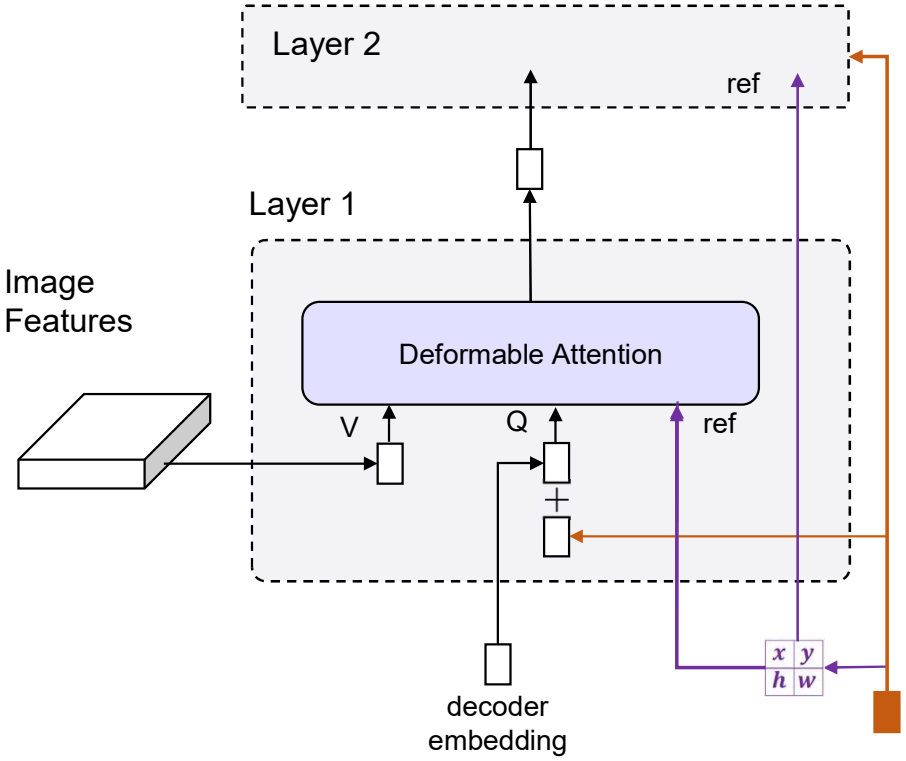


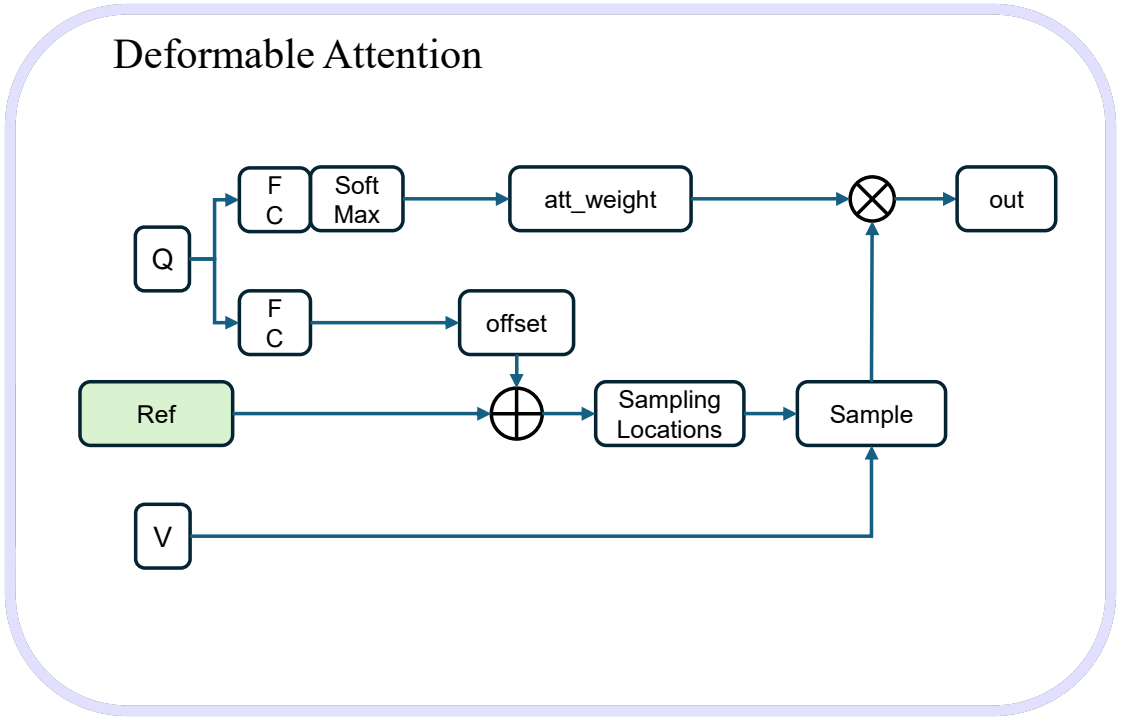
$$\text{DeformAttn}(\boldsymbol{z}_q, \boldsymbol{p}_q, \boldsymbol{x}) = \sum_{m=1}^{M} \boldsymbol{W}_m \left[\sum_{k=1}^{K} A_{mqk} \cdot \boldsymbol{W}'_m \boldsymbol{x}(\boldsymbol{p}_q + \Delta \boldsymbol{p}_{mqk}) \right]$$

Deformable-DETR

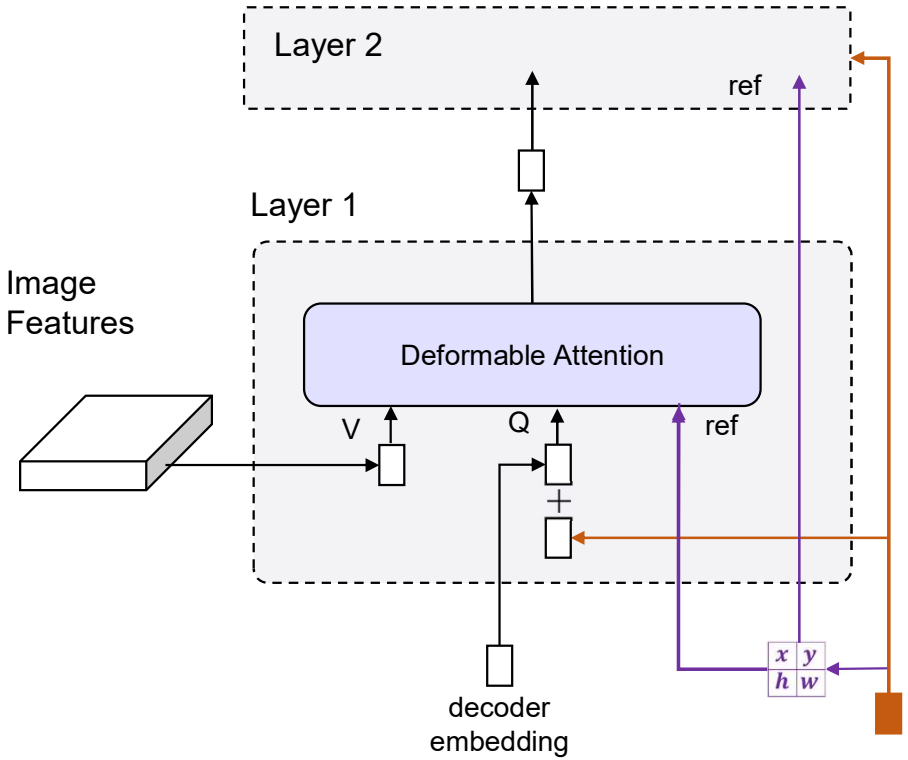


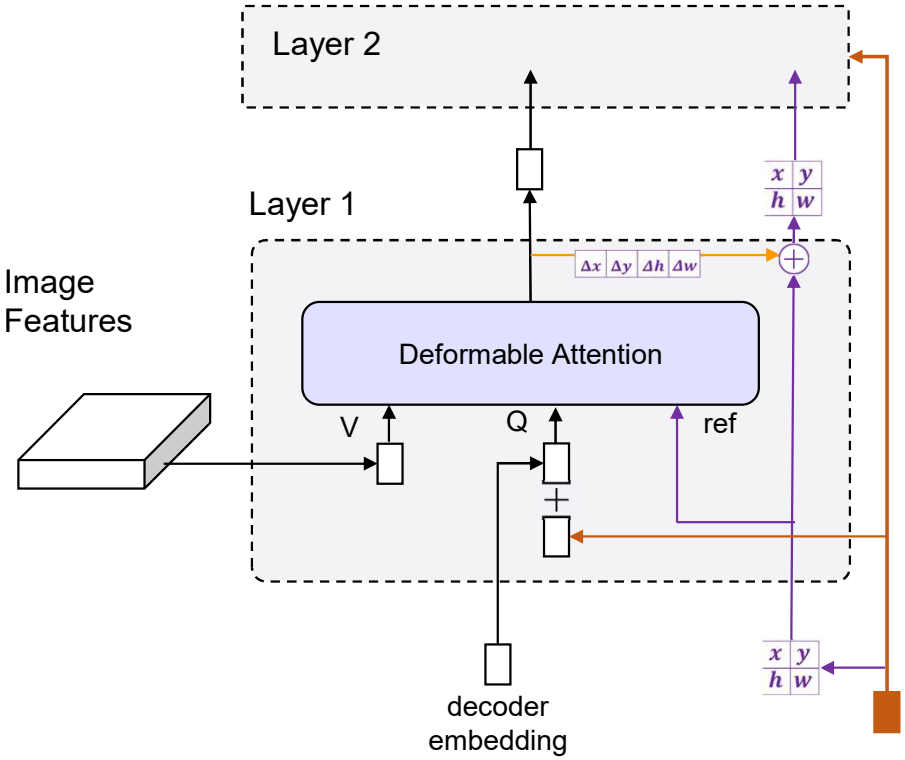
Deformable-DETR





Deformable-DETR

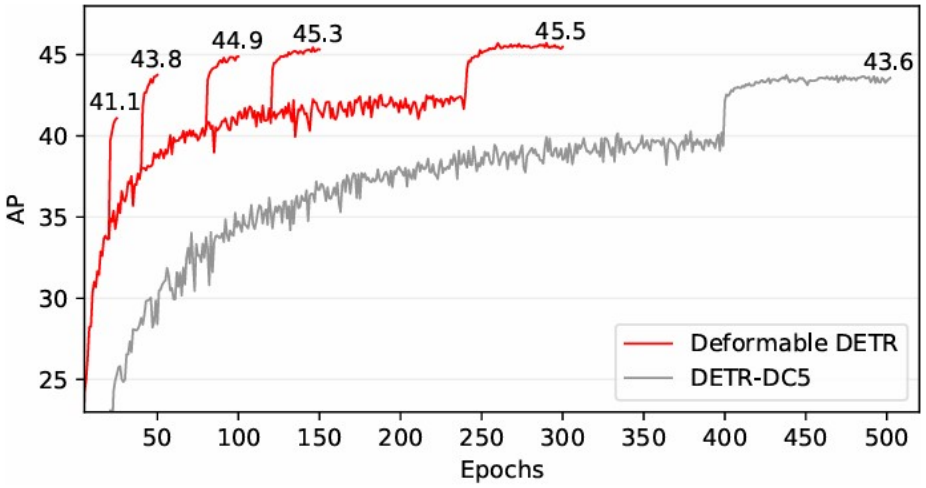




Deformable-DETR

Table 1: Comparision of Deformable DETR with DETR on COCO 2017 val set. DETR-DC5^{+} denotes DETR-DC5 with Focal Loss and 300 object queries.

Method	Epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	params	FLOPs	Training GPU hours	Inference FPS
Faster R-CNN + FPN	109	42.0	62.1	45.5	26.6	45.4	53.4	42M	180G	380	26
DETR	500	42.0	62.4	44.2	20.5	45.8	61.1	41M	86G	2000	28
DETR-DC5	500	43.3	63.1	45.9	22.5	47.3	61.1	41M	187G	7000	12
DETR-DC5	50	35.3	55.7	36.8	15.2	37.5	53.6	41M	187G	700	12
DETR-DC5^{+}	50	36.2	57.0	37.4	16.3	39.2	53.9	41M	187G	700	12
Deformable DETR	50	43.8	62.6	47.7	26.4	47.1	58.0	40M	173G	325	19
+ iterative bounding box refinement	50	45.4	64.7	49.0	26.8	48.3	61.7	40M	173G	325	19
++ two-stage Deformable DETR	50	46.2	65.2	50.0	28.8	49.2	61.7	40M	173G	340	19



模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

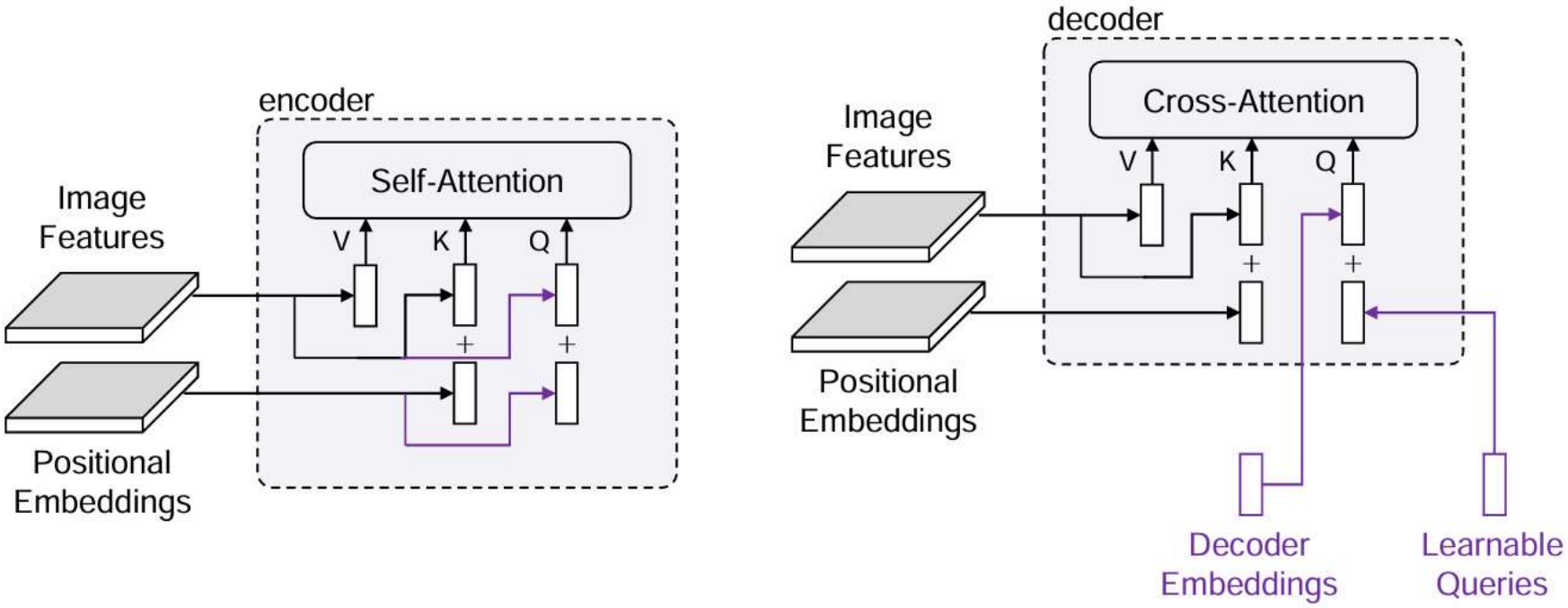
ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

DAB-DETR: Dynamic Anchor Boxes are Better Queries for DETR

Shilong Liu[1,2*], **Feng Li**[2,3], **Hao Zhang**[2,3], **Xiao Yang**[1],
Xianbiao Qi[2], **Hang Su**[1,4], **Jun Zhu**[1,4†], **Lei Zhang**[2†]
[1]Dept. of Comp. Sci. and Tech., BNRist Center, State Key Lab for Intell. Tech.
& Sys., Institute for AI, Tsinghua-Bosch Joint Center for ML, Tsinghua University.
[2]International Digital Economy Academy (IDEA).
[3]Hong Kong University of Science and Technology.
[4]Peng Cheng Laboratory, Shenzhen, Guangdong, China.

ICLR 2022

DAB-DETR

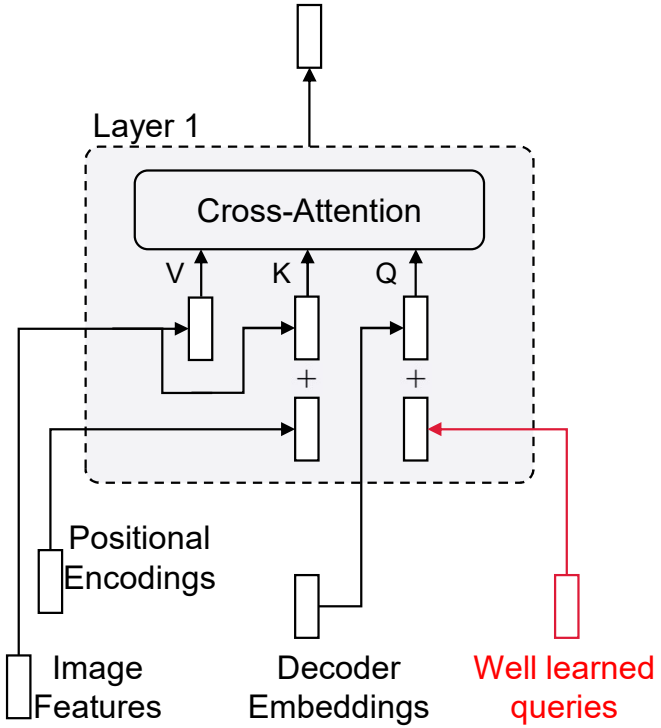


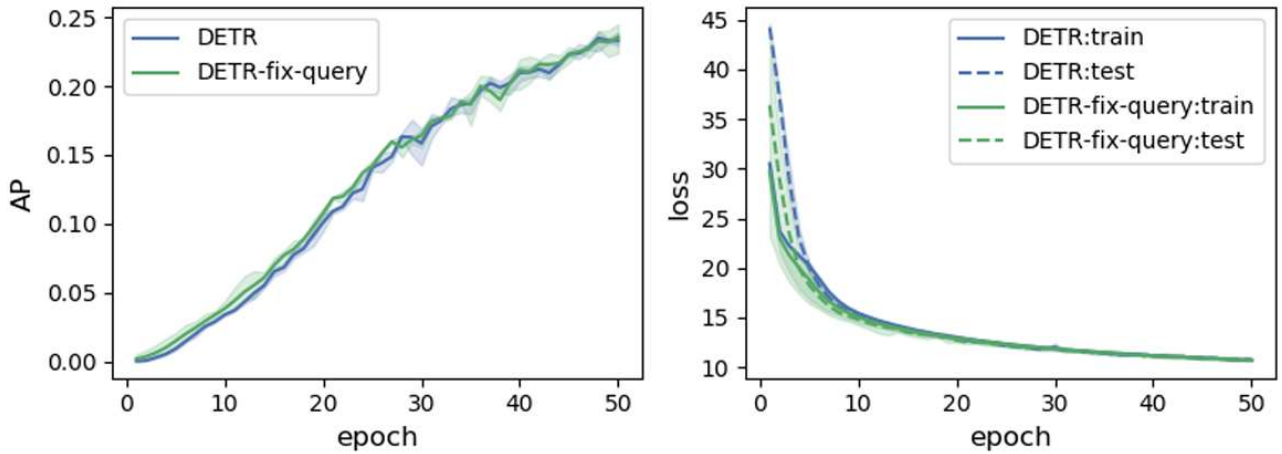
Each query in a decoder is composed of a decoder embedding (content information) and a learnable query (position information)

Reasons about slow training Convergence:

1. optimization challenge?
2. the positional information in the learned queries is not encoded in the same way as the sinusoidal positional encoding.

DAB-DETR

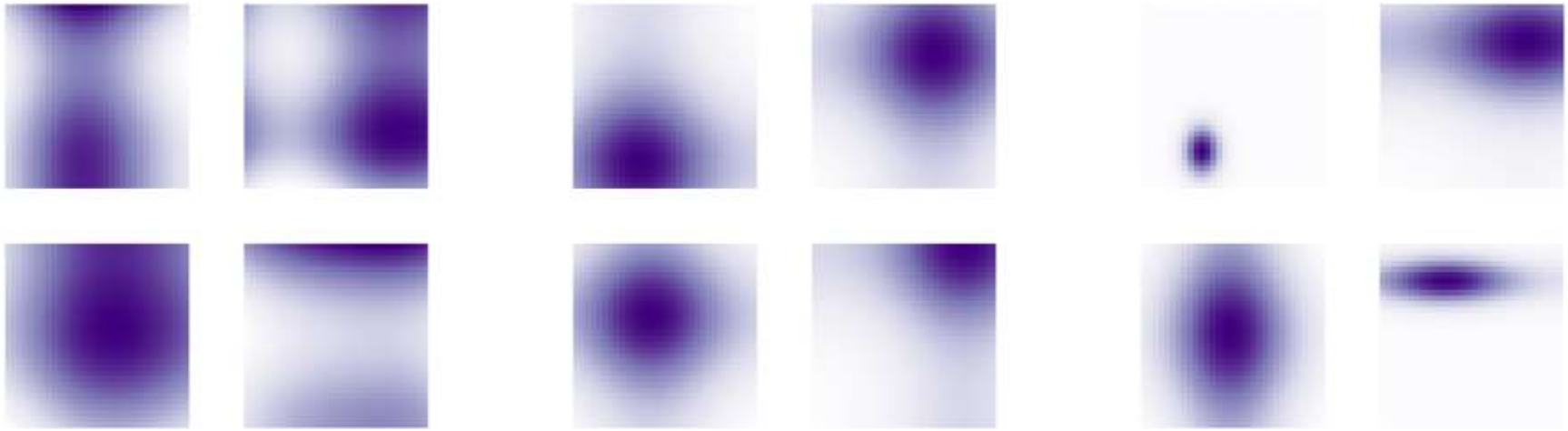




~~optimization challenge~~

Try to find out the undesirable properties about the learned queries!

DAB-DETR



(a) DETR (b) Conditional DETR (c) DAB-DETR

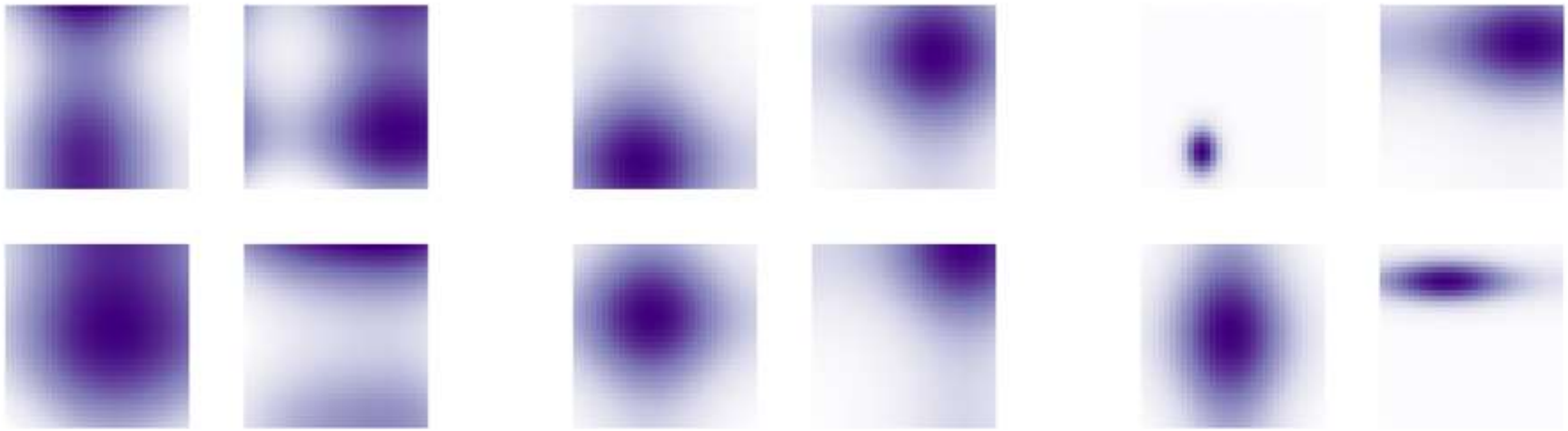
Positional attention maps between the learned queries and the positional embeddings from a feature map.

Each query can be regarded as a positional prior to let decoders focus on a region of interest.

DAB-DETR

Multi concentration centers which is hard to locate objects when multiple objects exist in an image.

explicit positional priors to constrain queries on local region



(a) DETR

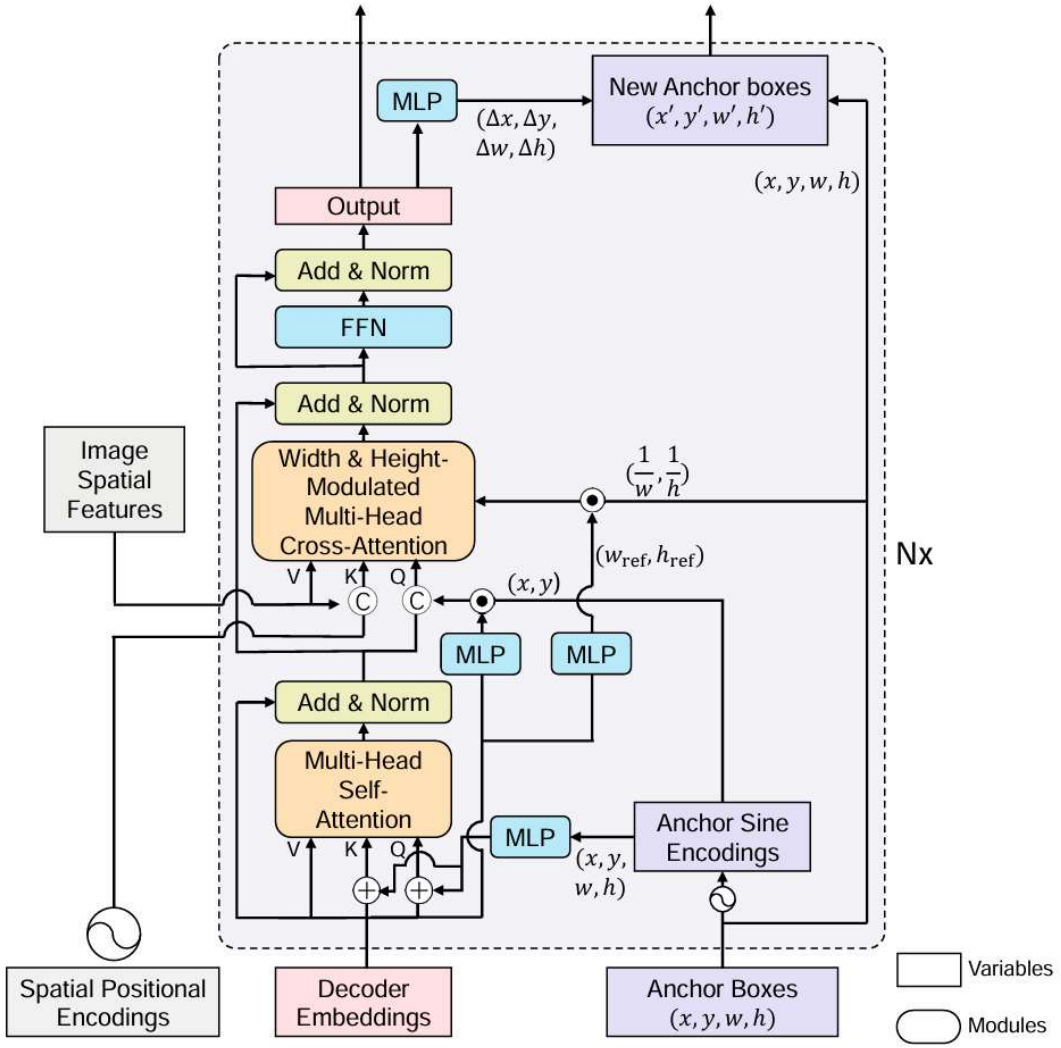
(b) Conditional DETR

(c) DAB-DETR

Focus on areas that are either too large or too small.

Position queries encoded in the same way as the image positional embeddings.

DAB-DETR



sinusoidal embeddings

$$P_q = \text{MLP}(\text{PE}(A_q))$$

$$\text{PE}(A_q) = \text{PE}(x_q, y_q, w_q, h_q) = \text{Cat}(\text{PE}(x_q), \text{PE}(y_q), \text{PE}(w_q), \text{PE}(h_q))$$

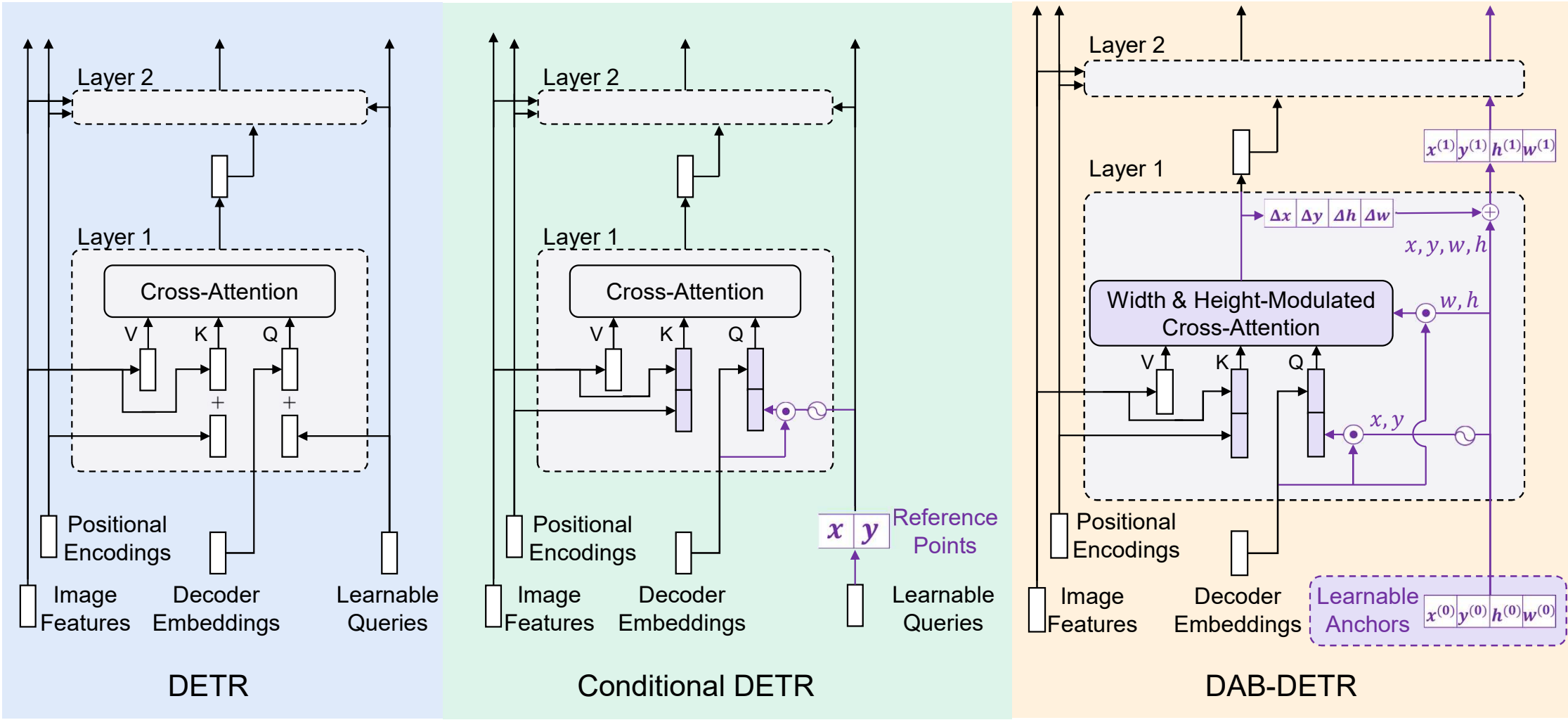
Self-Attn:

$$Q_q = C_q + P_q, \quad K_q = C_q + P_q, \quad V_q = C_q$$

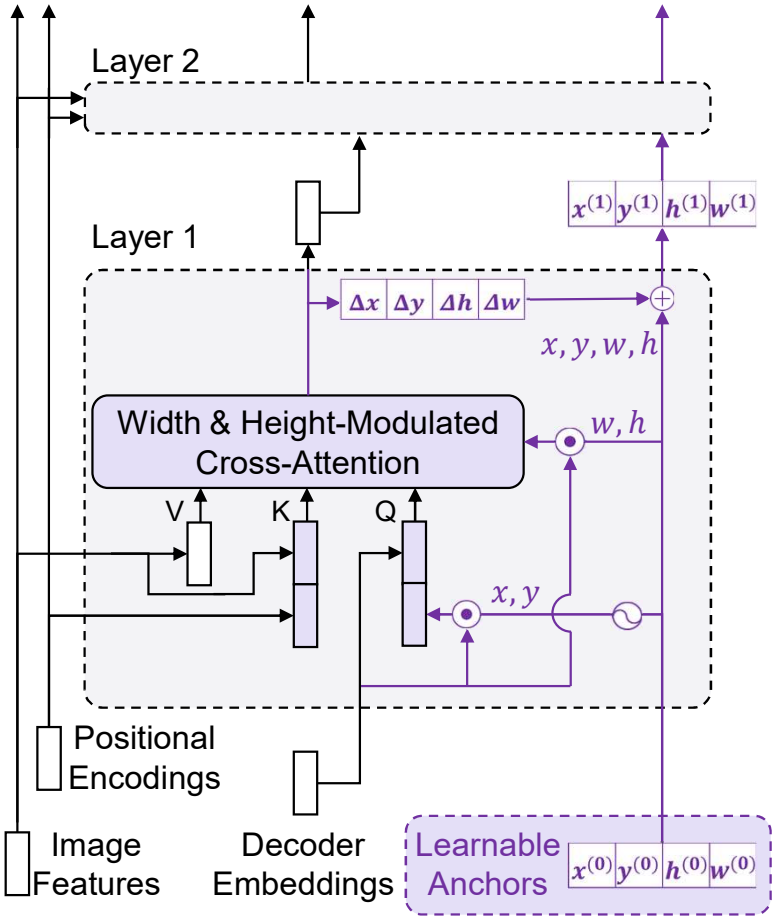
Cross-Attn:

$$Q_q = \text{Cat}(C_q, \text{PE}(x_q, y_q) \cdot \text{MLP}^{(\text{csq})}(C_q)),$$
$$K_{x,y} = \text{Cat}(F_{x,y}, \text{PE}(x, y)), \quad V_{x,y} = F_{x,y}$$

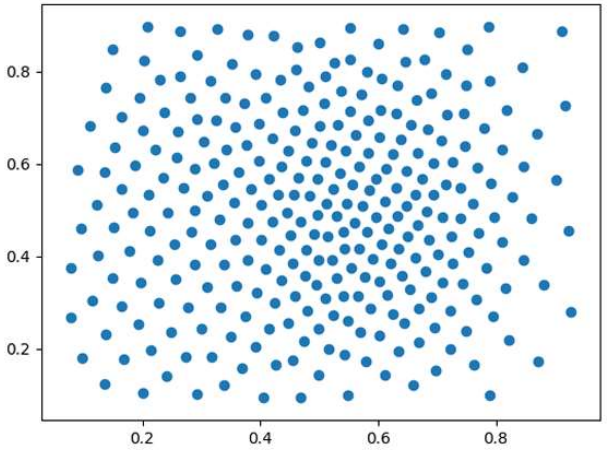
DAB-DETR


Layer 2
Layer 1
Cross-Attention
V
K
Q
Positional Encodings
Image Features
Decoder Embeddings
Learnable Queries
DETR
Layer 2
Layer 1
Cross-Attention
V
K
Q
Positional Encodings
Image Features
Decoder Embeddings
x y
Reference Points
Learnable Queries
Conditional DETR
Layer 2
Layer 1
x(1) y(1) h(1) w(1)
Δx Δy Δh Δw
x, y, w, h
Width & Height-Modulated Cross-Attention
w, h
V
K
Q
x, y
Positional Encodings
Image Features
Decoder Embeddings
Learnable Anchors
x(0) y(0) h(0) w(0)
DAB-DETR

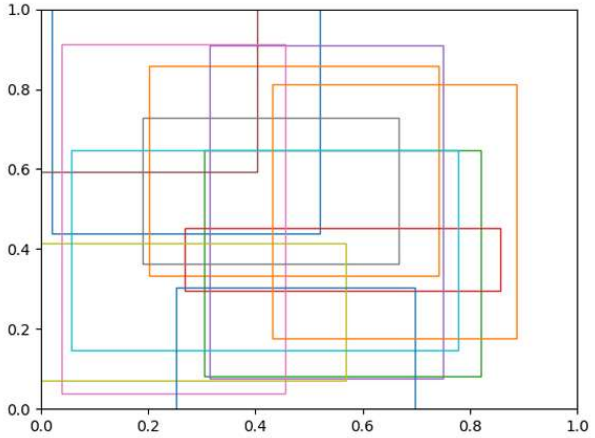
DAB-DETR



DAB-DETR



learn 2D coordinates only

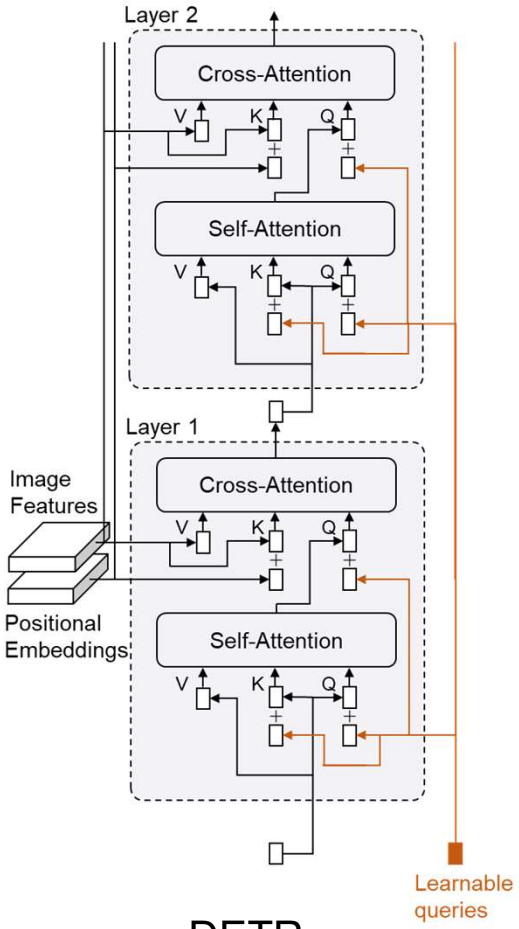


learn anchor boxes (partial)

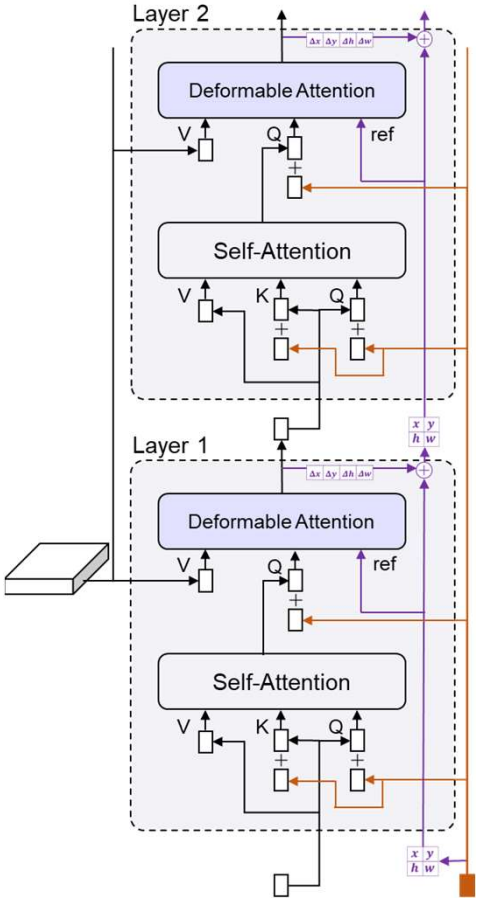
DAB-DETR

Model	MultiScale	#epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	GFLOPs	Params
DETR-R50		500	42.0	62.4	44.2	20.5	45.8	61.1	86	41M
Faster RCNN-FPN-R50		108	42.0	62.1	45.5	26.6	45.5	53.4	180	42M
Anchor DETR-R50*		50	42.1	63.1	44.9	22.3	46.2	60.0	–	39M
Conditional DETR-R50		50	40.9	61.8	43.3	20.8	44.6	59.2	90	44M
DAB-DETR-R50		50	42.2	63.1	44.7	21.5	45.7	60.3	94	44M
DAB-DETR-R50*		50	**42.6**	63.2	45.6	21.8	46.2	61.1	100	44M
DETR-DC5-R50		500	43.3	63.1	45.9	22.5	47.3	61.1	187	41M
Deformable DETR-R50	✓	50	43.8	62.6	47.7	26.4	47.1	58.0	173	40M
SMCA-R50	✓	50	43.7	63.6	47.2	24.2	47.0	60.4	152	40M
TSP-RCNN-R50	✓	96	45.0	64.5	49.6	29.7	47.7	58.0	188	–
Anchor DETR-DC5-R50*		50	44.2	64.7	47.5	24.7	48.2	60.6	151	39M
Conditional DETR-DC5-R50		50	43.8	64.4	46.7	24.0	47.6	60.7	195	44M
DAB-DETR-DC5-R50		50	44.5	65.1	47.7	25.3	48.2	62.3	202	44M
DAB-DETR-DC5-R50*		50	**45.7**	66.2	49.0	26.1	49.4	63.1	216	44M
DETR-R101		500	43.5	63.8	46.4	21.9	48.0	61.8	152	60M
Faster RCNN-FPN-R101		108	44.0	63.9	47.8	27.2	48.1	56.0	246	60M
Anchor DETR-R101*		50	43.5	64.3	46.6	23.2	47.7	61.4	–	58M
Conditional DETR-R101		50	42.8	63.7	46.0	21.7	46.6	60.9	156	63M
DAB-DETR-R101		50	43.5	63.9	46.6	23.6	47.3	61.5	174	63M
DAB-DETR-R101*		50	**44.1**	64.7	47.2	24.1	48.2	62.9	179	63M
DETR-DC5-R101		500	44.9	64.7	47.7	23.7	49.5	62.3	253	60M
TSP-RCNN-R101	✓	96	46.5	66.0	51.2	29.9	49.7	59.2	254	–
SMCA-R101	✓	50	44.4	65.2	48.0	24.3	48.5	61.0	218	50M
Anchor DETR-R101*		50	45.1	65.7	48.8	25.8	49.4	61.6	–	58M
Conditional DETR-DC5-R101		50	45.0	65.5	48.4	26.1	48.9	62.8	262	63M
DAB-DETR-DC5-R101		50	45.8	65.9	49.3	27.0	49.8	63.8	282	63M
DAB-DETR-DC5-R101*		50	**46.6**	67.0	50.2	28.1	50.5	64.1	296	63M

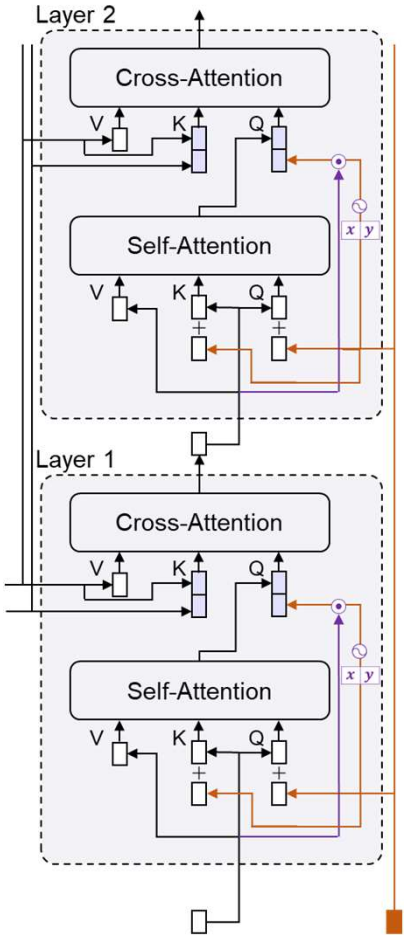
Understanding the queries



DETR

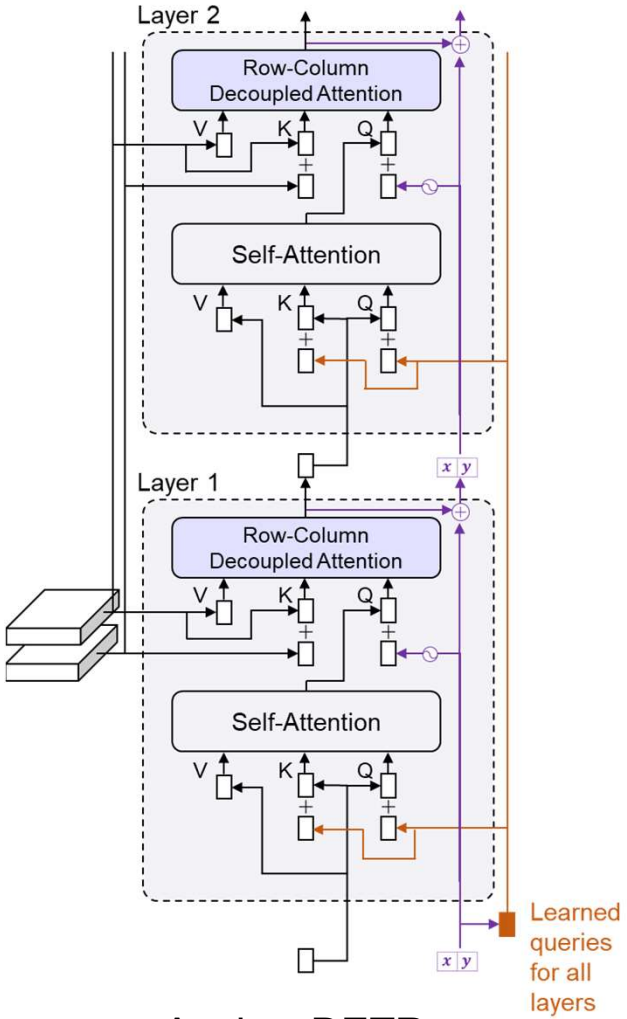


Deformable DETR

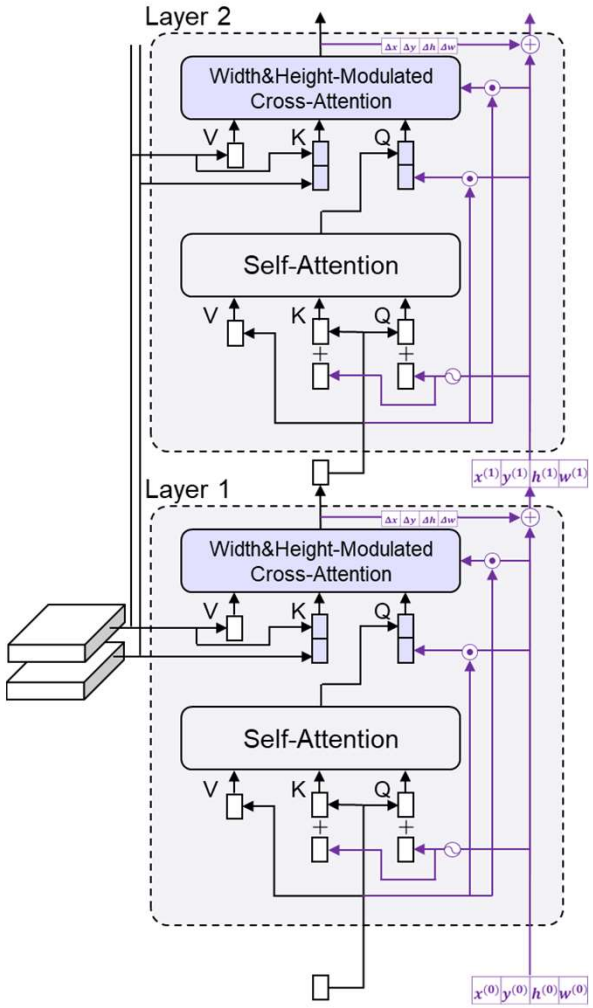


Conditional DETR

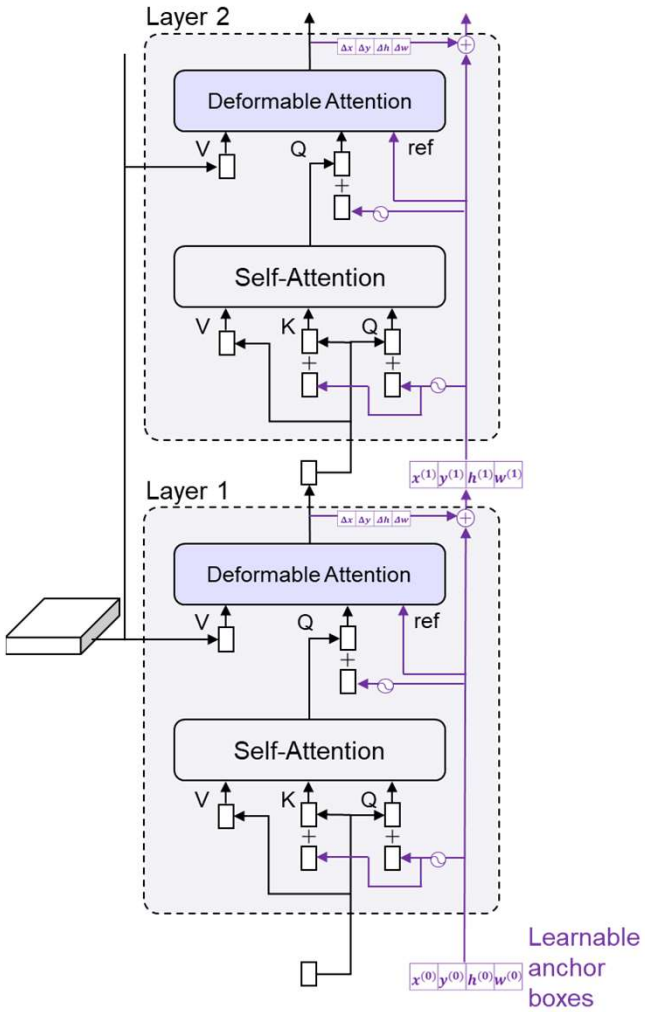
Understanding the queries



Anchor-DETR



DAB-DETR



DAB-Deformable-DETR

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

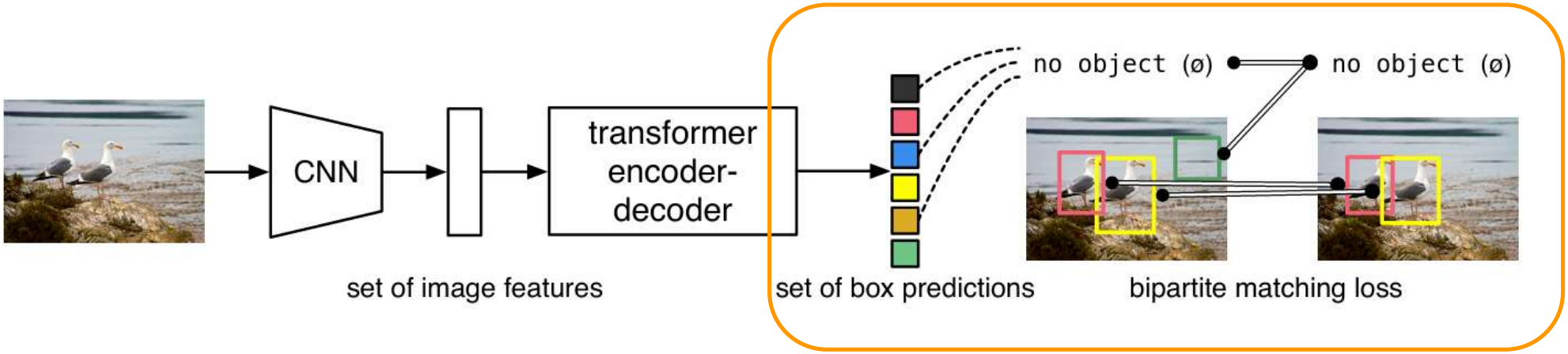
ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

DN-DETR: Accelerate DETR Training by Introducing Query Denoising

Feng Li*, Hao Zhang*, Shilong Liu, Jian Guo, Lionel M. Ni, and Lei Zhang

CVPR 2022

DN-DETR



bipartite graph matching

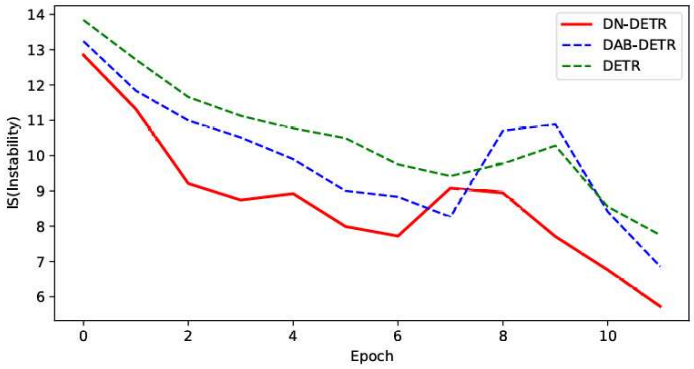
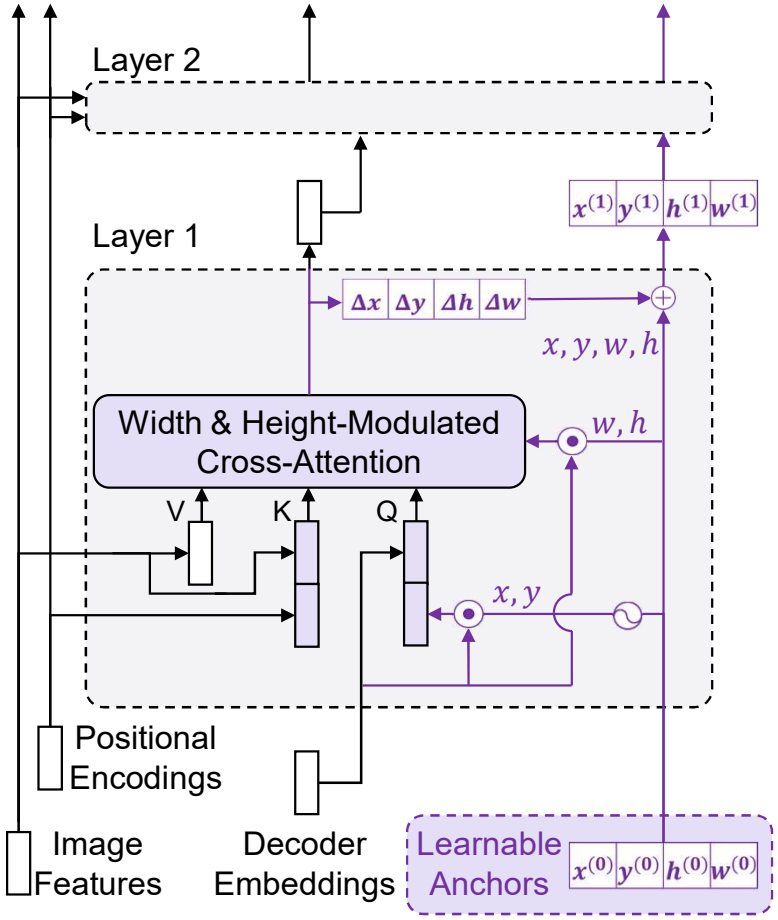


Fig. 2. The IS of DAB-DETR and DN-DETR during training. For each method, we train 12 epoch on the same setting. We test the change of the Hungarian matching between each two epochs on the Validation set as the IS.

$$V_n^i = \begin{cases} m, & \text{if } O_n^i \text{ matches } T_m \\ -1, & \text{if } O_n^i \text{ matches nothing} \end{cases}$$

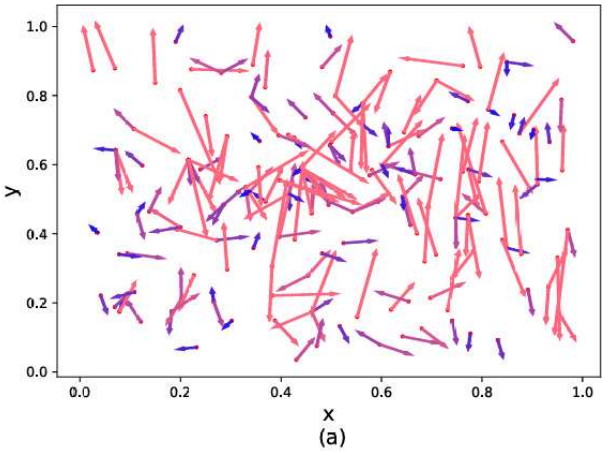
$$IS^i = \sum_{j=0}^{N} \mathbb{1}(V_n^i \neq V_n^{i-1})$$

DN-DETR

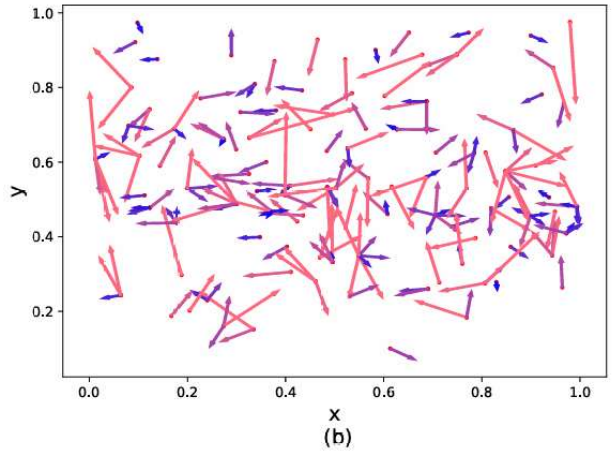


DAB-DETR

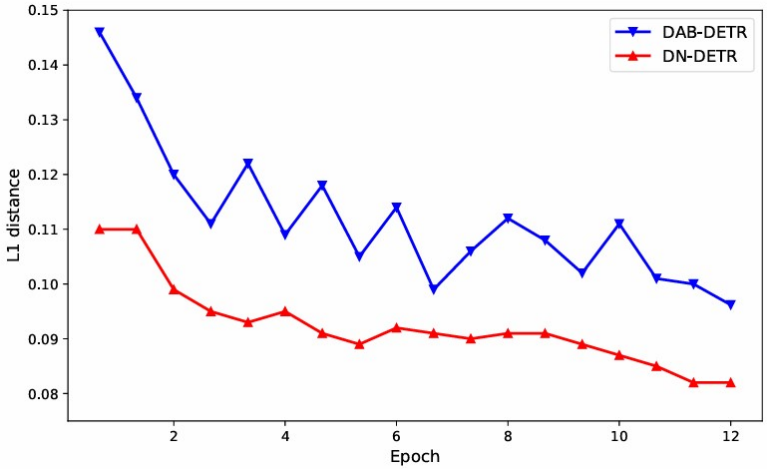
DETR



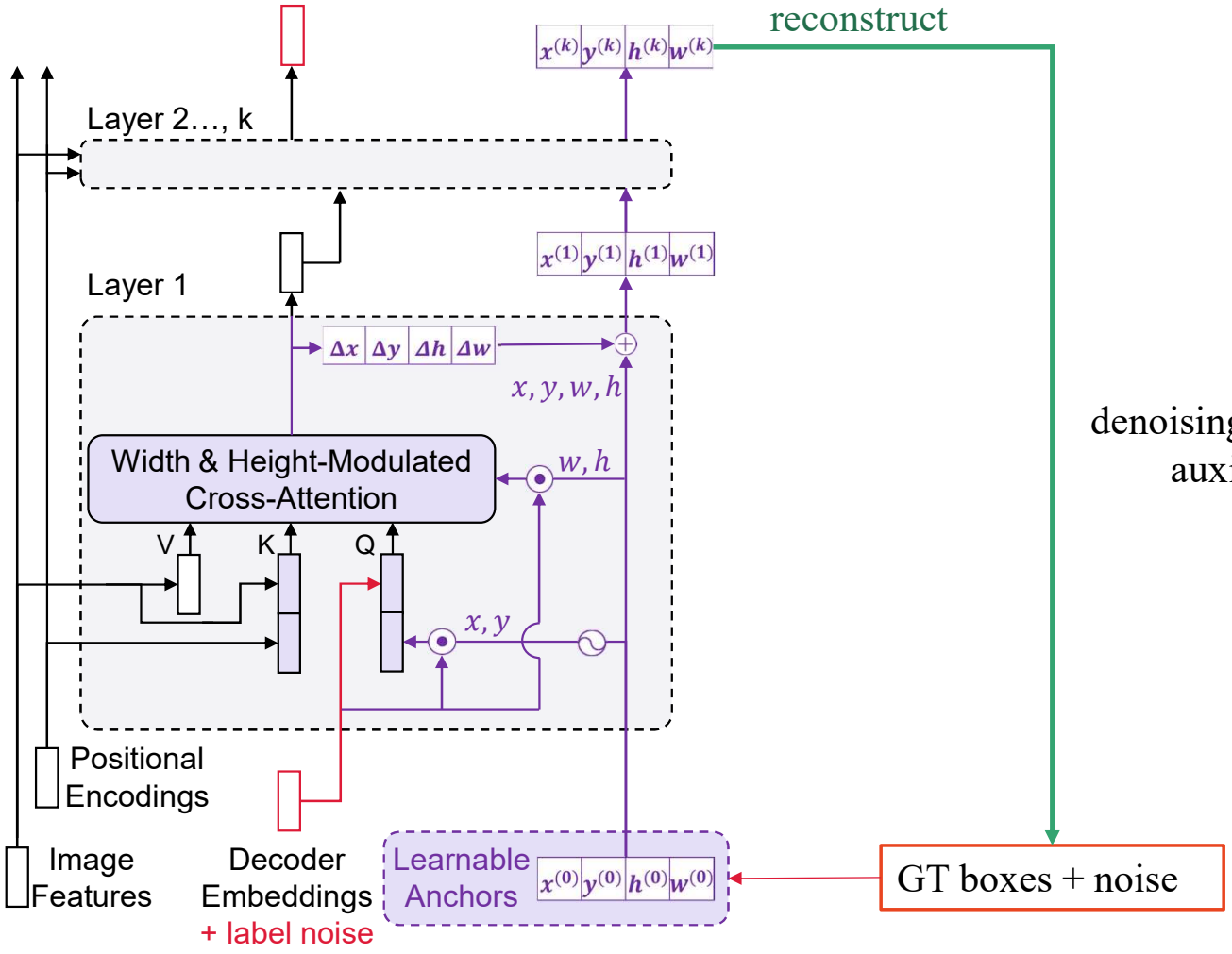
DN-DETR



Each arrow starts from an anchor and points to a target.

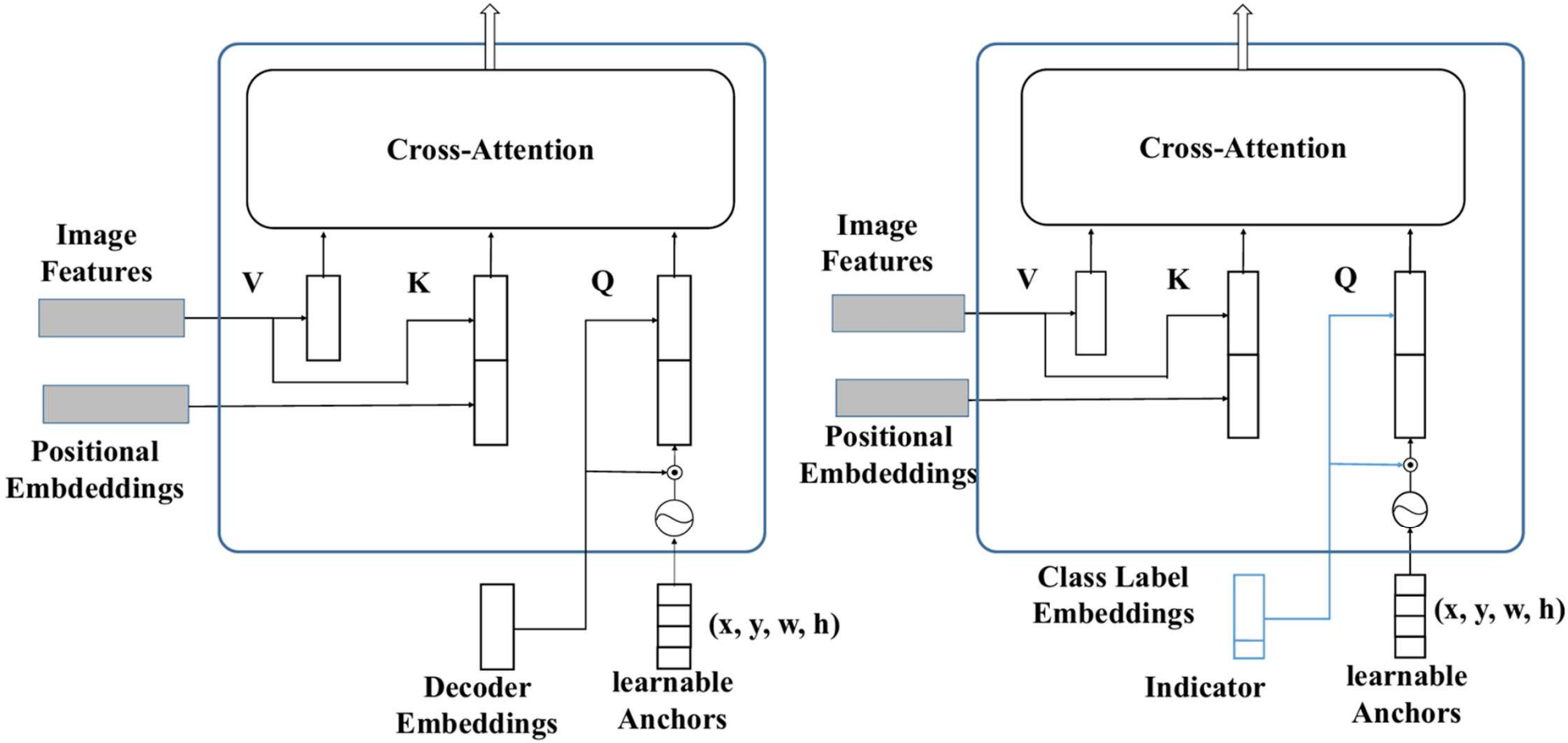


DN-DETR



Feed noised GT bounding boxes as noised queries together with learnable anchor queries into Transformer decoders.

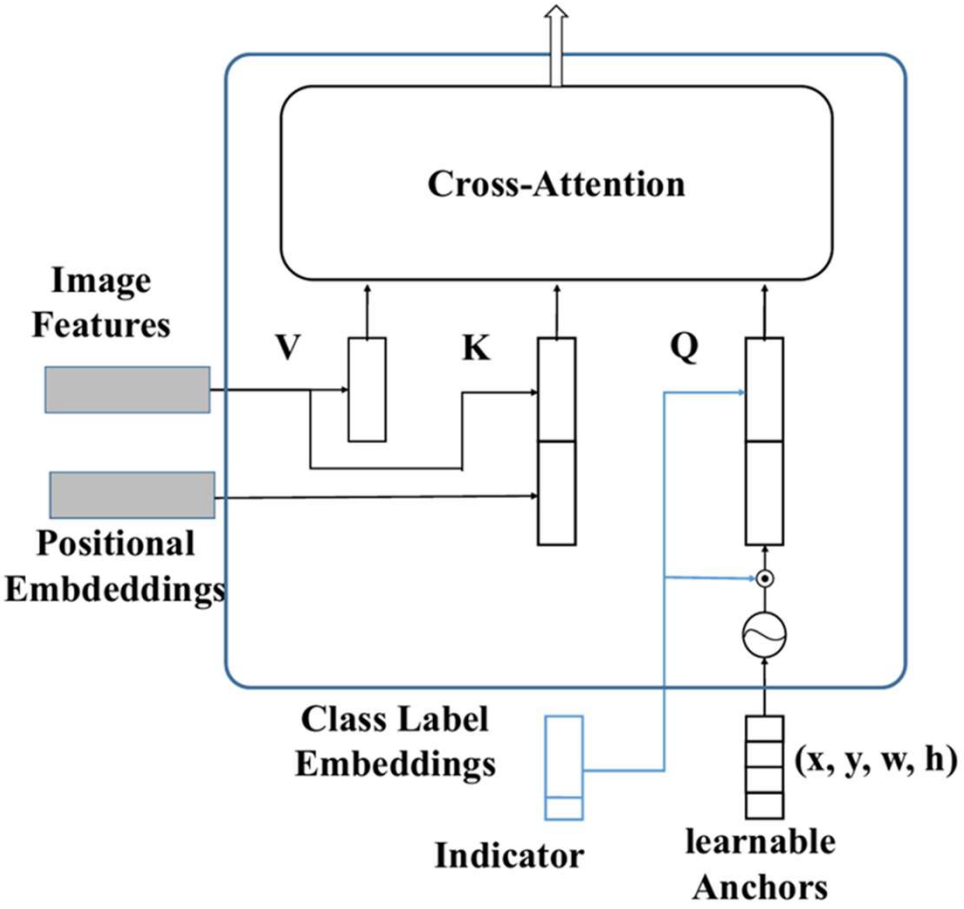
DN-DETR



Cross-attention in decoder of DAB-DETR

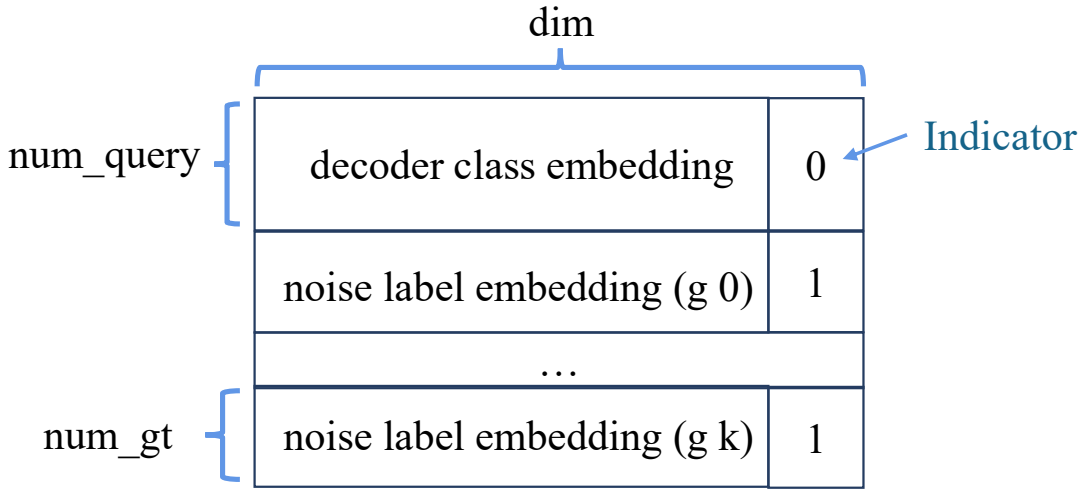
Cross-attention in decoder of DN-DETR

DN-DETR

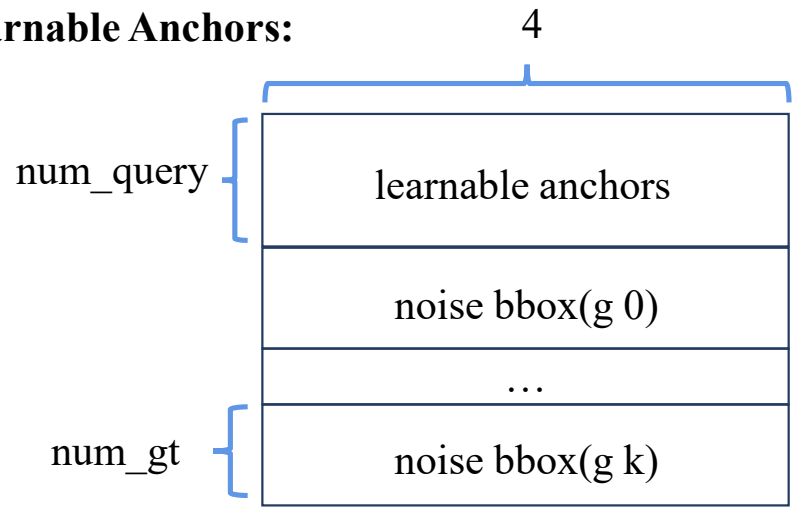


Cross-attention in decoder of DN-DETR

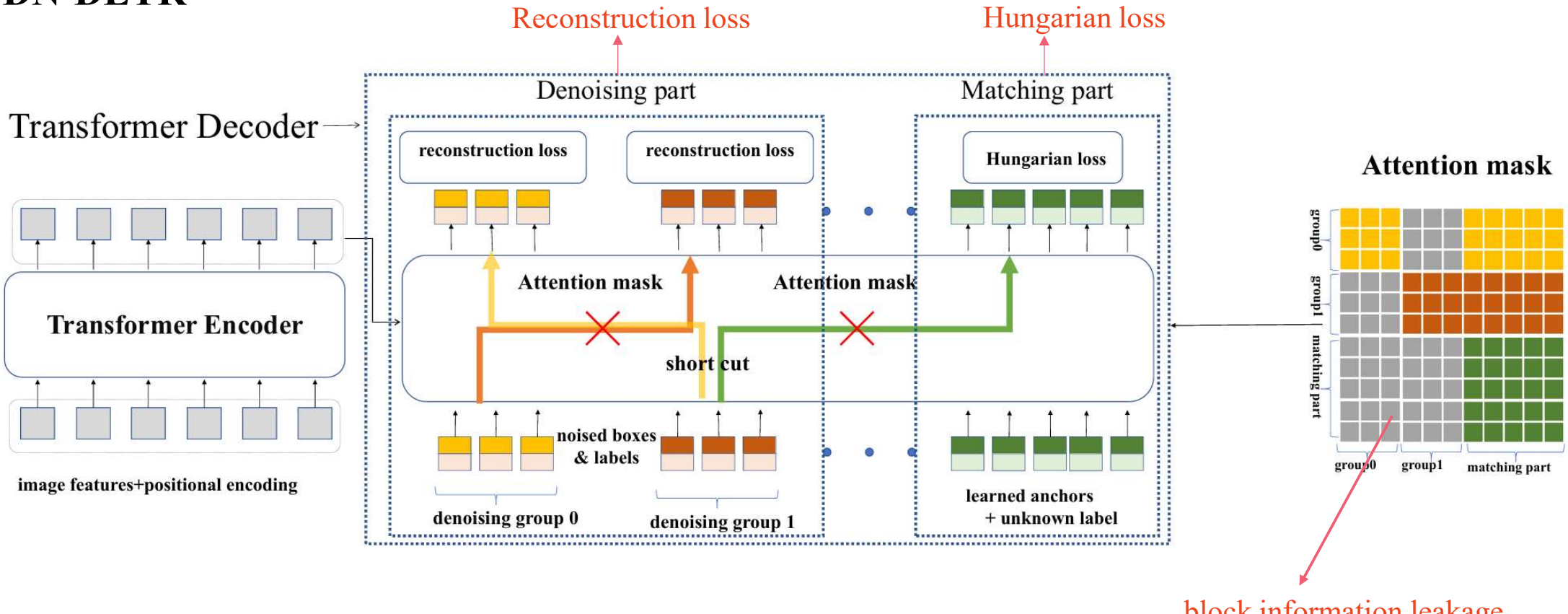
Class Label Embeddings:



Learnable Anchors:



DN-DETR



$$a_{ij} = \begin{cases} 1, & \text{if } j < P \times M \text{ and } \lfloor \frac{i}{M} \rfloor \neq \lfloor \frac{j}{M} \rfloor; \\ 1, & \text{if } j < P \times M \text{ and } i \geq P \times M; \\ 0, & \text{otherwise.} \end{cases}$$

DN-DETR

Model	#epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	GFLOPs	Params
DETR-R50 [1]	500	42.0	62.4	44.2	20.5	45.8	61.1	86	41M
Faster RCNN-FPN-R50 [18]	108	42.0	62.1	45.5	26.6	45.5	53.4	180	42M
Anchor DETR-R50 [21]	50	42.1	63.1	44.9	22.3	46.2	60.0	–	39M
Conditional DETR-R50 [15]	50	40.9	61.8	43.3	20.8	44.6	59.2	90	44M
DAB-DETR-R50 [14]	50	42.2	63.1	44.7	21.5	45.7	60.3	94	44M
DN-DETR-R50	50	**44.1(+1.9)**	64.4	46.7	22.9	48.0	63.4	94	44M
DETR-R101 [1]	500	43.5	63.8	46.4	21.9	48.0	61.8	152	60M
Faster RCNN-FPN-R101 [18]	108	44.0	63.9	47.8	27.2	48.1	56.0	246	60M
Anchor DETR-R101 [21]	50	43.5	64.3	46.6	23.2	47.7	61.4	–	58M
Conditional DETR-R101 [15]	50	42.8	63.7	46.0	21.7	46.6	60.9	156	63M
DAB-DETR-R101 [14]	50	43.5	63.9	46.6	23.6	47.3	61.5	174	63M
DN-DETR-R101	50	**45.2(+1.7)**	65.5	48.3	24.1	49.1	65.1	174	63M
DETR-DC5-R50 [1]	500	43.3	63.1	45.9	22.5	47.3	61.1	187	41M
Anchor DETR-DC5-R50 [21]	50	44.2	64.7	47.5	24.7	48.2	60.6	151	39M
Conditional DETR-DC5-R50 [15]	50	43.8	64.4	46.7	24.0	47.6	60.7	195	44M
DAB-DETR-DC5-R50 [14]	50	44.5	65.1	47.7	25.3	48.2	62.3	202	44M
DN-DETR-DC5-R50	50	**46.3(+1.8)**	66.4	49.7	26.7	50.0	64.3	202	44M
DETR-DC5-R101 [1]	500	44.9	64.7	47.7	23.7	49.5	62.3	253	60M
Anchor DETR-R101 [21]	50	45.1	65.7	48.8	25.8	49.4	61.6	–	58M
Conditional DETR-DC5-R101 [15]	50	45.0	65.5	48.4	26.1	48.9	62.8	262	63M
DAB-DETR-DC5-R101 [14]	50	45.8	65.9	49.3	27.0	49.8	63.8	282	63M
DN-DETR-DC5-R101	50	**47.3(+1.5)**	67.5	50.8	28.6	51.5	65.0	282	63M

模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

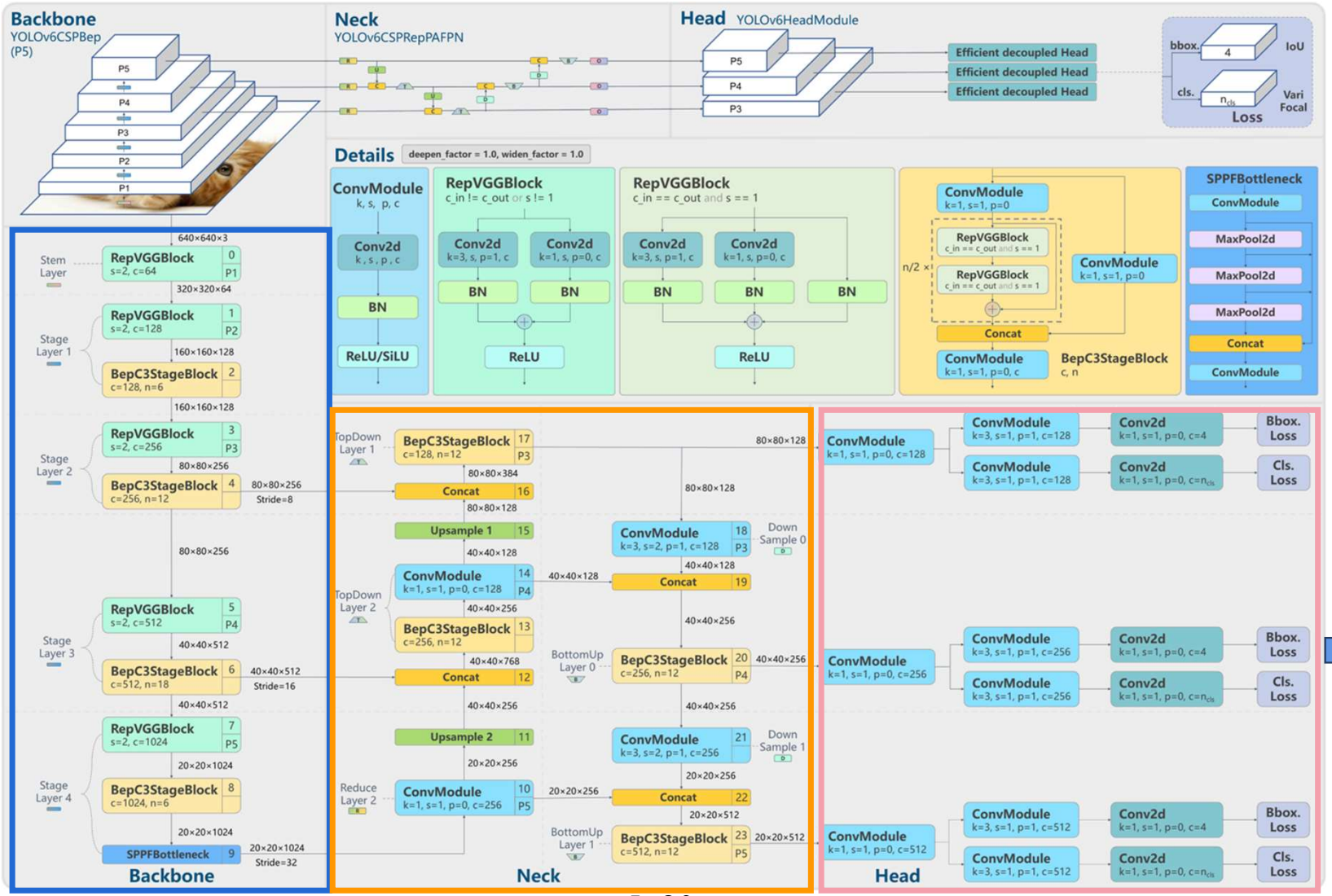
DETRs Beat YOLOs on Real-time Object Detection

Yian Zhao[1,2†] Wenyu Lv[1†‡] Shangliang Xu[1] Jinman Wei[1] Guanzhong Wang[1]
Qingqing Dang[1] Yi Liu[1] Jie Chen[2✉]

[1]Baidu Inc, Beijing, China [2]School of Electronic and Computer Engineering, Peking University, Shenzhen, China

CVPR 2024

Glance at YOLOs



post-process (NMS)

2~3 ms

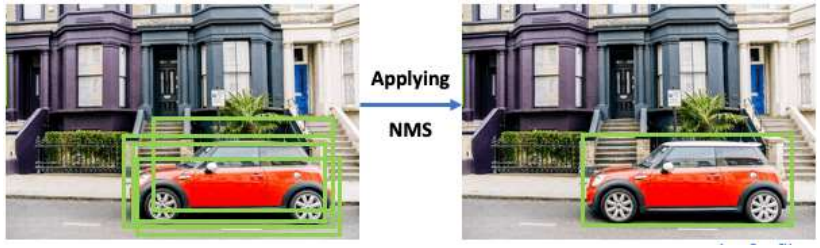
5~20 ms

github.com/open-mmlab/mmyolo

Glance at YOLOs

Model	Backbone	#Epochs	#Params (M)	GFLOPs	$FPS_{bs=1}$	AP^{val}	AP^{val}_{50}	AP^{val}_{75}	AP^{val}_{S}	AP^{val}_{M}	AP^{val}_{L}
Real-time Object Detectors											
YOLOv5-L [11]	-	300	46	109	54	49.0	67.3	-	-	-	-
YOLOv5-X [11]	-	300	86	205	43	50.7	68.9	-	-	-	-
PPYOLOE-L [40]	-	300	52	110	94	51.4	68.9	55.6	31.4	55.3	66.1
PPYOLOE-X [40]	-	300	98	206	60	52.3	69.9	56.5	33.3	56.3	66.4
YOLOv6-L [16]	-	300	59	150	99	52.8	70.3	57.7	34.4	58.1	70.1
YOLOv7-L [38]	-	300	36	104	55	51.2	69.7	55.5	35.2	55.9	66.7
YOLOv7-X [38]	-	300	71	189	45	52.9	71.1	57.4	36.9	57.7	68.6
YOLOv8-L [12]	-	-	43	165	71	52.9	69.8	57.5	35.3	58.3	69.8
YOLOv8-X [12]	-	-	68	257	50	53.9	71.0	58.7	35.7	59.3	70.7
End-to-end Object Detectors											
DINO-Deformable-DETR [44]	R50	36	47	279	5	50.9	69.0	55.3	34.6	54.1	64.6
Real-time End-to-end Object Detector (ours)											
RT-DETR	R50	72	42	136	**108**	**53.1**	**71.3**	**57.7**	34.8	58.0	70.0
RT-DETR	R101	72	76	259	**74**	**54.3**	**72.7**	58.6	36.0	58.8	**72.1**

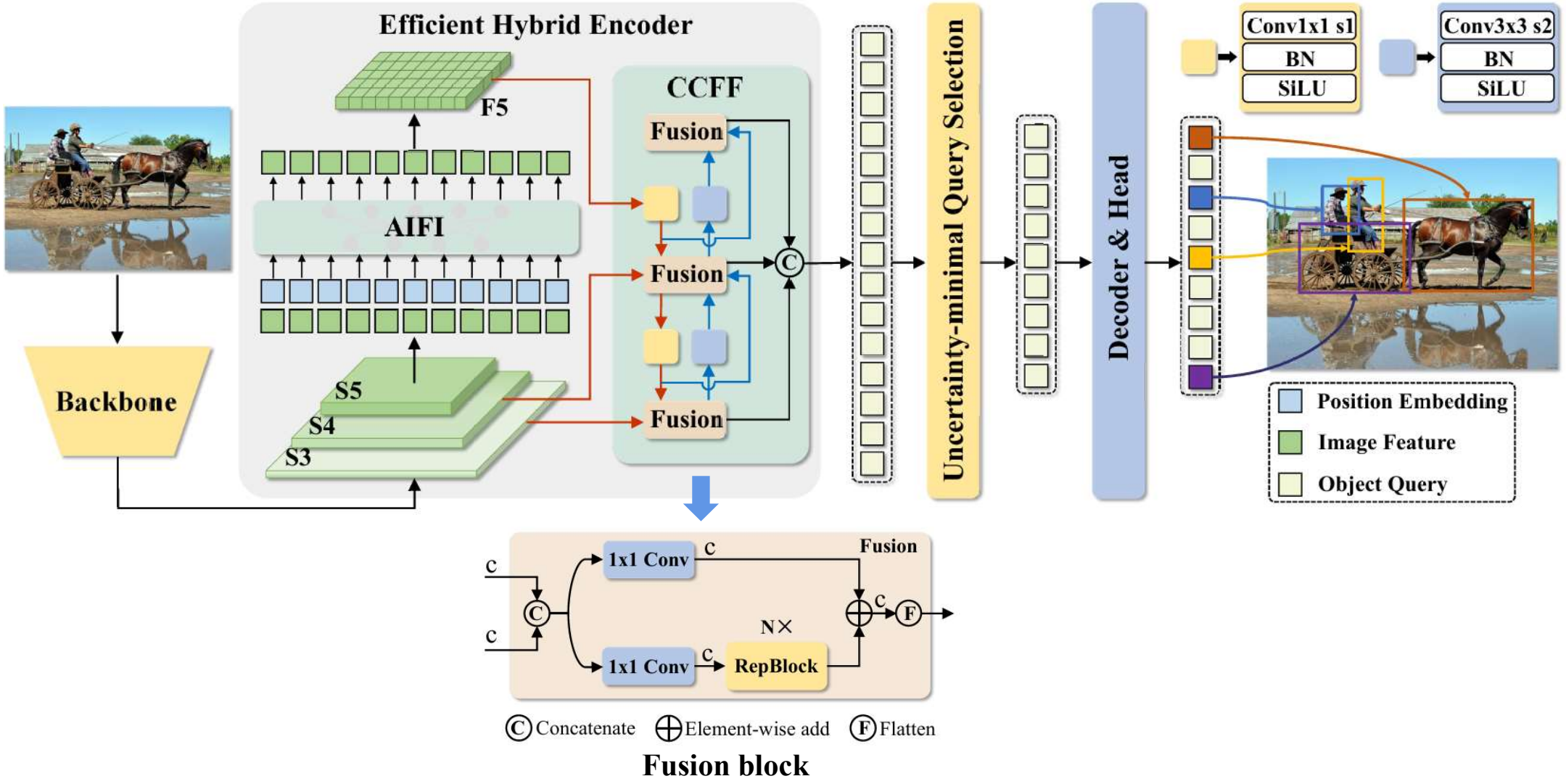
TensorRT FP16 and the input size is (800, 1333)



IoU thr. (Conf=0.001)	AP (%)	NMS (ms)	Conf thr. (IoU=0.7)	AP (%)	NMS (ms)
0.5	52.1	2.24	0.001	52.9	2.36
0.6	52.6	2.29	0.01	52.4	1.73
0.8	52.8	2.46	0.05	51.2	1.06

NMS execution time of YOLOv8

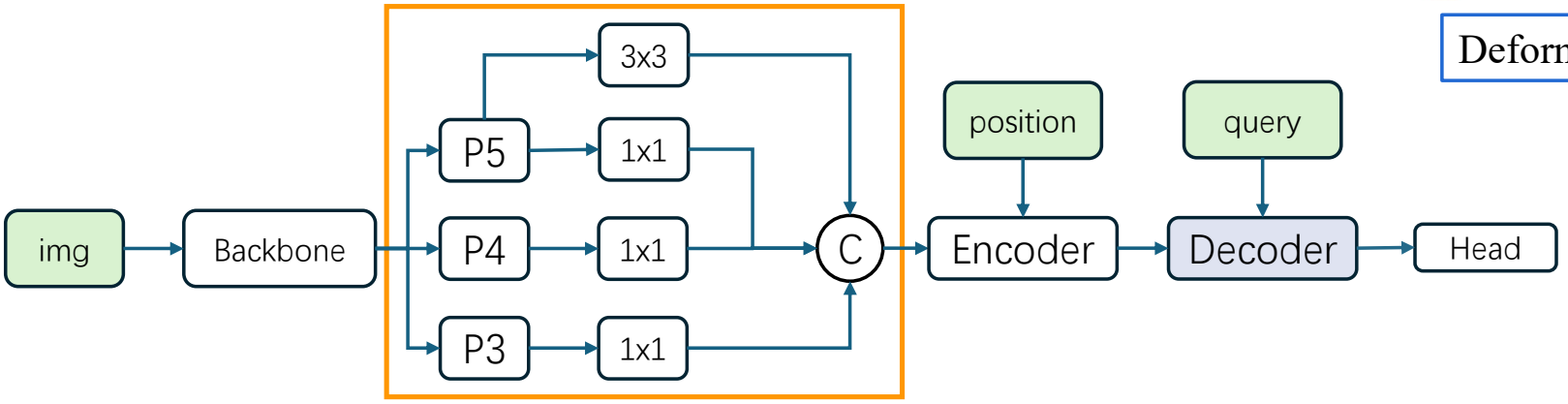
RT-DETR



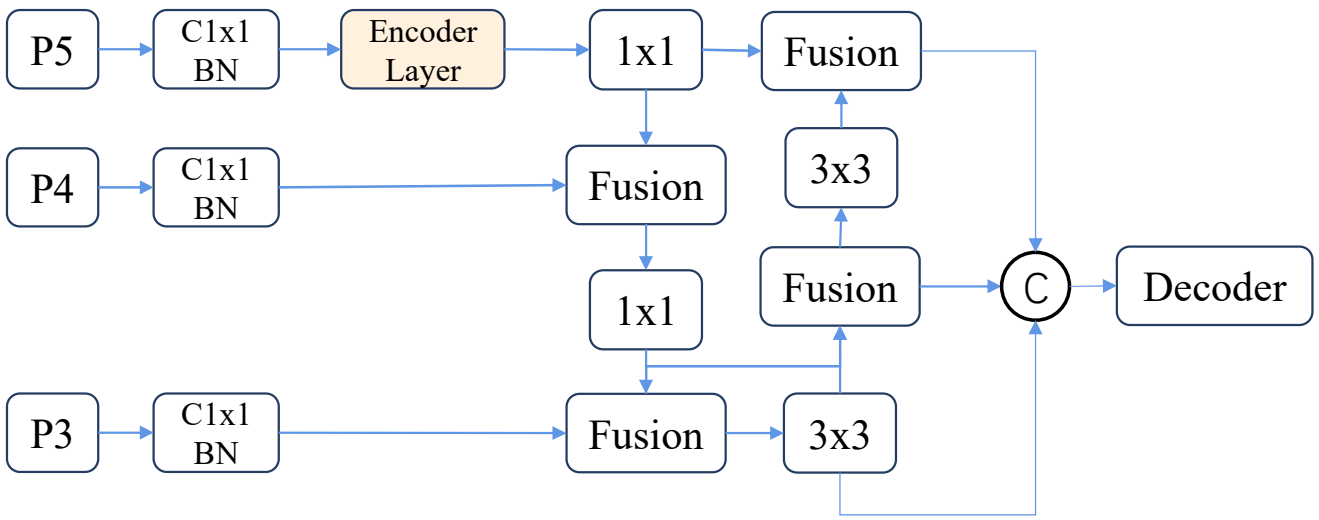
Fusion block

RT-DETR

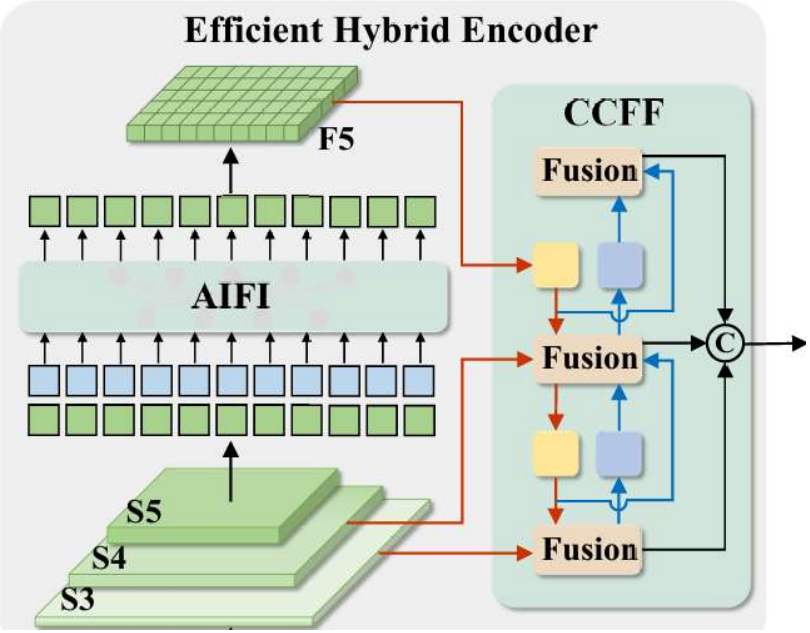
Deformable-DETR

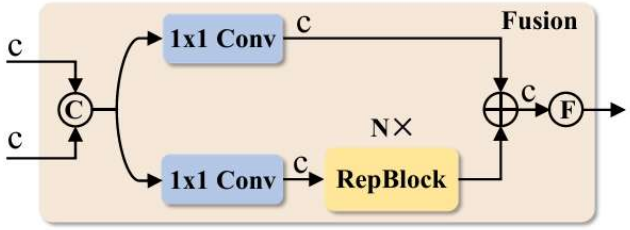


RT-DETR

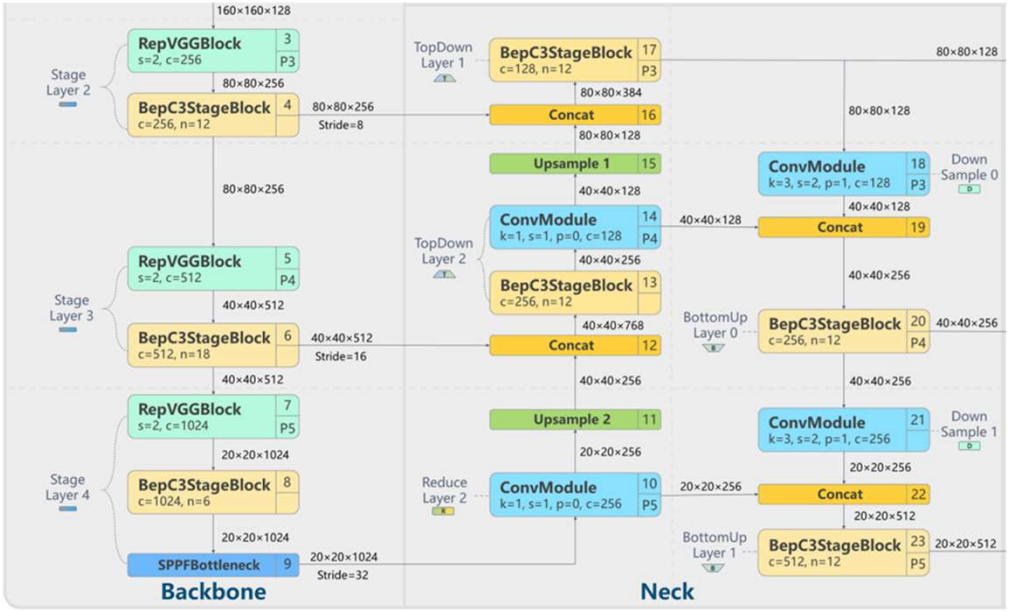


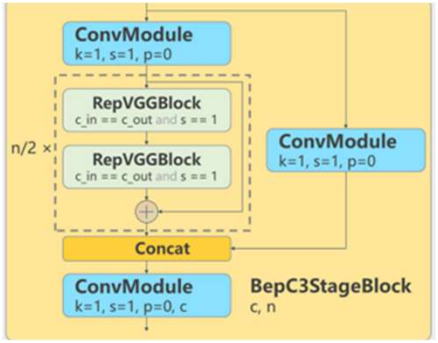
RT-DETR VS YOLOV6



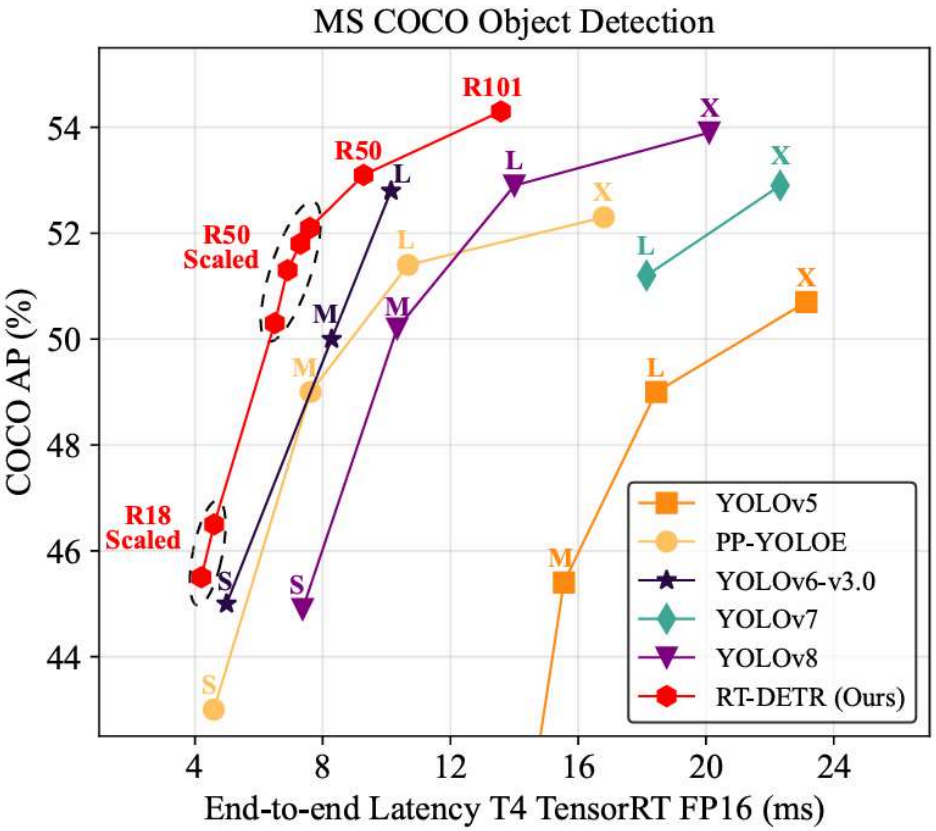


Fusion block





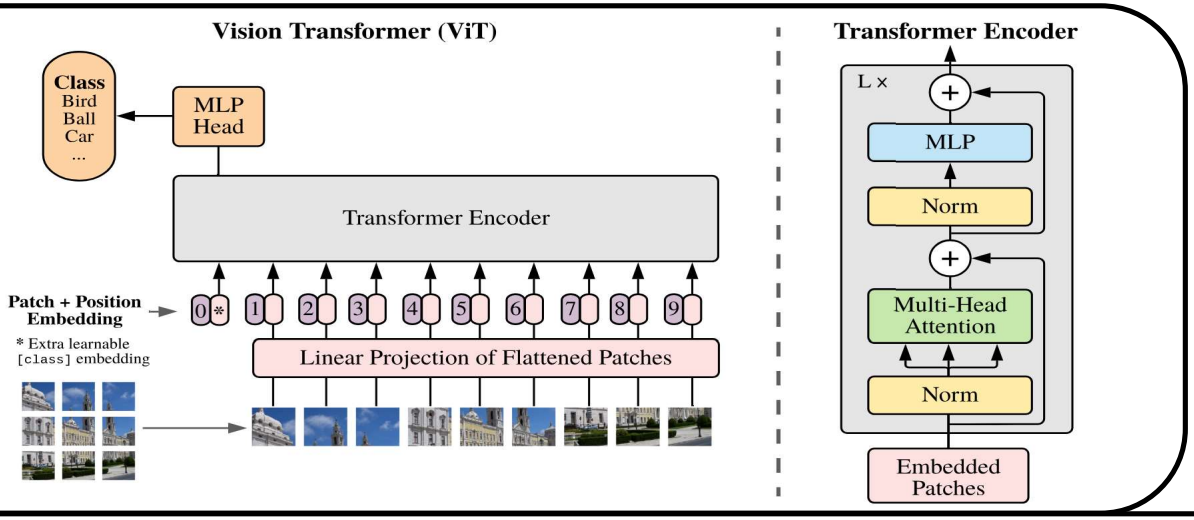
RT-DETR



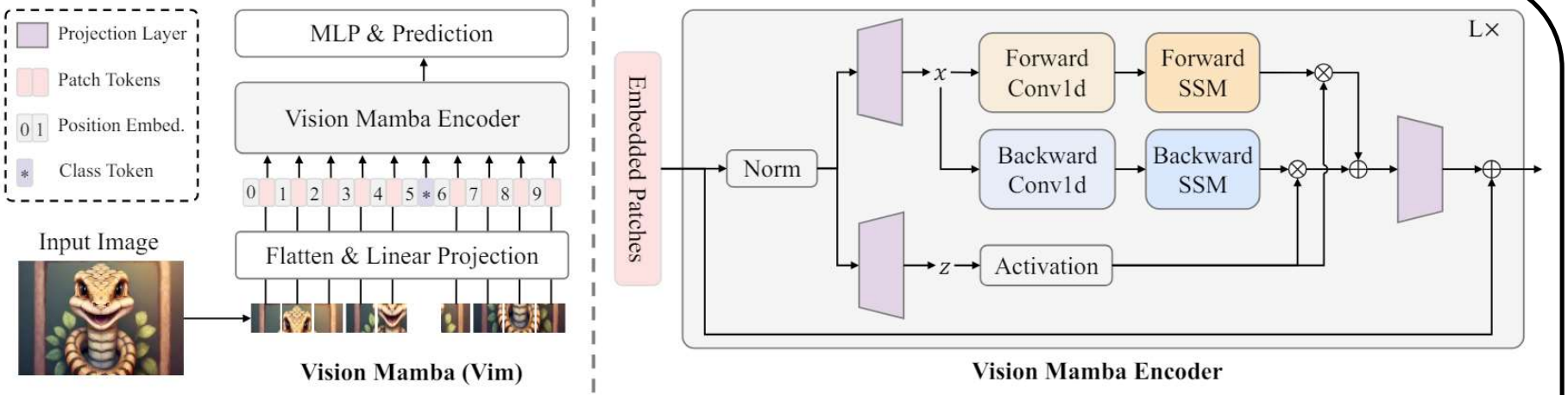
Model	#Epochs	#Params (M)	GFLOPs	$FPS_{bs=1}$	AP^{val}	AP^{val}_{50}	AP^{val}_{75}	AP^{val}_{S}	AP^{val}_{M}	AP^{val}_{L}
S and M models of YOLO Detectors										
YOLOv5-S[11]	300	7.2	16.5	74	37.4	56.8	-	-	-	-
YOLOv5-M[11]	300	21.2	49.0	64	45.4	64.1	-	-	-	-
PPYOLOE-S[40]	300	7.9	17.4	218	43.0	59.6	47.1	25.9	47.4	58.6
PPYOLOE-M[40]	300	23.4	49.9	131	48.9	65.8	53.7	30.8	53.4	65.3
YOLOv6-S[16]	300	18.5	45.3	201	45.0	61.8	48.9	24.3	50.2	62.7
YOLOv6-M[16]	300	34.9	85.8	121	50.0	66.9	54.6	30.6	55.4	67.3
YOLOv8-S[12]	-	11.2	28.6	136	44.9	61.8	48.6	25.7	49.9	61.0
YOLOv8-M[12]	-	25.9	78.9	97	50.2	67.2	54.6	32.0	55.7	66.4
Scaled RT-DETRs										
Scaled RT-DETR-R50-Dec^2	72	$36^{\dagger}$	98.4	**154**	**50.3**	**68.4**	54.5	**32.2**	55.2	**67.5**
Scaled RT-DETR-R50-Dec^3	72	$36^{\dagger}$	100.1	**145**	**51.3**	**69.6**	**55.4**	**33.6**	**56.1**	**68.6**
Scaled RT-DETR-R50-Dec^4	72	$36^{\dagger}$	101.8	**137**	**51.8**	**70.0**	**55.9**	**33.7**	**56.4**	**69.4**
Scaled RT-DETR-R50-Dec^5	72	$36^{\dagger}$	103.5	**132**	**52.1**	**70.5**	**56.2**	**34.3**	**56.9**	**69.9**
Scaled RT-DETR-R50-Dec^6	72	36	105.2	125	**52.2**	**70.6**	**56.4**	**34.4**	**57.0**	**70.0**
Scaled RT-DETR-R34-Dec^2	72	$31^{\dagger}$	89.3	185	**47.4**	**64.7**	**51.3**	**28.9**	**51.0**	**64.2**
Scaled RT-DETR-R34-Dec^3	72	$31^{\dagger}$	91.0	172	**48.5**	**66.2**	**52.3**	**30.2**	**51.9**	**66.2**
Scaled RT-DETR-R34-Dec^4	72	31	92.7	161	**48.9**	**66.8**	**52.9**	**30.6**	**52.4**	**66.3**
Scaled RT-DETR-R18-Dec^2	72	$20^{\dagger}$	59.0	**238**	**45.5**	**62.5**	**49.4**	**27.8**	48.7	61.7
Scaled RT-DETR-R18-Dec^3	72	20	60.7	217	**46.5**	**63.8**	**50.4**	**28.4**	49.8	**63.0**

Is the Mamba-based detector possible?

ViT

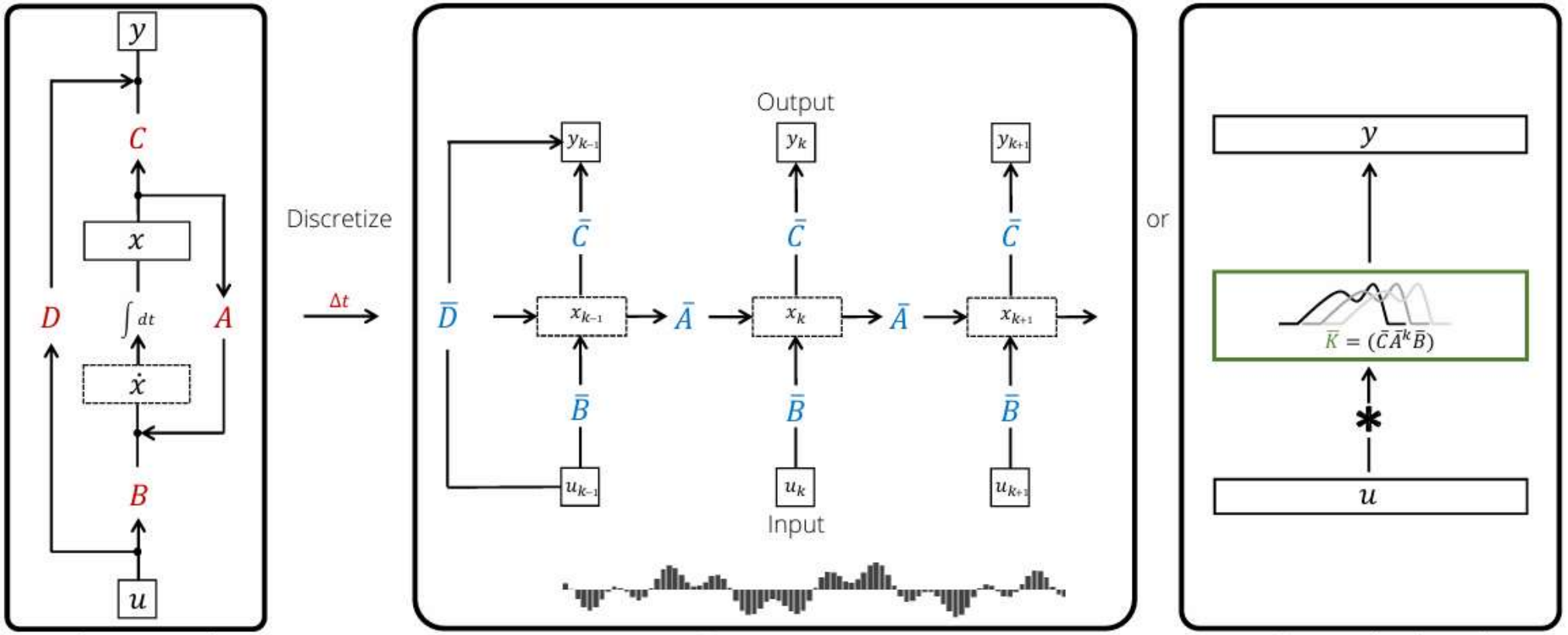


Vision Mamba



Zhu L, Liao B, Zhang Q, et al. Vision mamba: Efficient visual representation learning with bidirectional state space model[J]. arXiv preprint arXiv:2401.09417, 2024.

Mamba (Selective State Spaces)



Continuous-time
- ✓ *continuous data*
- ✓ *irregular sampling*

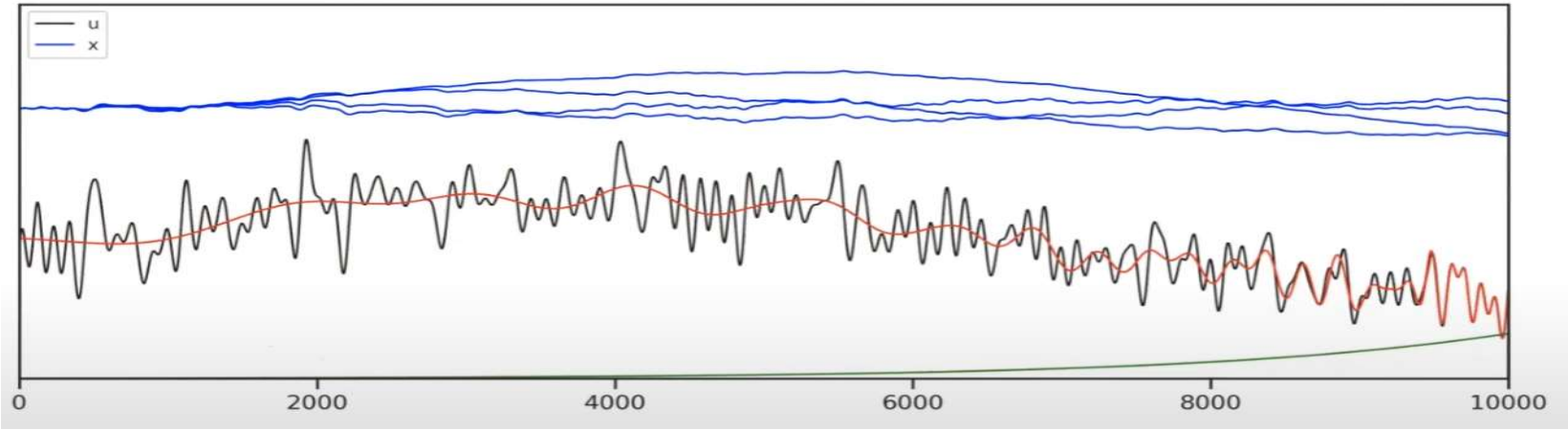
Recurrent
- ✓ *unbounded context*
- ✓ *efficient inference*

Convolutional
- ✓ *local information*
- ✓ *parallelizable training*

Mamba (HiPPO)

$$h_t = \overline{A}h_{t-1} + \overline{B}x_t$$
$$y_t = Ch_t$$
$$\overline{A} = \exp(\Delta A)$$
$$\overline{B} = (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B$$

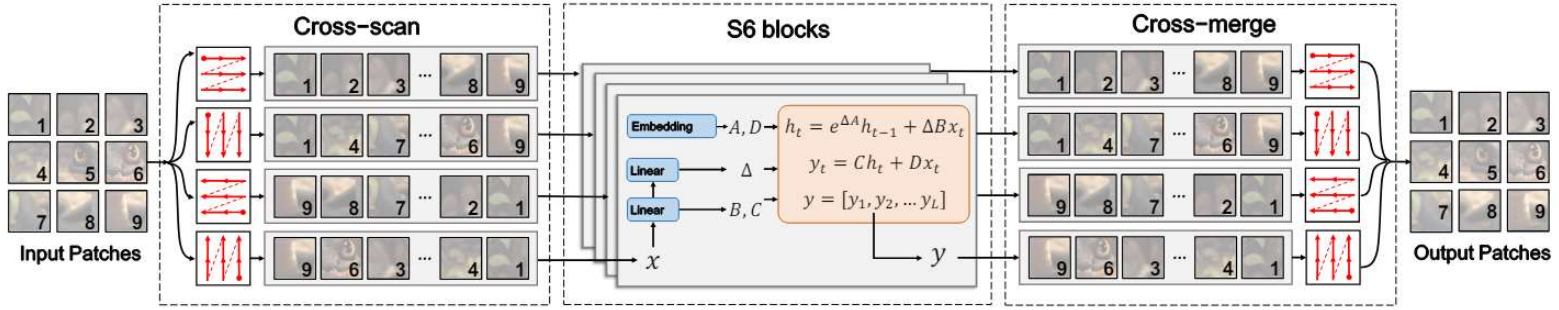
A matrix can build in such a way that it approximates all the input signal seen so far into a vector of coefficients by Legendre polynomials.

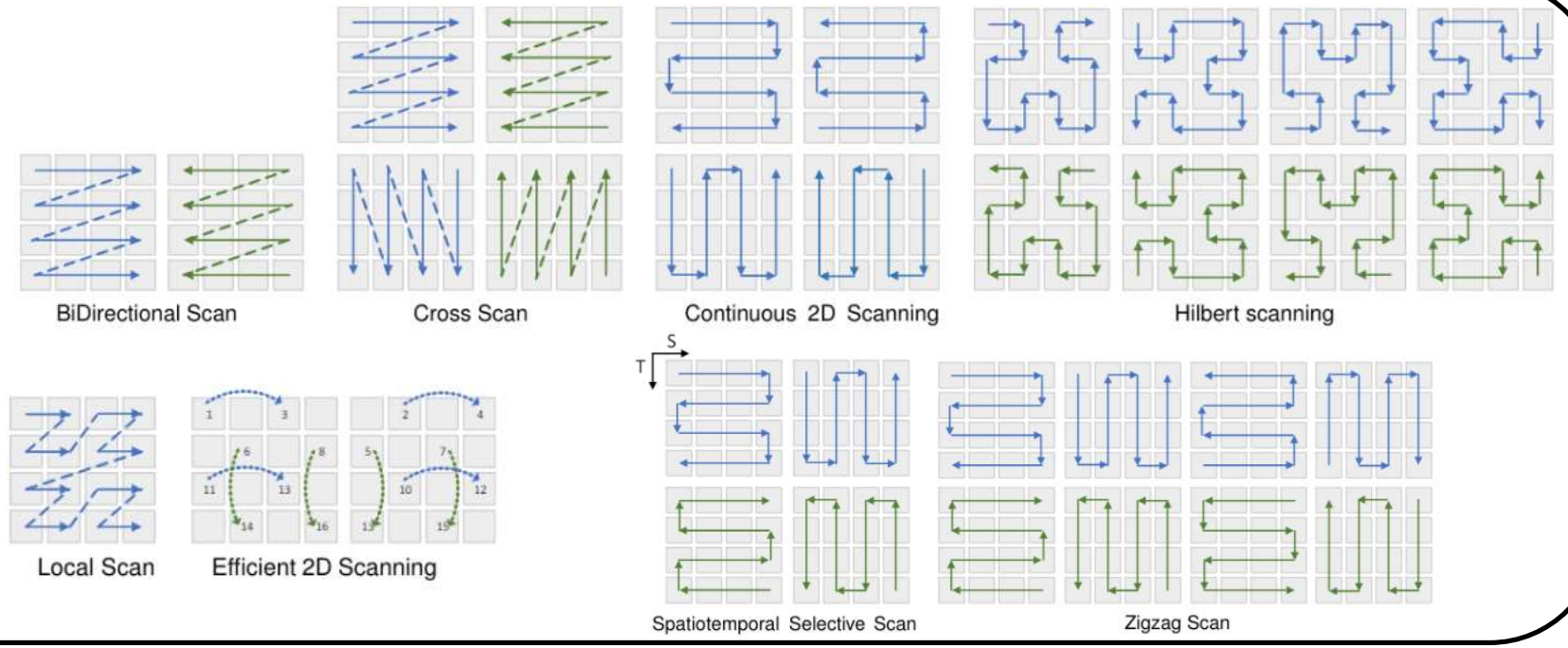


As opposed to traditional function approximation instead of constructing all signals perfectly, new signals are constructed exactly, and old signals decay exponentially, so being able to h(t) is able to capture recent tokens.

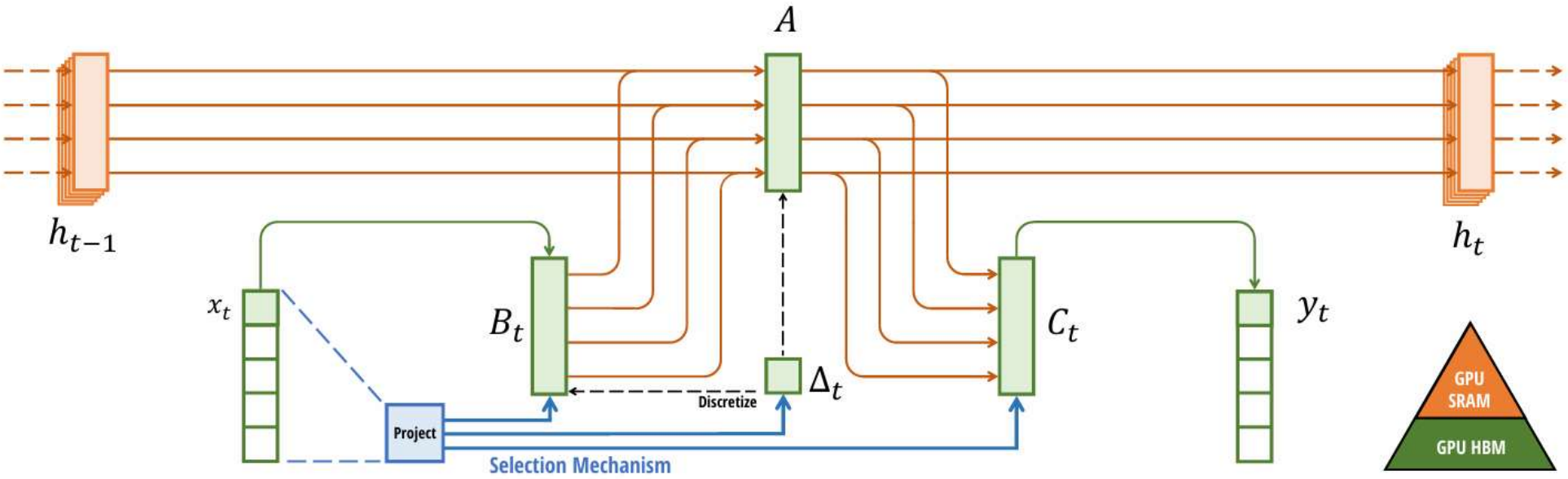
Mamba (Scan methods)

VMamba



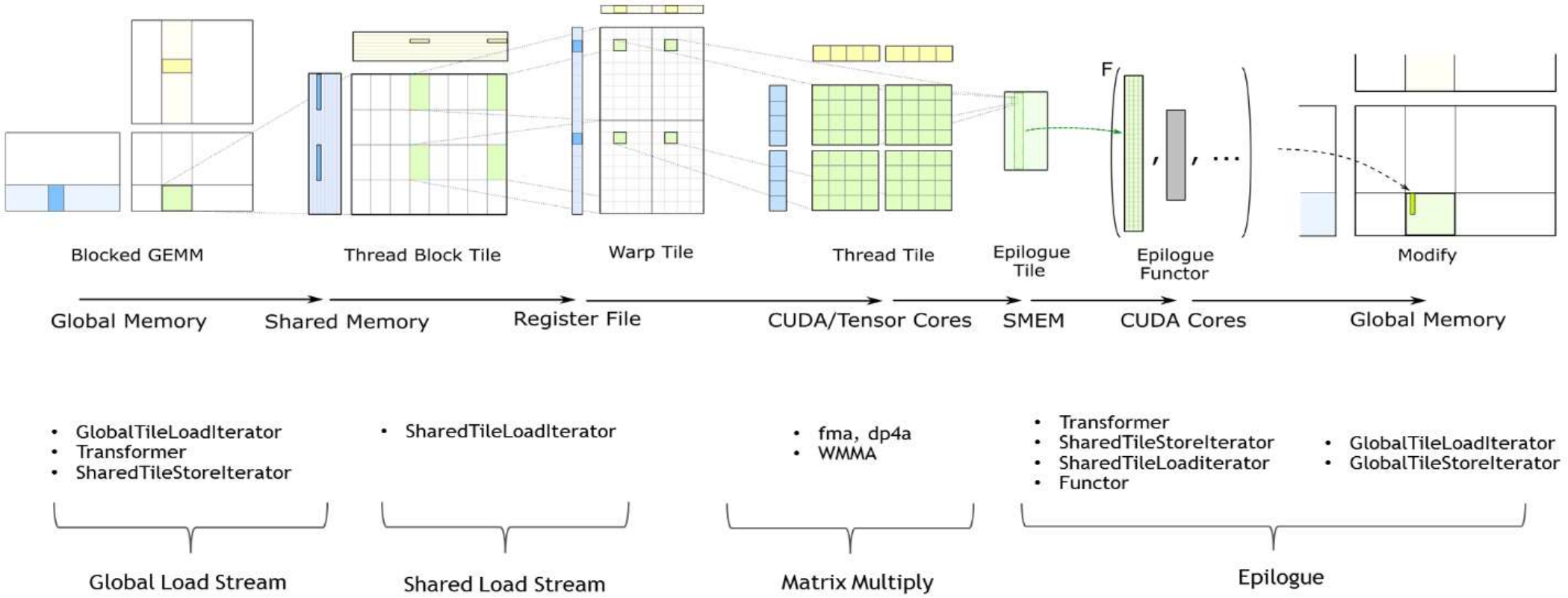


Mamba



1. We read in $O(BLD + DN)$ bytes of memory $(\Delta, \boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ from slow HBM to fast SRAM.
2. We discretize to produce $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}$ of size (B, L, D, N) in SRAM.
3. We perform a parallel associative scan, yielding intermediate states of size (B, L, D, N) in SRAM.
4. We multiply and sum with $\boldsymbol{C}$, producing outputs of size (B, L, D) and write it to HBM.

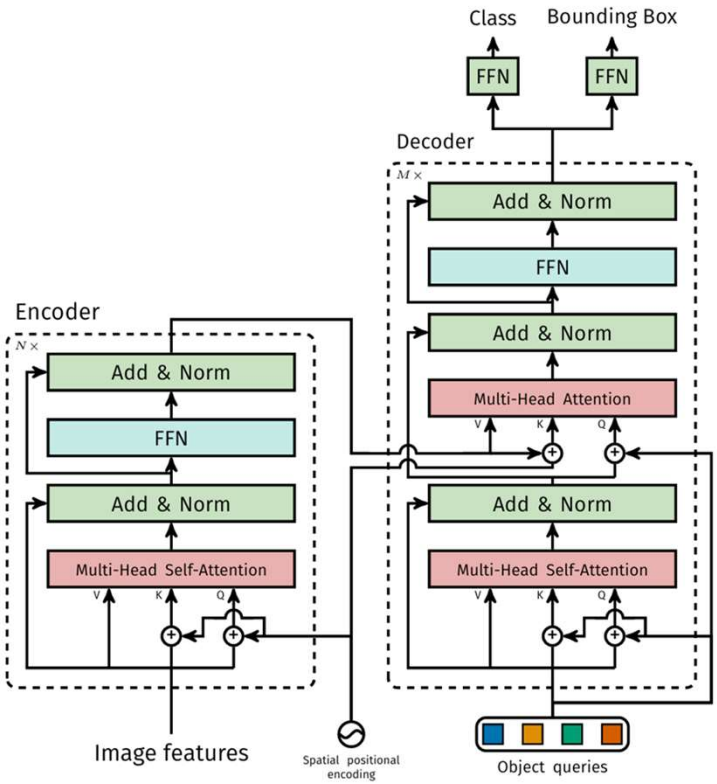
GPU Memory Hierarchy

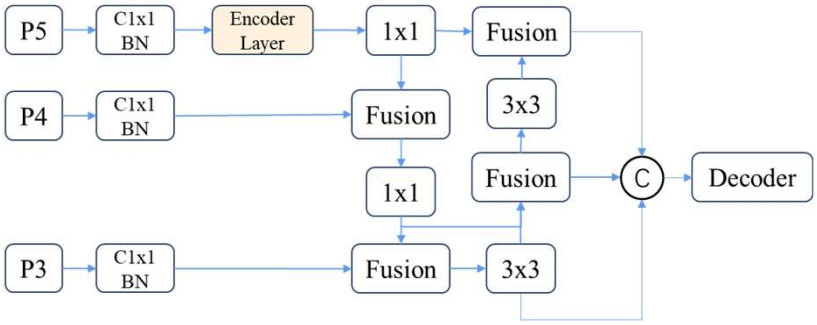


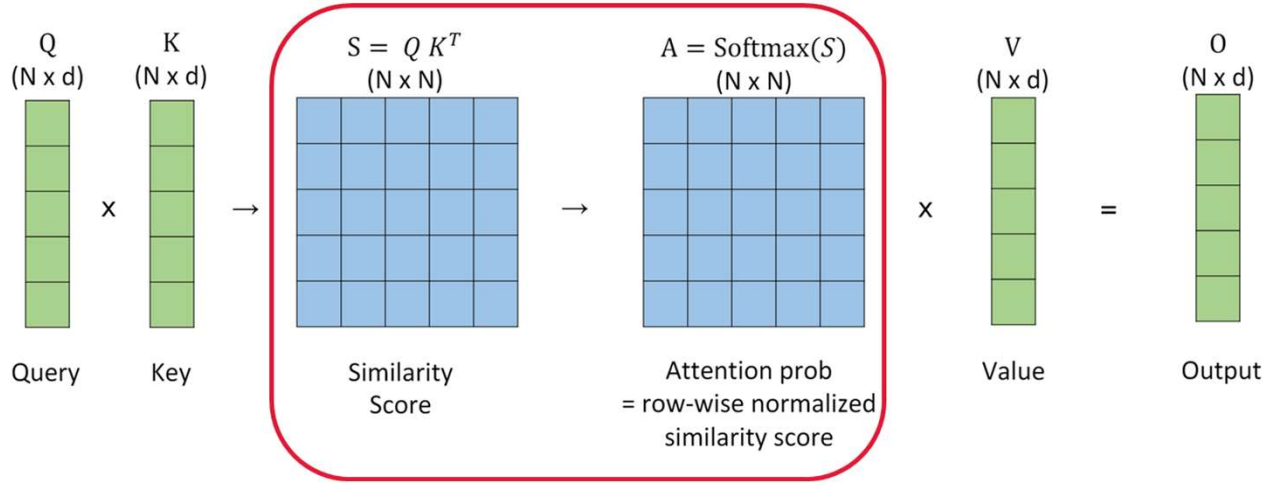
CUTLASS GEMM Structural Model

https://developer.nvidia.com/blog/cutlass-linear-algebra-cuda/

Is the Mamba-based detector possible?







Attention scales quadratically in sequence length N.