DN-DETR: Accelerate DETR Training by Introducing Query DeNoising

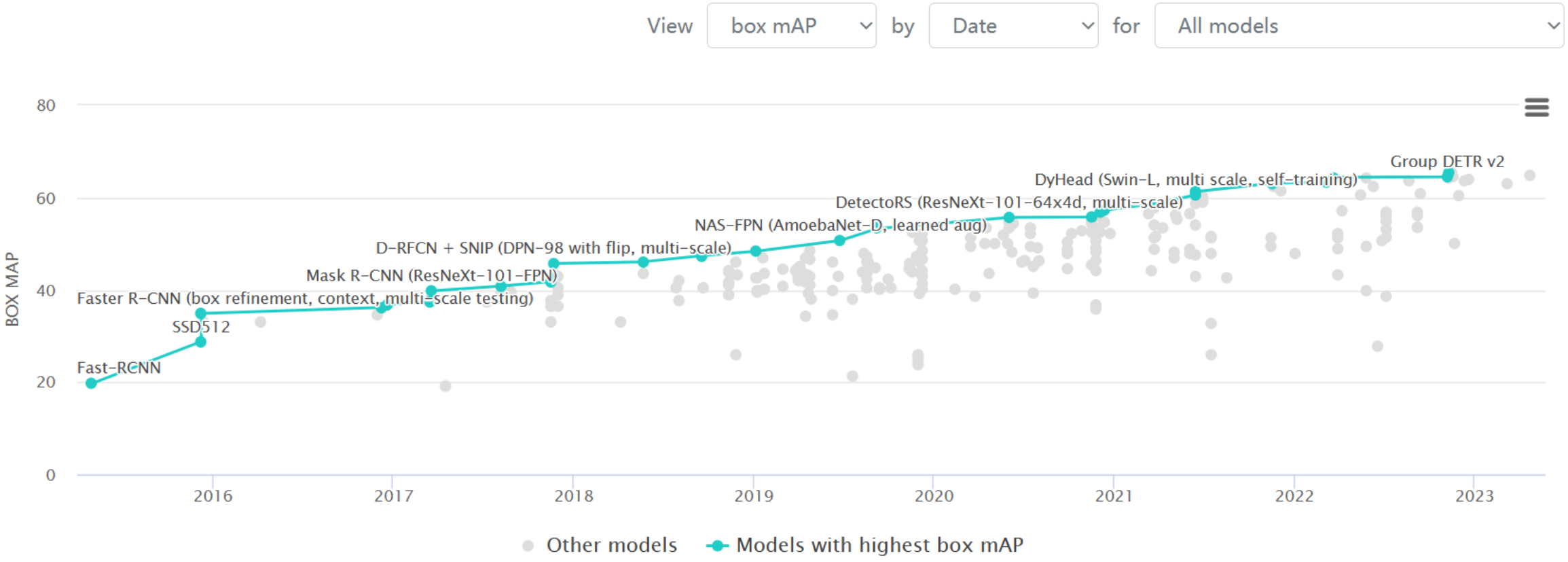
CVPR 2022 Oral

Feng Li∗, Hao Zhang∗, Shilong Liu, Jian Guo, Lionel M. Ni, and Lei Zhang, IEEE Fellow

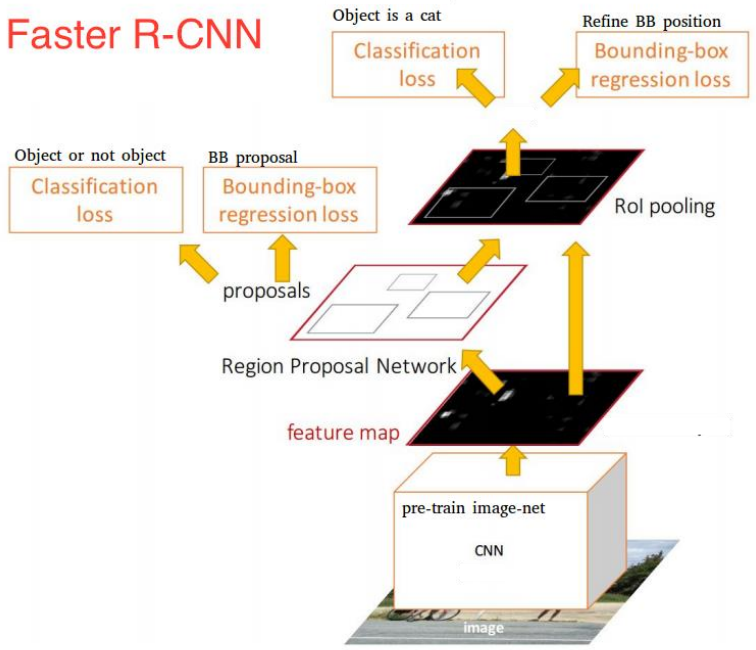
- Feng Li and Hao Zhang are with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong.
- Shilong Liu is with the Department of Computer Science and Engineering, Tsinghua University, Beijing.
- Lionel Ni is the president of The Hong Kong University of Science and Technology (Guangzhou).
- Jian Guo and Lei Zhang are with IDEA.
- ∗ denotes equal contribution.

Object Detection on COCO test-dev

Leaderboard Dataset



Problems & Challenges



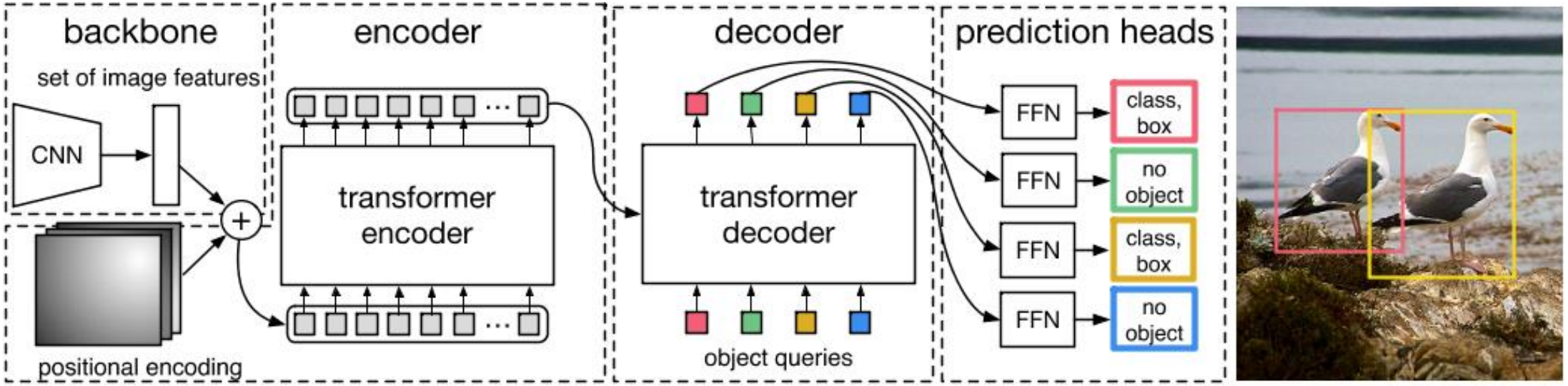
Classic detectors:

- Handcraft components like NMS and anchors

DETR-like models: (Detection transformer)

- Set-based prediction
- Bipartite matching
- End-to-end optimization

DETR: End-to-End Object Detection with Transformers

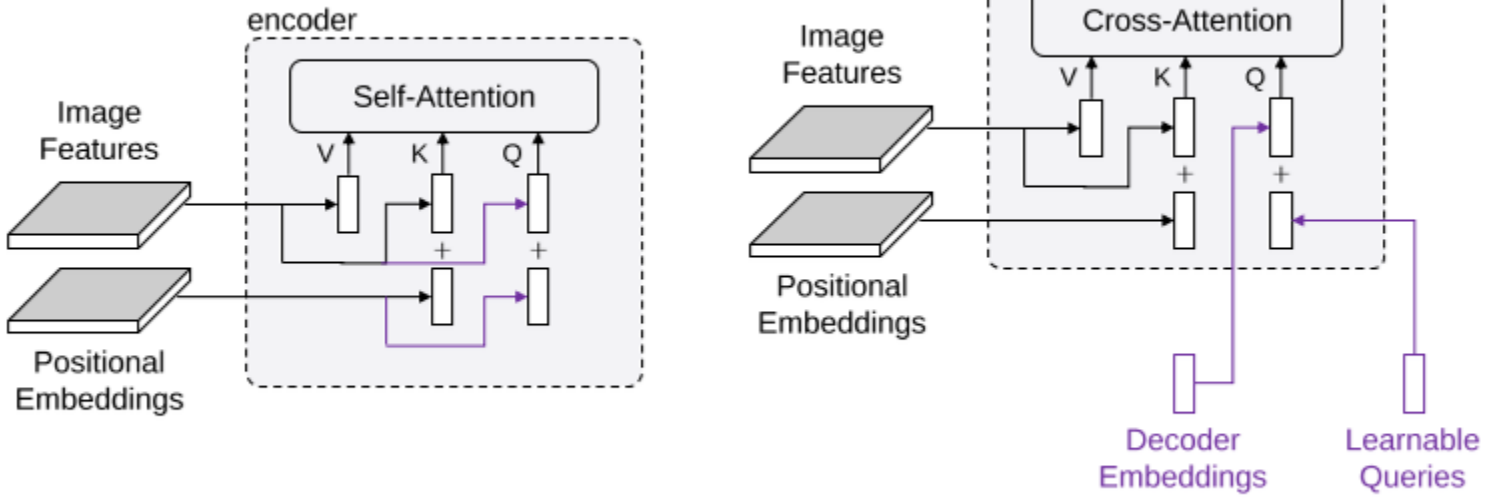


Problems:

- Slow training convergence.
- No clear meaning for positional queries.
- Sub performances.

[1]CARION N, MASSA F, SYNNAEVE G, 等. End-to-End Object Detection with Transformers[Z]//arXiv: Computer Vision and Pattern Recognition. 2020.

DETR: End-to-End Object Detection with Transformers



(a) Self-attention in encoder of DETR

(b) Cross-attention in decoder of DETR

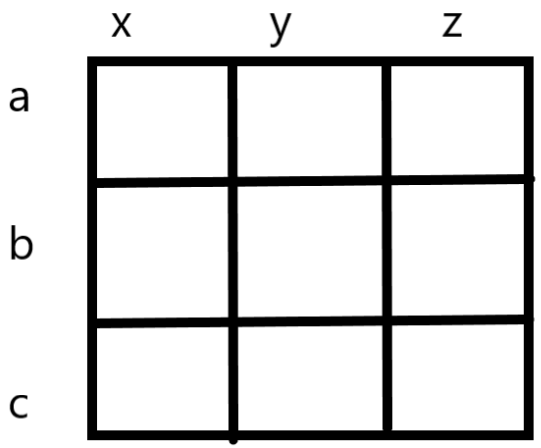
Decoder embeddings:

- Decoder self-attention output
- Encoded image feature
- A content part

Learnable queries:

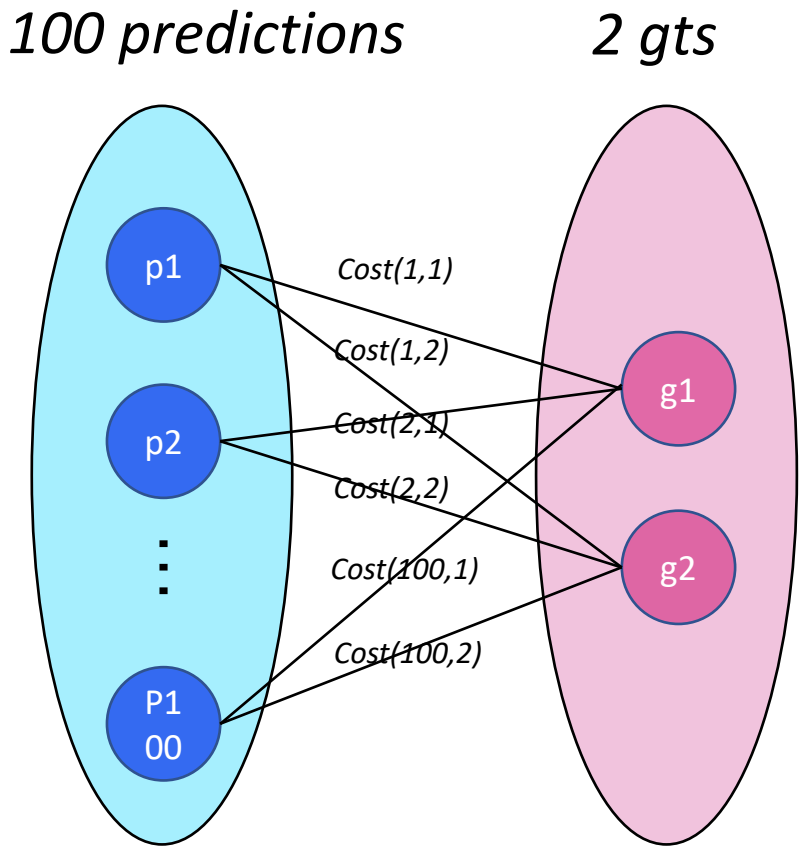
- Object queries(100*256)
- Position embedding
- A position part

Bipartite Graph Matching



- What is Bipartite Graph?
- What is Hungarian matching?
  - Worker: a、b、c
  - Work: x、y、z
  - When given a cost matrix, assign different workers to work, so as to minimize final total cost.

Bipartite Graph Matching

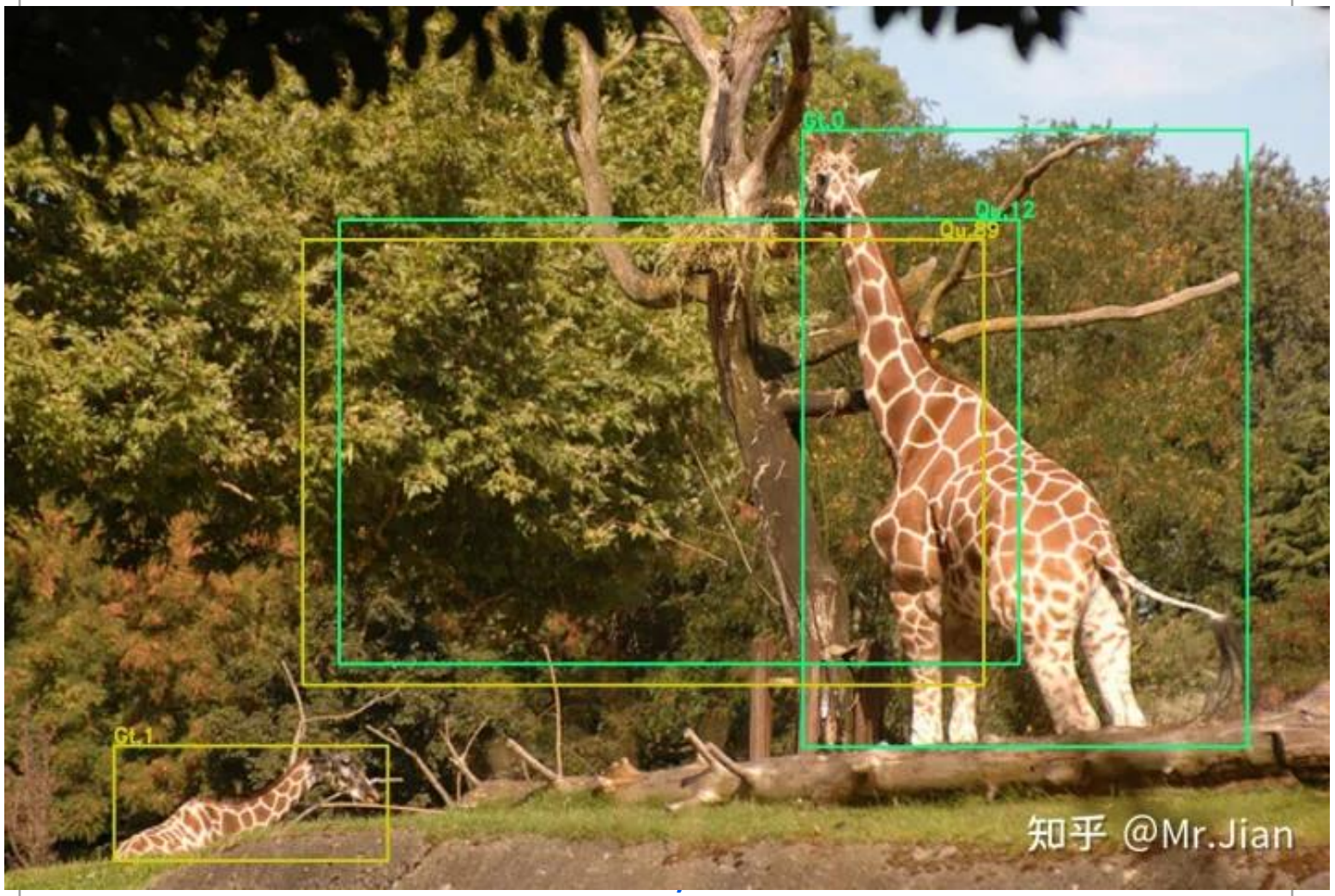


$\hat{y}_{\sigma(i)}$ $y_i = (c_i, b_i)$

$where\ c_i\ may\ be\ \emptyset\ \ (no\ object)$

- $\mathcal{L}_{\text{match}}\left(y_i, \hat{y}_{\sigma(i)}\right) =$
- $-\mathbb{1}_{\{c_i \neq \emptyset\}} \hat{p}_{\sigma(i)}(c_i) + \mathbb{1}_{\{c_i \neq \emptyset\}} \mathcal{L}_{\text{box}}\left(b_i, \hat{b}_{\sigma(i)}\right)$

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



[2]LIU S, LI F, ZHANG H, 等. DAB-DETR: Dynamic Anchor Boxes are Better Queries for DETR[Z].

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

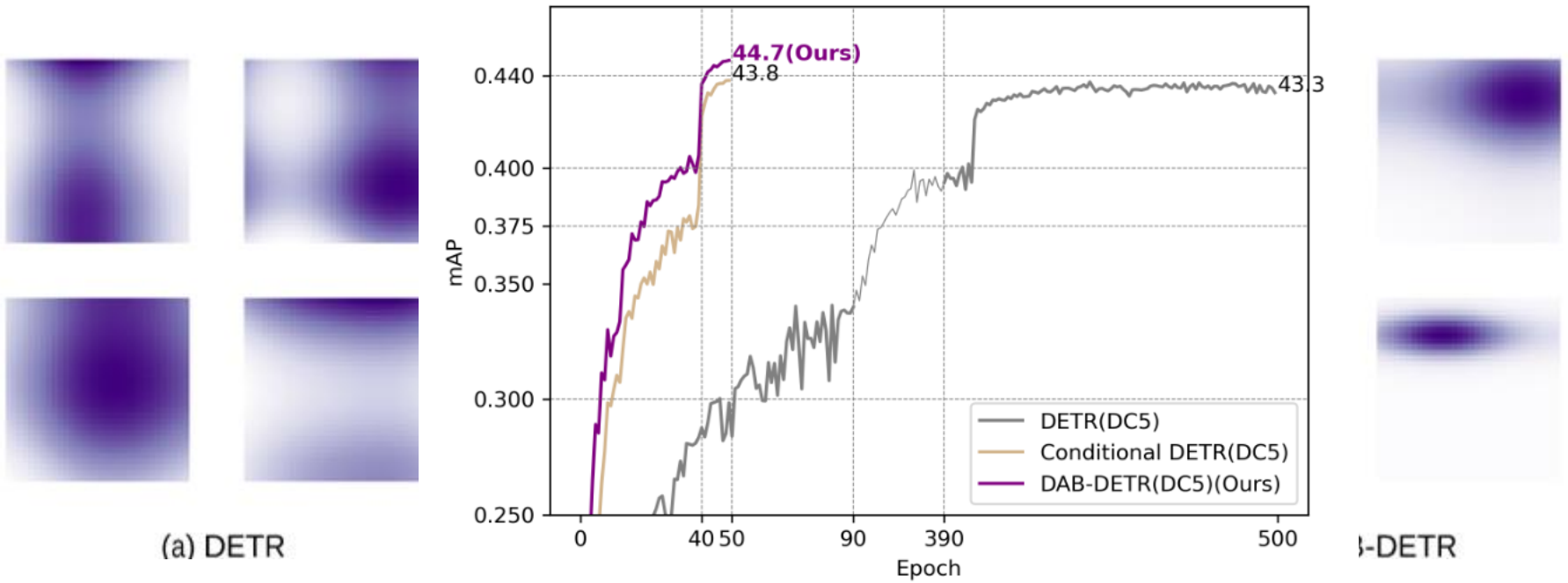


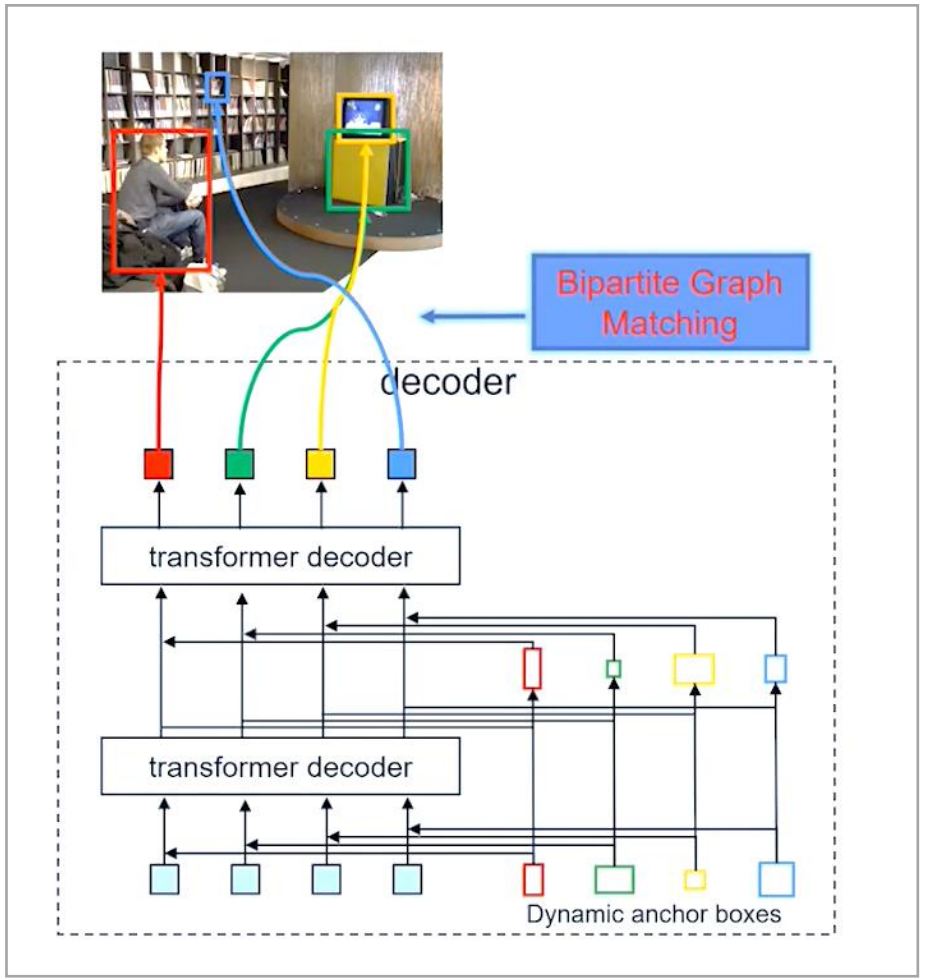
Figure 13: Convergence curves of DETR, Conditional DETR, and our DAB-DETR. All models are trained under the R50(DC5) settinng.

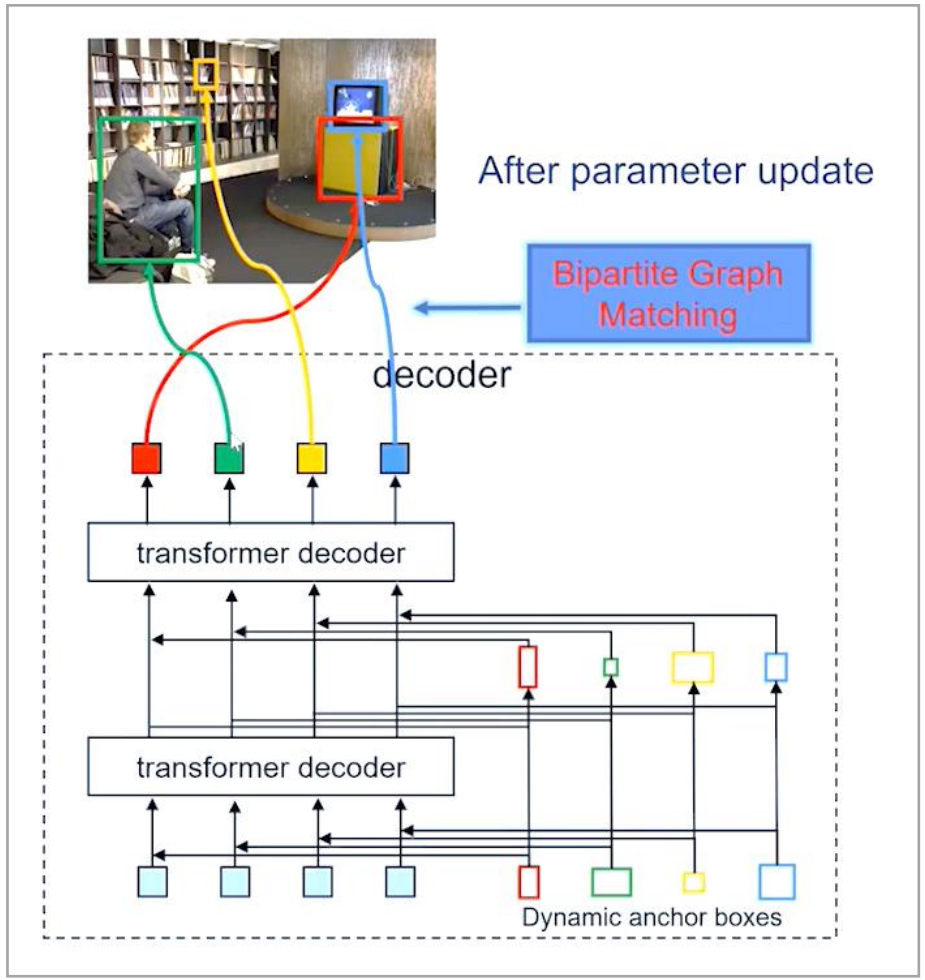
Contributions:

- A deeper understanding of the role of queries.
- Solving the problem of slow convergence and improved performance in DETR

Why Is DETR Training Slow?

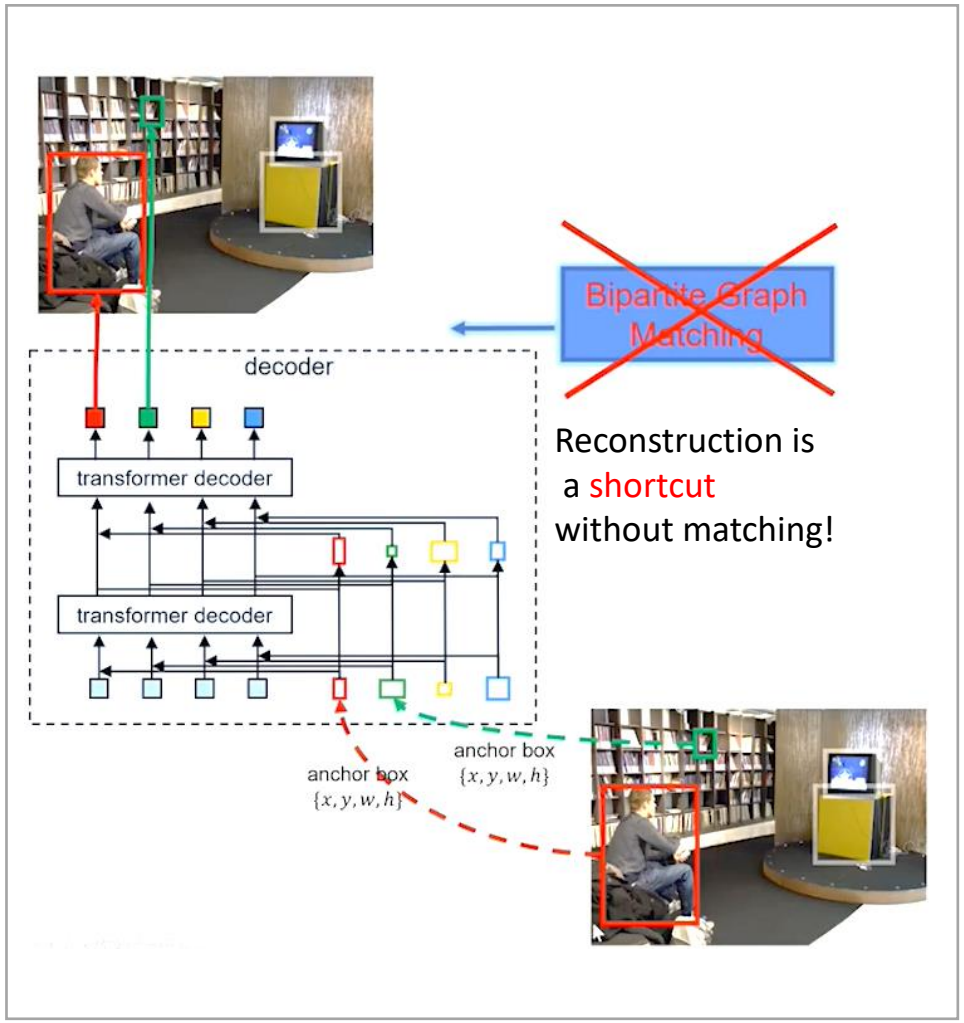
Instability of bipartite graph matching





- A small change in the cost matrix may cause an enormous change in the matching result, which will further lead to inconsistent optimization goals for decoder queries.

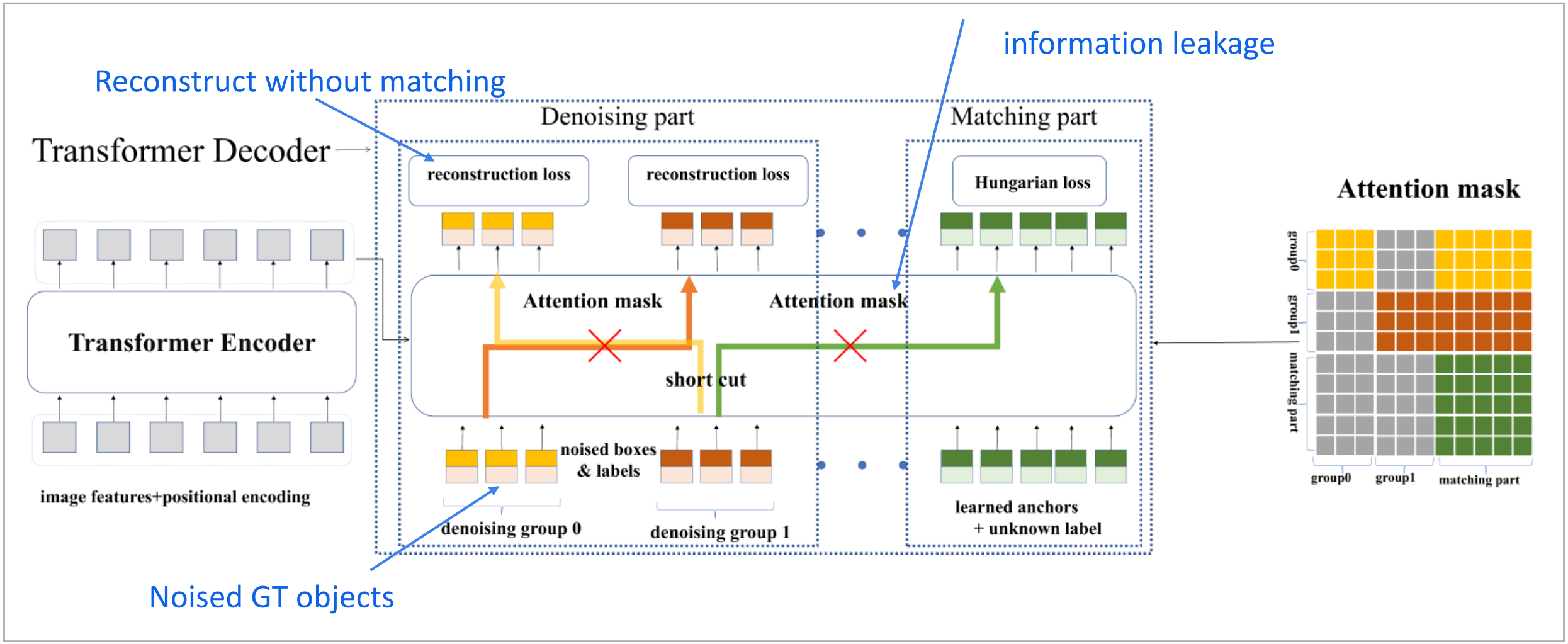
How To Address This Problem?



- Anchor box reconstruction:
  - Input: GT boxes as Anchor boxes
  - Output: Predict Box and Cls for each anchor

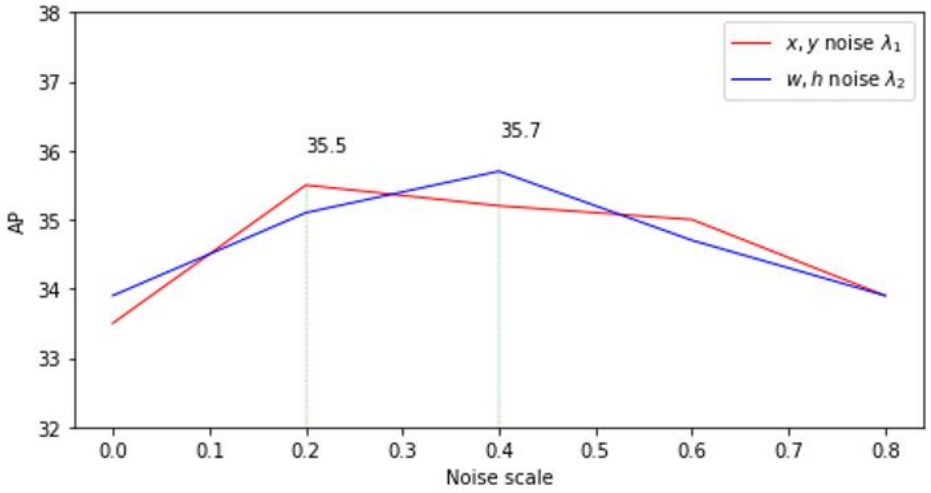
  Without bipartite graph matching.
- It did not work **until we add noises to anchor boxes**.
- The anchor box formulation makes this task very easy and straightforward to setup.

DN-DETR framework



- we leverage a denoising task as **a training shortcut** to make relative offset learning easier, as the denoising task **bypasses bipartite matching**.

Denoising : Noised Boxes



- Box Noising in two ways:
  - center shifting

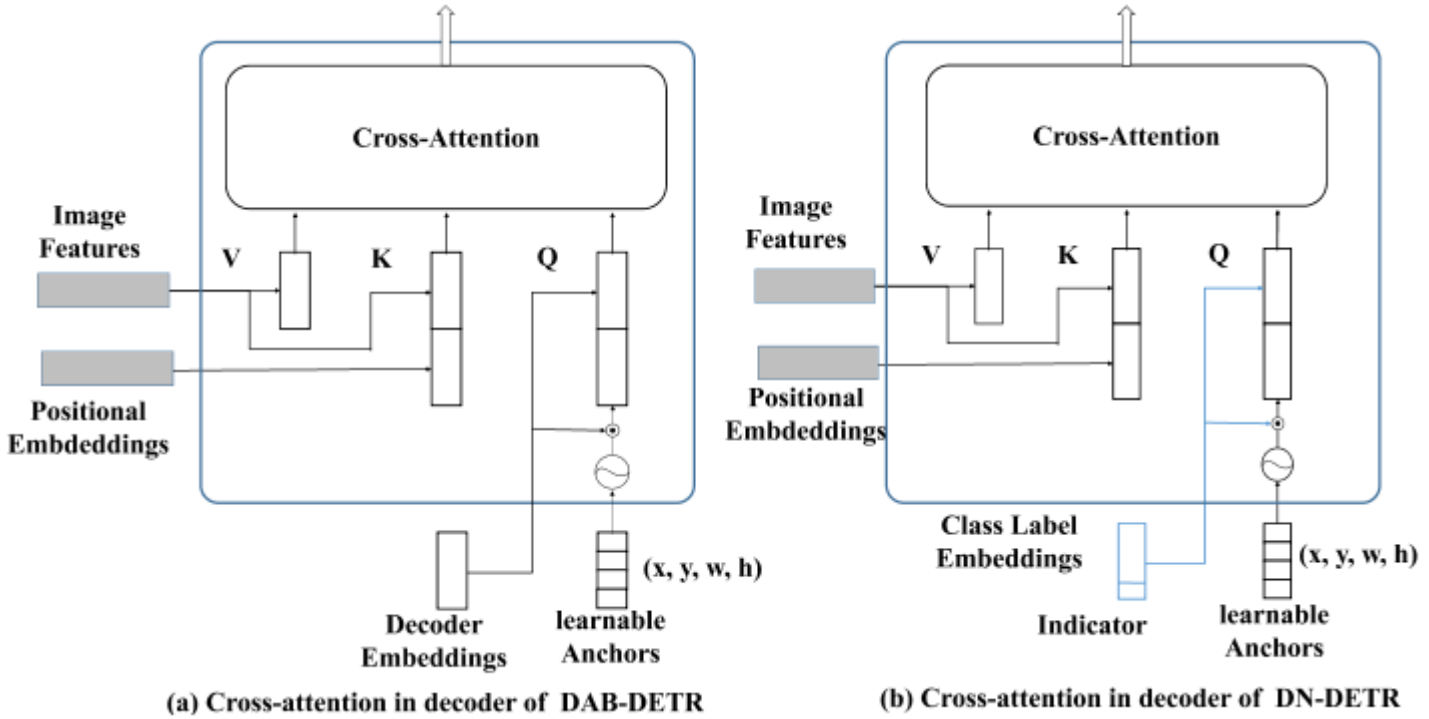
    $|\Delta x| = \lambda_1 w/2 \quad |\Delta y| = \lambda_1 h/2 \quad \lambda_1 \epsilon (0,1)$

  - box scaling

    $|\Delta w| = \lambda_2 w \quad |\Delta h| = \lambda_2 h \quad \lambda_2 \epsilon (0,1)$

Denoising : Noised Labels

- Label embedding:
  - DN-DETR specify the **decoder embedding** as **label embedding** and add an indicator to differentiate denoising task and matching task.



- Label Noising
  - adopt label flipping, which means we randomly flip some ground-truth labels to other labels

Denoising : DN Groups

- Each group is a noised version of all GT objects.

$$\mathbf{q} = \{\mathbf{g}_0, \mathbf{g}_1, \cdots, \mathbf{g}_{\mathbf{P}-1}\}$$

- Each denoising group contains M queries where M is the number of GT objects in the image.

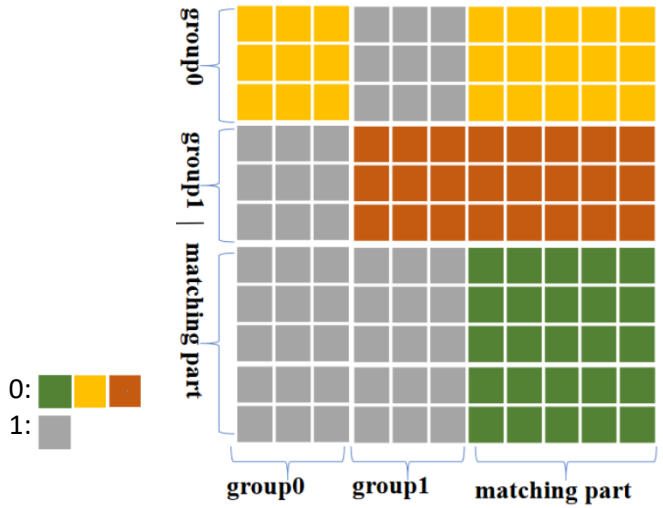
$$\mathbf{g}_\mathbf{p} = \{q_0^p, q_1^p, \cdots, q_{M-1}^p\}$$

- gt -> query : one to many

	No Group	1 Group	5 Groups
R50	42.2	43.4	44.1
R50-DC5	44.5	45.6	46.3
R101	43.5	45.0	45.2
R101-DC5	45.8	46.5	47.3

Denoising : Attention Mask

- Two types of potential information leakage:
  - the matching part may see the noised GT objects and easily predict GT objects.
  - one noised version of a GT object may see another version.
- Attention mask:
  - aij = 1 means the i-th query cannot see the j-th query and aij = 0 otherwise.



Box Denoising	Label Denoising	Attention Mask	AP
✓	✓	✓	43.4
✓		✓	43.0
		✓	42.2
✓	✓		24.0

Analysis of Training Instability

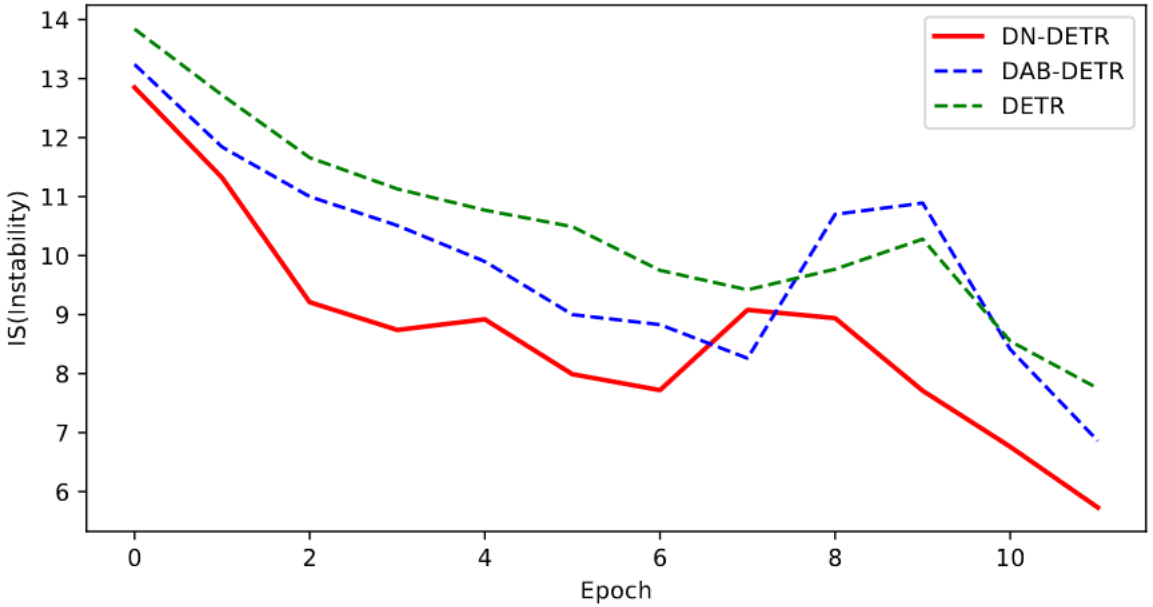


Fig. 2. The IS of DAB-DETR and DN-DETR during training. For each method, we train 12 epoch on the same setting. We test the change of the Hungarian matching between each two epochs on the Validation set as the IS.

- IS (InStability)

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

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

- IS is used to evaluate how many queries are matched with different targets after one epoch of parameter updating.

Main Results - 50 Epochs

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	R50
Faster RCNN-FPN-R50 [15]	108	42.0	62.1	45.5	26.6	45.5	53.4	180	42M	
Anchor DETR-R50 [18]	50	42.1	63.1	44.9	22.3	46.2	60.0	–	39M	
Conditional DETR-R50 [12]	50	40.9	61.8	43.3	20.8	44.6	59.2	90	44M	
DAB-DETR-R50 [11]	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	R101
Faster RCNN-FPN-R101 [15]	108	44.0	63.9	47.8	27.2	48.1	56.0	246	60M	
Anchor DETR-R101 [18]	50	43.5	64.3	46.6	23.2	47.7	61.4	–	58M	
Conditional DETR-R101 [12]	50	42.8	63.7	46.0	21.7	46.6	60.9	156	63M	
DAB-DETR-R101 [11]	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	R50-DC5
Anchor DETR-DC5-R50 [18]	50	44.2	64.7	47.5	24.7	48.2	60.6	151	39M	
Conditional DETR-DC5-R50 [12]	50	43.8	64.4	46.7	24.0	47.6	60.7	195	44M	
• DAB-DETR-DC5-R50 [11]	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	R101-DC5
Anchor DETR-R101 [18]	50	45.1	65.7	48.8	25.8	49.4	61.6	–	58M	
Conditional DETR-DC5-R101 [12]	50	45.0	65.5	48.4	26.1	48.9	62.8	262	63M	
DAB-DETR-DC5-R101 [11]	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	

Main Results - 12 Epochs(1x)

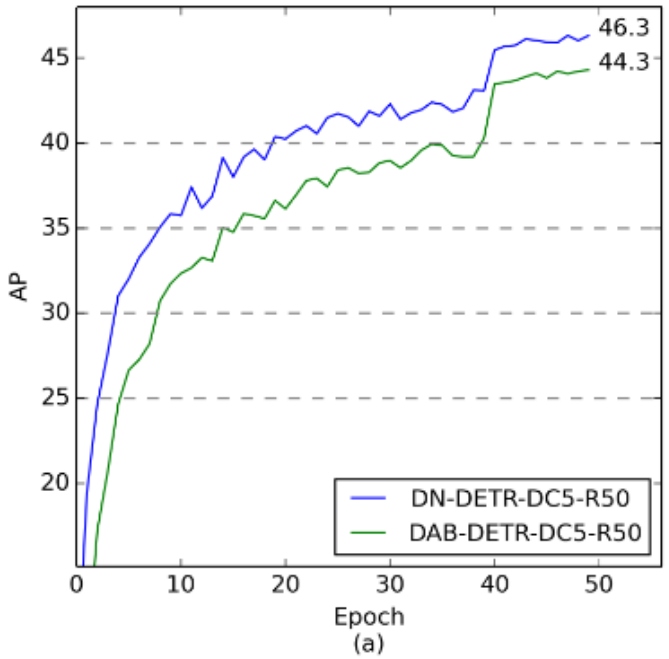
Model	MultiScale	#epochs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	GFLOPs	Params
• Faster R50-FPN 1x [15]	✓	12	37.9	58.8	41.1	22.4	41.1	49.1	180	40M
DETR-R50 1x [1]		12	15.5	29.4	14.5	4.3	15.1	26.7	86	41M
• DAB-DETR-DC5-R50 [11]		12	38.0	60.3	39.8	19.2	40.9	55.4	216	44M
• DN-DETR-DC5-R50		12	**41.7(+3.7)**	61.4	44.1	21.2	45.0	60.2	216	44M
• Deformable DETR-R50 1x [20]	✓	12	37.2	55.5	40.5	21.1	40.7	50.5	173	40M
Dynamic DETR-R50† 1x (without dynamic encoder)	✓	12	40.2	58.6	43.4	−−	–	–	–	–
• Dynamic DETR-R50† 1x [4]	✓	12	42.9	61.0	46.3	24.6	44.9	54.4	–	–
• DN-Deformable-DETR-R50 [4]	✓	12	**43.4**	61.9	47.2	24.8	46.8	59.4	195	48M
DAB-DETR-DC5-R101 [11]		12	40.3	62.6	42.7	22.2	44.0	57.3	282	63M
DN-DETR-DC5-R101		12	**42.8(+2.5)**	62.9	45.7	23.3	46.6	61.3	282	63M
Faster R101 FPN [15]	✓	108	44.0	63.9	47.8	27.2	48.1	56.0	246	60M
DN-Deformable-DETR-R101	✓	12	**44.1**	62.8	47.9	26.0	47.8	61.3	275	67M

R50-DC5

R50+ Deformable

Convergence Curves

DN vs. DAB
w/ DETR-R50-DC5



DN vs. DAB
w/ Deformable DETR-R50

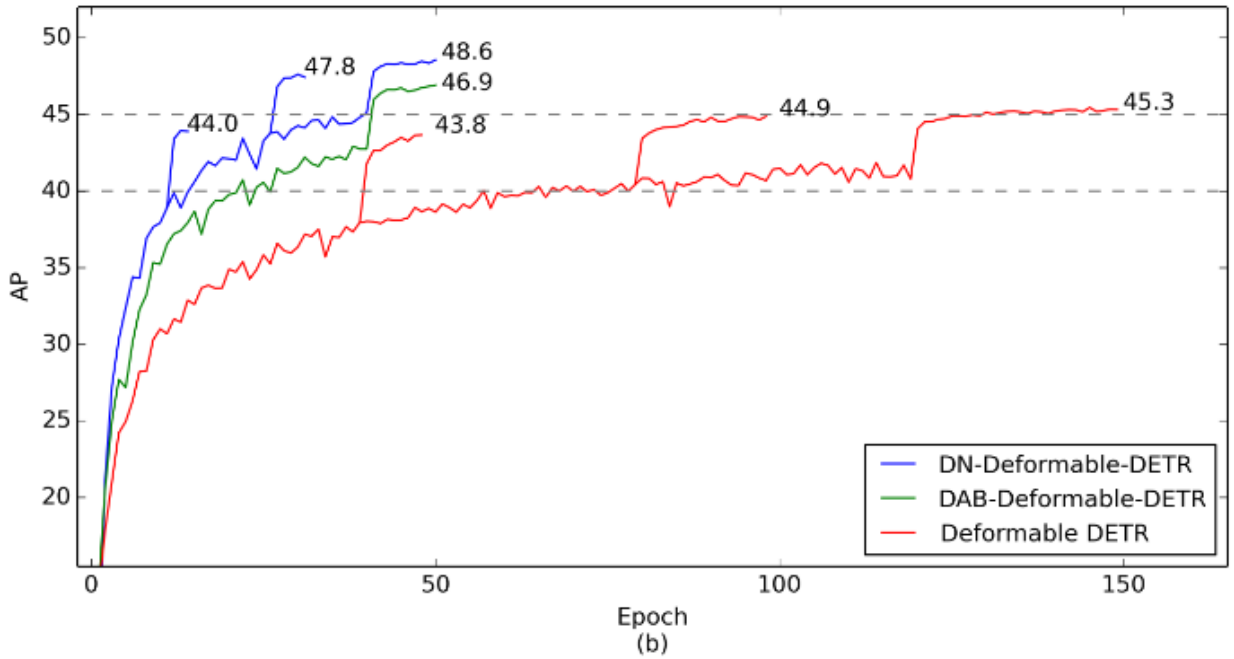


Fig. 7. (a) Convergence curves of DAB-DETR and DN-DETR with ResNet-DC5-50. Before learning rate drop, DN-DETR achieves 40 AP in 20 epochs, while DAB-DETR needs 40 epochs. (b) Convergence curves of multi-scale models with ResNet-50. With learning rate drop, DN-Deformable-DETR achieves 47.8 AP in 30 epochs, which is 0.9 AP higher than the converged DAB-Deformable-DETR.

Thanks for your listening