# Mr. DETR: Instructive Multi-Route Training for Detection Transformers Supplementary Material

Chang-Bin Zhang[1] Yujie Zhong[2] Kai Han[1†]
[1]Visual AI Lab, The University of Hong Kong [2]Meituan Inc.
cbzhang@connect.hku.hk jaszhong@hotmail.com kaihanx@hku.hk

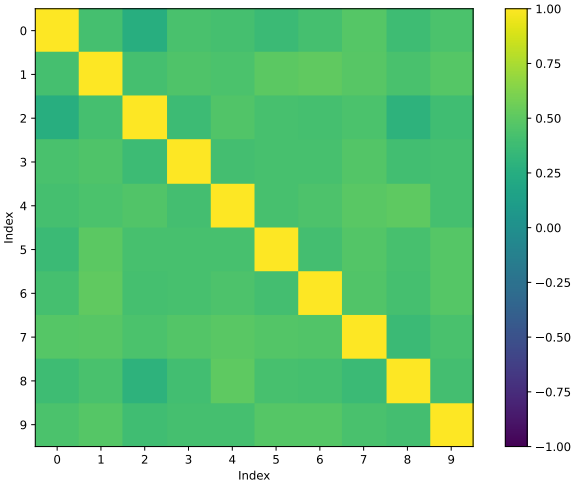
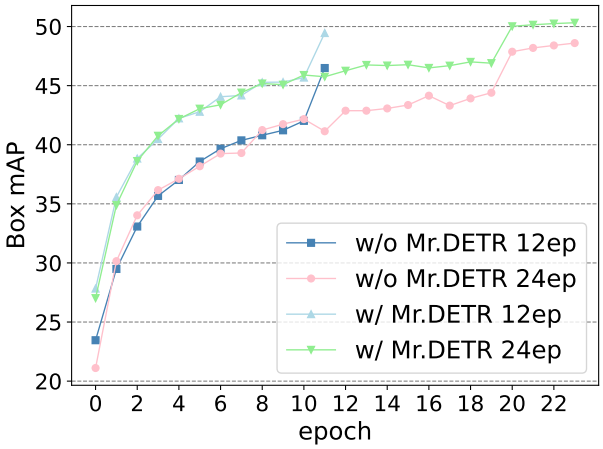



Figure 1. The cosine similarity between 10 instruction tokens used in Deformable-DETR++ [12].

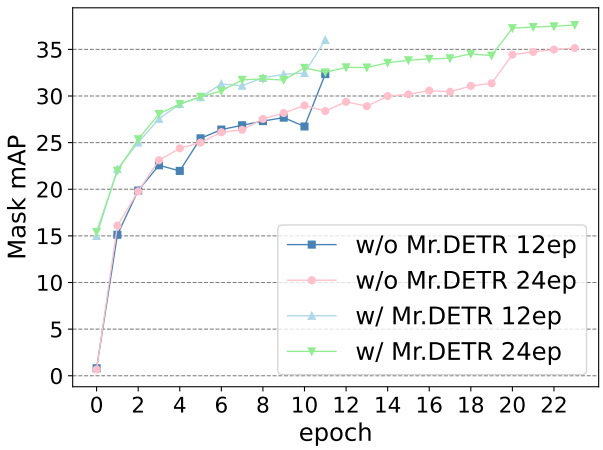
## A. Analysis of Instructive Self-Attention

We present the PyTorch-style pseudo-code for our proposed instructive self-attention method in Alg. 1. As detailed in our primary manuscript, our multi-route training approach involves three training routes. Route-2 serves as the primary route designated for one-to-one prediction, which is the same as the baseline model. Route-1 acts as an auxiliary route with an independent FFN aimed at one-to-many predictions. Meanwhile, Route-3 operates as an auxiliary route incorporating our novel instructive self-attention to facilitate one-to-many predictions. In this approach, we establish a collection of trainable tokens as instructions to direct the object queries and subsequent modules in executing one-to-many prediction. These trainable instruction tokens are affixed to the input object queries through concatenation, allowing for the dynamic and adaptable transmission of instructions via self-attention mechanisms. The resulting output from the instruction tokens after undergoing self-attention is not retained, as these tokens do not possess the capability to locate objects.

†Corresponding author.




(a) Evaluation results of box




(b) Evaluation results of mask

Figure 2. **Evaluation results of each epoch.** We utilize the Deformable-DETR++ (300 queries) as the baseline model, which is trained for 12 and 24 epochs, respectively.

In the main paper, we ablate the configurations of the proposed instructive self-attention, including the number of instruction tokens and layers using instruction tokens. Ablation studies demonstrate that the model is not sensitive to the layers and numbers of instruction tokens, due to the following reasons: **(i)** Regarding the impact of layers of instruction

**Algorithm 1** Pseudo-code of Instructive Self-Attention in a PyTorch-like style.

```
class InstructAttn(nn.Module):
    def __init__(self, embed_dim, num_heads, num_ins):
        # embed_dim: the embedding dimension used
        # num_heads: the number of heads in self-attention
        # num_ins: the number of instruction tokens
        self.ins_sa = nn.MultiheadAttention(
            embed_dim = embed_dim,
            num_heads = num_heads,
        )

        # define learnable instruction tokens
        self.instruct_o2m = nn.Embedding(num_ins, embed_dim)
        self.pos_o2m = nn.Embedding(num_ins, embed_dim)
        self.num_ins = num_ins

    def forward(self, query, key, value, query_pos, key_pos, route="route-3"):
        # route: identify the current route
        query = query + query_pos
        key = key + key_pos
        if route == "route-3":
            ins_token = self.instruct_o2m.weight
            ins_pos = self.pos_o2m.weight

            # concatenate instruction tokens into the input sequence
            query = torch.cat([query, ins_token + ins_pos], dim=1)
            key = torch.cat([key, ins_token + ins_pos], dim=1)
            value = torch.cat([value, ins_token], dim=1)

        # perform self-attention
        out = self.ins_sa(
            query, key, value
        )
        if route == "route-3":
            # discard the corresponding output of instruction tokens
            out = out[:, :-self.num_ins, :]
        return out
```

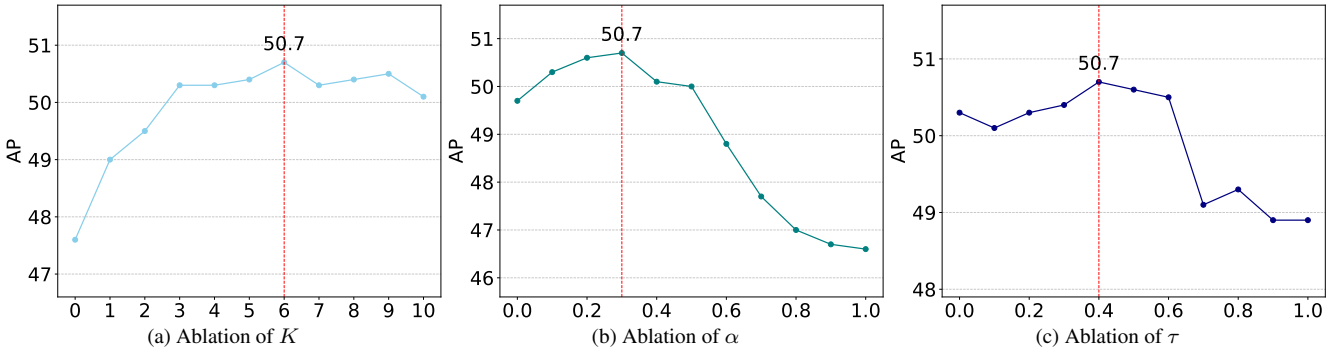



Figure 3. **Influence of hyper-parameters $K$, $\alpha$ and $\tau$ in the one-to-many assignment.** (a) Influence of $K$ for selecting top-$K$ positive candidates. (b) Influence of $\alpha$ that denotes the weight of classification confidence when forming the matching score $M$. (c) Influence of $\tau$ that is used to filter out low-quality candidates.

tokens, we hypothesize that the information from instruction tokens in the first layer can be retained and utilized by subsequent layers. Meanwhile, residual connections across transformer decoder layers may help preserve the instruction information for later layers. **(ii)** To study the impact of the number of instruction tokens, as shown in Fig. 1, we calculate the cosine similarity between the 10 instruction tokens and find that most of them are very similar. This indicates that most instruction tokens may play a similar role, making the model insensitive to their number.

## B. Convergence Curves

Employing Deformable-DETR++ [12] with 300 queries, we perform training on the instance segmentation task both with and without the integration of our proposed approach. Consistent with established methods [4, 5, 10–12], models

| Route | Epoch | NMS | Box | | | | | | Mask | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | mAP | $AP_{50}$ | $AP_{75}$ | $AP_s$ | $AP_m$ | $AP_l$ | mAP | $AP_{50}$ | $AP_{75}$ | $AP_s$ | $AP_m$ | $AP_l$ |
| Baseline | 12 | | 46.5 | 64.2 | 50.8 | 28.8 | 50.0 | 60.7 | 32.4 | 55.7 | 32.6 | 11.9 | 35.6 | 54.4 |
| Baseline | 12 | ✔ | 46.7 | 65.0 | 50.7 | 29.0 | 50.1 | 60.9 | 32.5 | 56.3 | 32.6 | 12.0 | 35.8 | 54.8 |
| Route-2 (primary) | 12 | | 49.5 | 66.6 | 54.1 | 30.3 | 52.6 | 64.7 | 36.0 | 59.8 | 37.2 | 13.6 | 39.6 | 59.7 |
| Route-2 (primary) | 12 | ✔ | 49.7 | 67.6 | 54.1 | 30.4 | 52.8 | 65.0 | **36.2** | 60.5 | 37.2 | 13.7 | 39.8 | 60.1 |
| Route-1 (auxiliary) | 12 | | 14.5 | 18.8 | 16.0 | 15.6 | 21.6 | 19.1 | 10.9 | 17.1 | 11.6 | 5.9 | 13.4 | 20.2 |
| Route-1 (auxiliary) | 12 | ✔ | 49.8 | 67.3 | 54.6 | 31.0 | 53.0 | 65.3 | 36.1 | 60.1 | 37.2 | 13.9 | 39.4 | 59.9 |
| Route-3 (auxiliary) | 12 | | 14.5 | 18.7 | 15.9 | 15.3 | 22.1 | 19.7 | 10.8 | 17.0 | 11.5 | 5.6 | 13.6 | 20.3 |
| Route-3 (auxiliary) | 12 | ✔ | **49.9** | 67.3 | 54.6 | 30.8 | 53.2 | 65.4 | **36.2** | 60.2 | 37.3 | 13.6 | 39.5 | 59.8 |
| Baseline | 24 | | 48.6 | 66.4 | 53.1 | 30.8 | 51.9 | 62.9 | 35.1 | 59.1 | 36.0 | 14.3 | 38.9 | 57.5 |
| Baseline | 24 | ✔ | 48.6 | 67.1 | 52.8 | 30.9 | 51.8 | 63.0 | 35.2 | 59.6 | 35.9 | 14.4 | 39.0 | 57.7 |
| Route-2 (primary) | 24 | | 50.3 | 68.0 | 54.7 | 31.5 | 53.2 | 65.0 | **37.6** | 61.4 | 38.9 | 15.1 | 41.4 | 60.5 |
| Route-2 (primary) | 24 | ✔ | 50.4 | 68.7 | 54.6 | 31.6 | 53.3 | 65.2 | 37.7 | 61.9 | 38.9 | 15.1 | 41.6 | 60.9 |
| Route-1 (auxiliary) | 24 | | 14.0 | 18.1 | 15.2 | 15.4 | 21.1 | 18.4 | 10.7 | 16.6 | 11.3 | 5.8 | 13.4 | 19.1 |
| Route-1 (auxiliary) | 24 | ✔ | **50.4** | 68.2 | 55.1 | 31.5 | 53.5 | 65.2 | **37.6** | 61.7 | 38.8 | 14.9 | 41.2 | 60.4 |
| Route-3 (auxiliary) | 24 | | 14.1 | 18.3 | 15.3 | 15.6 | 21.2 | 18.8 | 10.7 | 16.7 | 11.3 | 5.9 | 13.3 | 19.3 |
| Route-3 (auxiliary) | 24 | ✔ | **50.4** | 68.2 | 54.8 | 31.7 | 53.4 | 65.2 | **37.6** | 61.6 | 38.8 | 14.9 | 41.2 | 60.5 |

Table 1. **The detailed performance of each route in Mr. DETR.** The Deformable-DETR++ [12] employing 300 queries serves as our baseline model. 'Route-2': the primary route employed for one-to-one prediction, identical in functionality to the baseline model. 'Route-1': the auxiliary route for one-to-many prediction, which is built with an independent FFN. 'Route-3': the auxiliary route for one-to-many prediction, which utilizes our proposed instructive self-attention.

are trained using 12 and 24 epoch schedules, respectively. The learning rate is reduced by a factor of 0.1 at the 11th and 20th epochs according to the 12 and 24 epoch schedules, respectively. We illustrate the evaluation results for bounding box predictions in Fig. 2(a) and for instance mask predictions in Fig. 2(b). The evaluation results demonstrate that our approach significantly enhances the training process of the baseline model.

## C. Impact of Hyper-Parameters

Fig. 3 shows the influence of the hyper-parameters $K$, $\alpha$, and $\tau$ in the one-to-many assignment [4, 8, 11]. In Fig. 3a, we observe that as the number of positive candidates increases, the model achieves its highest performance when $K = 6$. However, when $K > 7$, the one-to-many assignment increases the difficulty of removing duplicates in the primary route, leading to decreased performance. For the weight of classification confidence $\alpha$, as shown in Fig. 3b, the model achieves the best performance at $\alpha = 0.3$. Similarly, in Fig. 3c, the performance improves as the filter threshold $\tau$ increases, reaching its peak at $\tau = 0.4$. Beyond this value, the performance declines, potentially due to the filtering out of many high-quality candidates. In our method, we empirically set $K = 6$, $\alpha = 0.3$, and $\tau = 0.4$.

## D. Detailed Performance of Each Route

As detailed in Sec. B, we train the Mr. DETR for the instance segmentation task based on Deformable-DETR++ [12]. We report the evaluation results of each route of our method in Tab. 1. Experimental results indicate that the primary route is adept at accomplishing one-to-one prediction, as evidenced by the fact that Non-Maximum-Suppression (NMS) yields only a minor improvement of approximately 0.1 - 0.2% in both box mAP and mask mAP. Conversely, for the auxiliary routes, the application of NMS significantly improves the performance (about 35% and 25% in terms of box mAP and mask mAP) of Route-1 and Route-3, highlighting their capability for effective one-to-many prediction. This further substantiates that our introduced instructive self-attention is proficient in efficiently guiding object queries for one-to-many prediction.

## E. Experiments on the Objects365

Objects365 [9] is a large-scale dataset with 365 classes, which contains about 2,000,000 images. To further verify the scalability of our method on the large-scale dataset, we conduct experiments on Objects365 using Deformable-DETR++ [12] model with 900 queries. To save training time, we train the baseline model and our method for 4 epochs only. The initial learning rate is set to 2e-4 and decays at the third epoch. Other training settings are the same as the model trained on the COCO dataset [6]. We report the performance comparison in Tab. 2.

## F. Training Cost

We measure the training time of different methods in Tab. 3. The training time denotes the average duration of each epoch. We evaluate training time on 8 NVIDIA 3090 GPUs with

| Models | Epochs | Queries | mAP | $AP_{50}$ | $AP_{75}$ | $AP_s$ | $AP_m$ | $AP_l$ |
|---|---|---|---|---|---|---|---|---|
| Deformable-DETR++ [12] | 4 | 900 | 30.4 | 40.8 | 33.1 | 16.1 | 30.1 | 39.1 |
| w/ Mr. DETR | 4 | 900 | **32.7** (+2.3) | **42.7** (+1.9) | **35.8** (+2.7) | **17.1** (+1.0) | **32.3** (+2.2) | **42.6** (+3.5) |

Table 2. **Experiments on the large-scale Obejcts365 dataset [9].** Deformable-DETR++ and our model use the ResNet-50 [3] backbone.

a batch size of 16. The experimental results indicate that our proposed method achieves an effective trade-off between performance and training costs.

| Model | Time | GFLOPs | mAP |
|---|---|---|---|
| Deformable-DETR++ [12] | 84 | 234.9 | 47.0 |
| H-DETR [5] | 104 (+20) | 265.0 | 48.7 (+1.7) |
| DAC-DETR [4] | 94 (+10) | - | - |
| MS-DETR [11] | 96 (+12) | - | 48.8 (+1.8) |
| Group-DETR [2] | 123 (+39) | - | - |
| Mr. DETR (Ours) | 101 (+17) | 258.0 | **49.5** (+2.5) |

Table 3. **Comparison of training time (minutes) and training GFLOPs of various methods.** All methods utilize Deformable-DETR++ with 300 queries as the baseline. The training time represents the average duration per training epoch.

## G. Performance of Intermediate Layers

Typically, DETR-like object detectors consist of six layers each in their transformer encoders and decoders. As mentioned in Sec. B and Sec. D, our approach for mask prediction is based on Deformable-DETR++ [12], utilizing solely the last decoder layer. All six decoder layers are employed for object detection tasks. Therefore, we only evaluate the box prediction for all layers as shown in Tab. 4. Evaluation results suggest that our method can effectively improve the performance of the primary route across all six decoder layers, demonstrating the efficacy of our approach. Moreover, the one-to-many prediction training routes, namely, Route-1 and Route-3, significantly surpass the primary route in the shallower layers. For example, with a model trained on a 12-epoch schedule, Route-3 achieves a 6.4% improvement over the primary route in layer 0, and a 0.4% improvement in layer 5. These experiments indicate that the primary route needs more decoder layers to reach comparable performance as the auxiliary routes equipped with NMS. For instance, Route-1 and Route-3 can reach 49.4% mAP in layer 2, while the primary route achieves 49.4% mAP in layer 4.

## H. Qualitative Results

We present the prediction results of our method in Fig. 4. The model is based on the DINO [10] with IA-BCE loss [1] as the baseline using Swin-L [7] backbone.

## References


[1] Zhi Cai, Songtao Liu, Guodong Wang, Zheng Ge, Xiangyu Zhang, and Di Huang. Align-detr: Improving detr with simple iou-aware bce loss. In *Brit. Mach. Vis. Conf.*, 2024. 4

[2] Qiang Chen, Xiaokang Chen, Jian Wang, Shan Zhang, Kun Yao, Haocheng Feng, Junyu Han, Errui Ding, Gang Zeng, and Jingdong Wang. Group detr: Fast detr training with group-wise one-to-many assignment. In *Int. Conf. Comput. Vis.*, 2023. 4

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2016. 4

[4] Zhengdong Hu, Yifan Sun, Jingdong Wang, and Yi Yang. Dac-detr: Divide the attention layers and conquer. *Adv. Neural Inform. Process. Syst.*, 2024. 2, 3, 4

[5] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 2, 4

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, 2014. 3

[7] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, 2021. 4

[8] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv preprint arXiv:2212.06137*, 2022. 3

[9] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Int. Conf. Comput. Vis.*, 2019. 3, 4

[10] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. In *Int. Conf. Learn. Represent.*, 2023. 2, 4

[11] Chuyang Zhao, Yifan Sun, Wenhao Wang, Qiang Chen, Errui Ding, Yi Yang, and Jingdong Wang. Ms-detr: Efficient detr training with mixed supervision. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2024. 3, 4

[12] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *Int. Conf. Learn. Represent.*, 2021. 1, 2, 3, 4, 5

| Method | Layer | Route | Epoch | NMS | mAP | $AP_{50}$ | $AP_{75}$ | $AP_s$ | $AP_m$ | $AP_l$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Baseline [12] | 0 | - | 12 | | 38.5 | 53.4 | 42.2 | 23.1 | 41.8 | 50.0 |
| w/ Mr.DETR | 0 | Route-2 | | | 41.0 (+2.5) | 55.5 | 45.0 | 24.8 | 43.8 | 54.3 |
| w/ Mr.DETR | 0 | Route-1 | | ✔ | **47.5** | 64.9 | 52.2 | 29.0 | 51.0 | 62.1 |
| w/ Mr.DETR | 0 | Route-3 | | ✔ | 47.4 | 64.9 | 52.3 | 29.1 | 50.9 | 62.4 |
| Baseline [12] | 1 | - | 12 | | 42.6 | 58.7 | 46.5 | 25.7 | 46.2 | 55.1 |
| w/ Mr.DETR | 1 | Route-2 | | | 45.8 (+3.2) | 61.8 | 50.3 | 27.6 | 49.0 | 60.3 |
| w/ Mr.DETR | 1 | Route-1 | | ✔ | 48.9 | 66.1 | 53.5 | 30.4 | 52.4 | 63.8 |
| w/ Mr.DETR | 1 | Route-3 | | ✔ | **49.0** | 66.3 | 53.6 | 30.4 | 52.6 | 64.2 |
| Baseline [12] | 2 | - | 12 | | 45.0 | 62.0 | 49.2 | 27.7 | 48.7 | 58.6 |
| w/ Mr.DETR | 2 | Route-2 | | | 48.0 (+3.0) | 64.7 | 52.5 | 29.3 | 51.3 | 62.4 |
| w/ Mr.DETR | 2 | Route-1 | | ✔ | **49.4** | 66.7 | 54.2 | 31.1 | 52.8 | 64.3 |
| w/ Mr.DETR | 2 | Route-3 | | ✔ | **49.4** | 66.5 | 54.1 | 31.0 | 52.8 | 64.3 |
| Baseline [12] | 3 | - | 12 | | 46.1 | 63.5 | 50.4 | 28.6 | 49.6 | 60.1 |
| w/ Mr.DETR | 3 | Route-2 | | | 49.0 (+2.9) | 66.0 | 53.5 | 30.4 | 52.3 | 63.7 |
| w/ Mr.DETR | 3 | Route-1 | | ✔ | 49.7 | 67.1 | 54.5 | 31.3 | 53.0 | 64.8 |
| w/ Mr.DETR | 3 | Route-3 | | ✔ | **49.9** | 67.2 | 54.6 | 31.4 | 53.3 | 64.8 |
| Baseline [12] | 4 | - | 12 | | 46.4 | 64.1 | 50.7 | 29.0 | 50.0 | 60.7 |
| w/ Mr.DETR | 4 | Route-2 | | | 49.4 (+3.0) | 66.5 | 54.0 | 30.4 | 52.6 | 64.3 |
| w/ Mr.DETR | 4 | Route-1 | | ✔ | 49.9 | 67.4 | 54.7 | 31.1 | 53.2 | 65.3 |
| w/ Mr.DETR | 4 | Route-3 | | ✔ | **50.0** | 67.5 | 54.7 | 31.3 | 53.3 | 65.7 |
| Baseline [12] | 5 | - | 12 | | 46.5 | 64.2 | 50.8 | 28.8 | 50.0 | 60.7 |
| w/ Mr.DETR | 5 | Route-2 | | | 49.5 (+3.0) | 66.6 | 54.1 | 30.3 | 52.6 | 64.7 |
| w/ Mr.DETR | 5 | Route-1 | | ✔ | 49.8 | 67.3 | 54.6 | 31.0 | 53.0 | 65.3 |
| w/ Mr.DETR | 5 | Route-3 | | ✔ | **49.9** | 67.3 | 54.6 | 30.8 | 53.2 | 65.4 |
| Baseline [12] | 0 | - | 24 | | 41.3 | 56.3 | 45.4 | 25.5 | 44.6 | 53.2 |
| w/ Mr.DETR | 0 | Route-2 | | | 42.6 (+1.3) | 57.3 | 46.4 | 25.8 | 44.8 | 55.9 |
| w/ Mr.DETR | 0 | Route-1 | | ✔ | **48.2** | 65.8 | 52.7 | 29.1 | 51.4 | 62.5 |
| w/ Mr.DETR | 0 | Route-3 | | ✔ | **48.2** | 66.0 | 52.8 | 29.2 | 51.6 | 62.2 |
| Baseline [12] | 1 | - | 24 | | 45.1 | 61.5 | 49.5 | 28.2 | 48.3 | 58.1 |
| w/ Mr.DETR | 1 | Route-2 | | | 47.1 (+2.0) | 63.3 | 51.3 | 28.4 | 49.9 | 61.2 |
| w/ Mr.DETR | 1 | Route-1 | | ✔ | 49.6 | 67.1 | 54.2 | 30.5 | 52.9 | 63.9 |
| w/ Mr.DETR | 1 | Route-3 | | ✔ | **49.7** | 67.3 | 54.2 | 30.6 | 53.0 | 64.1 |
| Baseline [12] | 2 | - | 24 | | 47.2 | 64.3 | 51.8 | 29.8 | 50.3 | 60.8 |
| w/ Mr.DETR | 2 | Route-2 | | | 49.2 (+2.0) | 66.2 | 53.6 | 30.2 | 52.2 | 63.4 |
| w/ Mr.DETR | 2 | Route-1 | | ✔ | 50.1 | 67.7 | 54.8 | 31.1 | 53.3 | 64.8 |
| w/ Mr.DETR | 2 | Route-3 | | ✔ | **50.2** | 67.9 | 54.7 | 31.4 | 53.5 | 64.7 |
| Baseline [12] | 3 | - | 24 | | 48.1 | 65.7 | 52.7 | 30.8 | 51.3 | 61.8 |
| w/ Mr.DETR | 3 | Route-2 | | | 50.0 (+1.9) | 67.4 | 54.4 | 30.9 | 53.0 | 64.5 |
| w/ Mr.DETR | 3 | Route-1 | | ✔ | 50.2 | 68.0 | 54.8 | 31.7 | 53.4 | 64.8 |
| w/ Mr.DETR | 3 | Route-3 | | ✔ | **50.3** | 68.1 | 54.8 | 31.5 | 53.4 | 65.0 |
| Baseline [12] | 4 | - | 24 | | 48.6 | 66.3 | 53.1 | 30.9 | 51.8 | 62.3 |
| w/ Mr.DETR | 4 | Route-2 | | | 50.3 (+1.7) | 67.9 | 54.7 | 31.3 | 53.2 | 65.0 |
| w/ Mr.DETR | 4 | Route-1 | | ✔ | **50.5** | 68.3 | 55.1 | 31.7 | 53.7 | 65.1 |
| w/ Mr.DETR | 4 | Route-3 | | ✔ | 50.4 | 68.3 | 54.9 | 32.0 | 53.5 | 65.3 |
| Baseline [12] | 5 | - | 24 | | 48.6 | 66.4 | 53.1 | 30.8 | 51.9 | 62.9 |
| w/ Mr.DETR | 5 | Route-2 | | | 50.3 (+1.7) | 68.0 | 54.7 | 31.5 | 53.2 | 65.0 |
| w/ Mr.DETR | 5 | Route-1 | | ✔ | **50.4** | 68.2 | 55.1 | 31.5 | 53.5 | 65.2 |
| w/ Mr.DETR | 5 | Route-3 | | ✔ | **50.4** | 68.2 | 54.8 | 31.7 | 53.4 | 65.2 |

Table 4. **Evaluation results of box prediction in all six decoder layers.** 'Route-2': the primary route for one-to-one prediction. 'Route-1': the auxiliary route with an independent FFN. 'Route-3': the auxiliary route with an instructive self-attention.

Figure 4. **Qualitative results of our method.** Left: prediction results. Right: ground truth.