# MI-DETR: An Object Detection Model with Multi-time Inquiries Mechanism

## Supplementary Material

Due to the space limitation of the main text, we provide more results and discussions in the supplementary material, which are organized as follows:

- Section 7: More Diagnostic Experiments.
  – Section 7.1: The effectiveness of our method on another DETR-like model.
  – Section 7.2: The comparison between our proposed **MI-DETR** and *Lite* **MI-DETR**.
  – Section 7.3: Diagnostic experiments on different query fusion mechanisms.
- Section 8: Further Analysis of Model Complexity.
- Section 9: More Visualization Analysis.

## 7. More Diagnostic Experiments

### 7.1. The effectiveness of our method on another DETR-like model.

We have conducted experiments to verify the effects of ***MI*** on other models in Sec. 4.3.2, including the most representative model DINO [44] and SOTA model Relation-DETR [12]. Due to the space limitation of the main text, we present additional experiments based on recent proposed Align-DETR [2] to further validate the effectiveness and generalization of ***MI***. The results are available in Tab. 7, from which we can observe that our method show consistent improvement on Align-DETR.

| Method | Backbone | Epochs | AP |
|---|---|---|---|
| Align-DETR [2] | ResNet-50 | 12 | 50.2 |
| ours | ResNet-50 | 12 | **51.5(+1.3)** |
| Align-DETR [2] | ResNet-50 | 24 | 51.3 |
| ours | ResNet-50 | 24 | **51.8(+0.5)** |

Table 7. Effectiveness of our method on Align-DETR.

| Method | AP | $AP_{50}$ | $AP_{75}$ | GFLOPS | Params |
|---|---|---|---|---|---|
| **MI-DETR** | 50.2 | 68.1 | 54.8 | 311 | 76M |
| *Lite* **MI-DETR** | 50.1 | 67.9 | 54.7 | 299 | 72M |

Table 8. The comparison between our proposed **MI-DETR** and *Lite* **MI-DETR**.

### 7.2. The comparison between our proposed MI-DETR and *Lite* MI-DETR.

**MI-DETR** and *Lite* **MI-DETR** are two kinds of architectures of our model, and the latter targets to reduce the model complexity by sharing the self-attention layer in ***MI*** decoder layers. As shown in Tab. 8, *Lite* **MI-DETR** achieves comparable performance with **MI-DETR** while requiring less parameters and computation.

### 7.3. Diagnostic experiments on different query fusion mechanisms.

To verify the impact of different query fusion mechanisms on model performance, we conduct experiments with three different fusion mechanisms, including "add", "linear+concat", and "concat+linear". "add" fusion directly adds up object queries from different inquiry heads. "linear+concat" fusion first projects the object queries to $C/M$ dimensions along the feature dimension, and then concatenates them, where $C$ is the original dimensions of object queries and $M$ is the number of inquiry heads. "concat+linear" fusion is the classic concatenation fusion as illustrated in Eq. (4). The results are reported in Tab. 9, from which we can observe that the fusion mechanism has slight impact on the performance, which potentially proves that ***MI*** is the main contributor to performance improvement.

| query fusion | AP | $AP_{50}$ | $AP_{75}$ | $AP_S$ | $AP_M$ | $AP_L$ |
|---|---|---|---|---|---|---|
| add | 50.1 | 67.9 | 54.7 | 32.7 | 53.2 | 64.9 |
| linear+concat | 49.8 | 67.7 | 54.2 | 32.7 | 52.9 | 64.8 |
| concat+linear | **50.2** | **68.1** | **54.8** | **33.4** | **53.6** | **64.5** |

Table 9. Diagnostic experiments on different query fusion mechanisms in ***MI*** decoder layer. "add", "linear+concat", and "concat+linear" represent three different fusion mechanisms.

## 8. Further Analysis of Model Complexity

We have conducted experiments in Sec. 4.4 to eliminate the misunderstanding that our performance improvement might result from increasing the parameters complexity. To further analyze the complexity of our method, we conduct additional experiments by comparing the training time, inference speed, and AP of basically-equal-parameter models (with different layer number, head number, and channel number configurations), and the results are summarized in Tab. 10. Specifically, two kinds of adding-parameters strategies, vertically deepening 2x/4x layers (#2/#4) and horizontally widening 2x/4x heads and channels (#6/#8), generate four models with the single decoder. We note all models in Tab. 10 use the same number of queries as the DINO baseline. All these experiments consistently prove the superiority of our parallel multi-time inquires architecture (#3, #5, #7, and #9) on performance and complexity.

| ID | Strategy | LN | HN | CN | IHN | Params | Train ↓ | Test ↑ | AP ↑ |
|---|---|---|---|---|---|---|---|---|---|
| #1 | baseline | 6 | 8 | 256 | 1 | 47M | 70min | 14.2 | 49.0 |
| #2 | vertical | 12 | 8 | 256 | 1 | 58M | 88min | 12.2 | 49.0 |
| #3 | | 6 | 8 | 256 | 2 | 57M | 78min | 12.8 | **49.5** |
| #4 | | 24 | 8 | 256 | 1 | 78M | 124min | 9.2 | 47.3 |
| #5 | | 6 | 8 | 256 | 4 | 75M | 95min | 11.3 | **49.8** |
| #6 | horizontal | 6 | 16 | 512 | 1 | 58M | 83min | 13.3 | 49.0 |
| #7 | | 6 | 8 | 256 | 2 | 57M | 78min | 12.8 | **49.5** |
| #8 | | 6 | 32 | 1024 | 1 | 97M | 103min | 10.6 | 49.1 |
| #9 | | 6 | 8 | 256 | 4 | 75M | 95min | 11.3 | **49.8** |

Table 10. Comparisons on performance and complexity under diverse conditions. LN: Layer Number; HN and CN: Head Number and Channel Number of the multi-head self-attention and multi-head cross-attention; IHN: Inquiry Head Number. "Train" denotes the average training time per epoch, and "Test" indicates the inference FPS (*i.e.*, Frames Per Second) tested on the same GPUs.

## 9. More Visualization Analysis

We have conducted rich visualization experiments in Sec. 4.3.5, including object queries visualization and object detection results visualization. To avoid the randomness of visualization results, we present more visualization results, as shown in Fig. 7, Fig. 8, and Fig. 9. Fig. 7 illustrates that object queries in different inquiry heads generally present distinct distributions, validating that multi-pattern information are learnt in different inquiry heads. Fig. 8 presents that different inquiry heads are mutually collaborate. For example, on the first example, the "car" is not detected in head 1, 2, and 3, and the "person" on road is not detected in head 1, 2, 4. These two objects are respectively noticed in head 4 and head 3 and are finally detected by heads 1-4. As shown in Fig. 9, our method exhibits advantages in challenging natural scenes (*e.g.*, extremely-small, heavily-occluded, and confusingly mixed with the background).

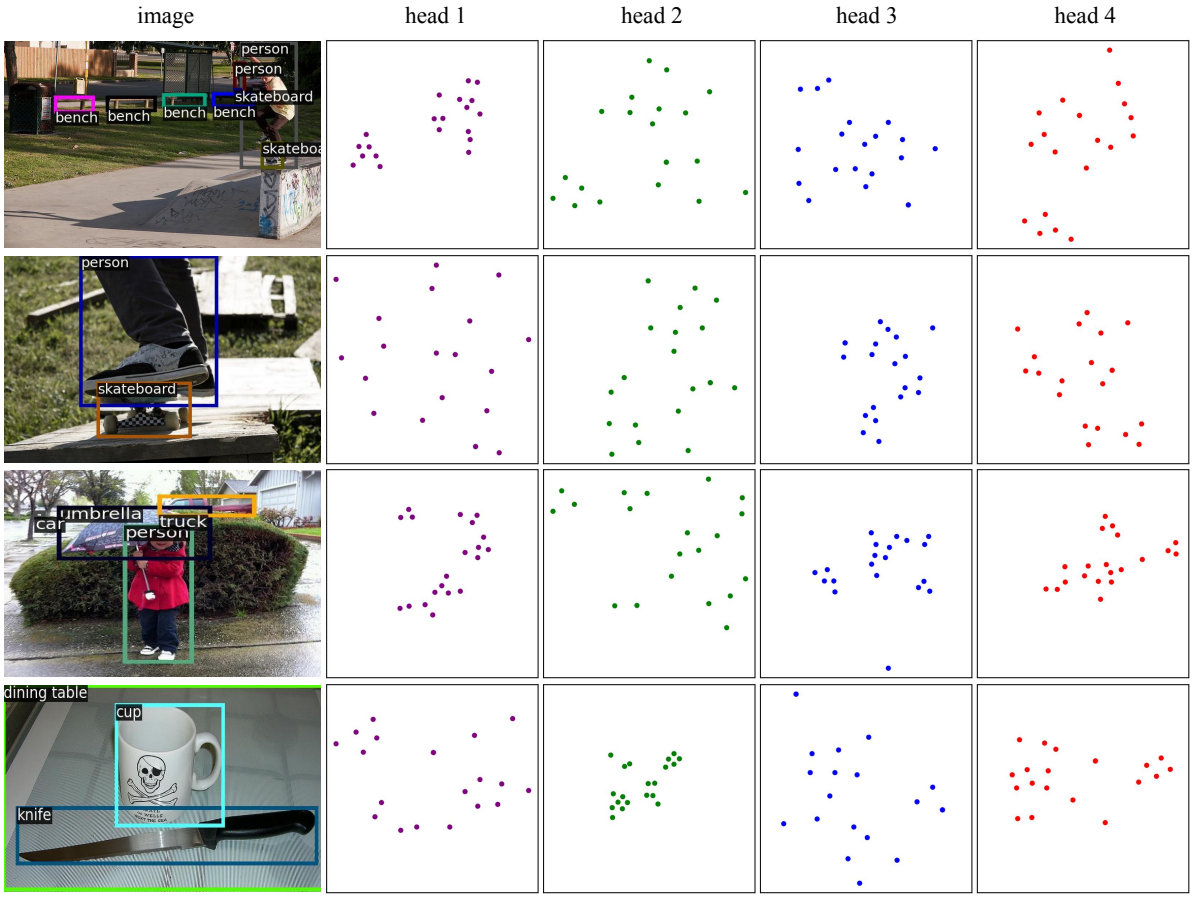


Figure 7. The visualization of object queries in different inquiry heads by T-SNE high-dimensional data visualization tool. This is a supplement to Fig. 4 of the main text.

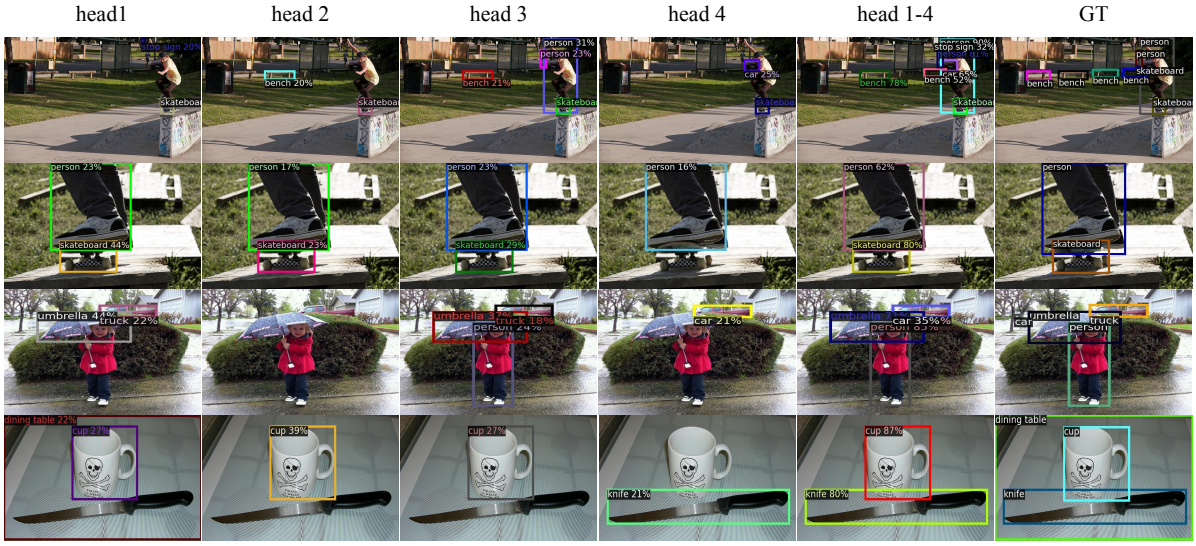


Figure 8. More object detection results based on the single inquiry head and multiple inquiry heads. This is a supplement to Fig. 5 of the main text.

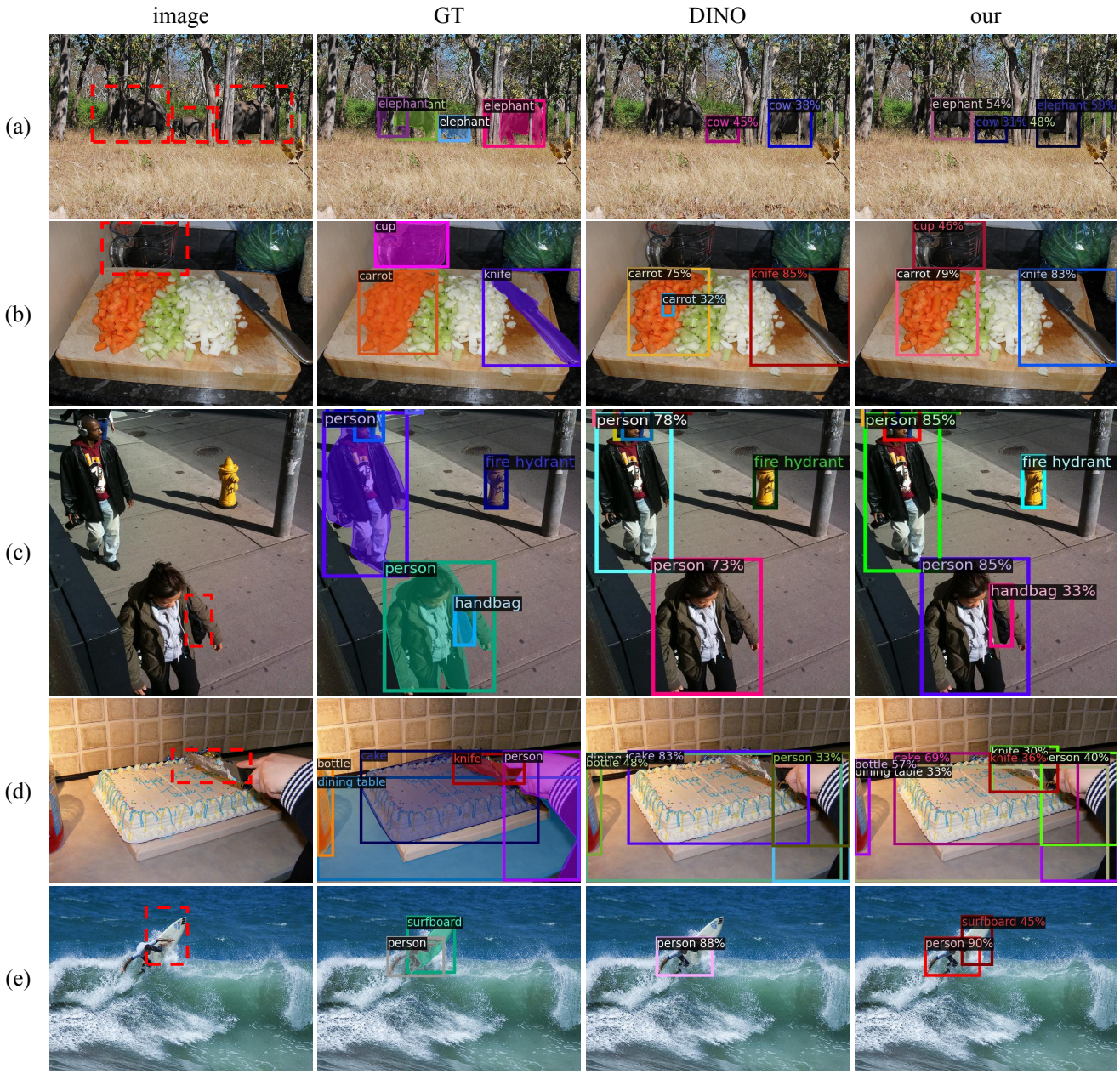


Figure 9. The "elephants" in (a), the "cup" in (b), and the "surfboard" in (e) are confusingly mixed with the background. The "elephants" in (a) and the "handbag" in (c) are heavily-occluded. The "handbag" in (c) and the "knife" in (d) are small. These objects are difficult to detect, and DINO fails to detect them (*e.g.*, the elephants are falsely detected as cows or missed). In contrast, our method successfully detect these challenging objects. Suggest zooming in to view clearer details. This is a supplement to Fig. 6 of the main text.