Dynamic-DINO: Fine-Grained Mixture of Experts Tuning for Real-time Open-Vocabulary Object Detection

Supplementary Material

A. Appendix

A.1. Datasets Details

Tab. 7 presents the dataset specifications utilized for pre-training Dynamic-DINO, including the Objects365 (V1) [35], GQA [16], Flickr30k [28], and V3Det [41] datasets, where Texts denotes the number of categories for the detection dataset and the number of phrases for the grounding dataset, Images denotes the number of images and Annotation denotes the number of instance annotations. The total number of samples in our pre-training dataset is 1.56M.

Table 7. **Pre-Training Data.**

Dataset	Type	Texts	Images	Annotation
O365 [35]	Detection	365	609K	9621K
V3Det [41]	Detection	13K	184K	1233K
GQA [16]	Grounding	387K	621K	3681K
Flickr30k [28]	Grounding	94K	149K	641K

A.2. Core Codes

The core implementation of our MoE-Tuning is detailed in Algorithm 1, encompassing expert initialization and router initialization. Following MoE [8] paradigm, we scale up the model by expanding the FFN in each layer of the decoder into N FFNs of identical size. For each FFN, its intermediate hidden dimension is evenly divided into k partitions, thereby constructing $k \times N$ experts. In addition, we initialize the experts by assigning the pre-trained FFN weights from the base model to each expert. For router initialization, we first randomly initialize the weights $W_r' \in \mathbb{R}^{N \times D}$, and then replicate each centroid vector in W_r' k times to form the router weights $W_r \in \mathbb{R}^{kN \times D}$. With this initialization, the router is guaranteed to select the k experts derived from the same FFN at the start of fine-tuning, ensuring incremental performance improvements during MoE-Tuning.

A.3. More Experiments

Ablation Study on Parameter Numbers. Our method can flexibly adjust total parameters while keeping activated parameters unchanged. As shown in Table 8, even +6M parameters bring +0.73 AP on average, with scaling parameters yielding greater improvements.

Ablation Study on MoE Deployment. As shown in Table 9, extending MoE layers to FFN in image encoder, the performance further increases by +0.5 AP on average.

Algorithm 1 MoE Initialization

```
"""
Input:
n: int
k: int
ffn: nn.Module
"""
embed_dim = ffn.embed_dim
ffd_dim = ffn.ffd_dim // k

ffns = [
    FFN(embed_dim, ffd_dim)
    for _ in range(k)
]
for i in range(k):
    ffns[i].w1
      =ffn.w1[i*ffd_dim:(i+1)*ffd_dim,:]
    ffns[i].b1
      =ffn.b1[i*ffd_dim:(i+1)*ffd_dim]
    ffns[i].w2
      =ffn.w2[:,i*ffd_dim:(i+1)*ffd_dim]
    ffns[i].b2 = ffn.b2 / k

self.experts = nn.ModuleList([])
for i in range(n):
    for j in range(k):
        self.experts.append(
            copy.deepcopy(ffns[j])
        )

w_gate = torch.randn(n, 1, embed_dim)
w_gate = w_gate.repeat(1, k, 1)
w_gate = w_gate.reshape(n*k, embed_dim)
self.router = nn.Parameter(
    w_gate, requires_grad=True)
```

Ablation Study on Model Initialization. We validate the effectiveness of our initialization modification. As shown in Table 10, it boosts the accuracy ceiling.

Results on RefCOCO. Experiments on RefCOCO, RefCOCO+ and RefCOCOg are added in Table 11. Results show that our method still works on zero-shot REC tasks.

Performance Comparisons on Edge Devices. We evaluate the pre-trained model on Jetson Orin NX SUPER 8GB. As shown in Table 12, our method introduces only +0.24M

Table 8. Comparison of the parameter numbers. All models are trained on O365, GoldG, and V3Det. Image resolution is 640 × 640. "Parameters" represents active parameters / total parameters. Dynamic-DINO×N-Top2 indicates a model with N experts, where 2 experts are activated per inference.

Method	Parameters	COCO-val	LVIS-minival	LVIS-val
G-DINO 1.5 Edge	178M/178M	42.6	31.1	25.4
Dynamic-DINO×4-Top2	178M/184M	43.2(+0.6)	31.6(+0.5)	26.5(+1.1)
Dynamic-DINO×8-Top2	178M/197M	43.4(+0.8)	32.4(+1.3)	26.9(+1.5)
Dynamic-DINO×16-Top2	178M/222M	43.7(+1.1)	33.6(+2.5)	27.4(+2.0)

Table 9. Ablation study of MoE deployment across model parts. Dynamic-DINO×16-Top2 is utilized. All models are trained on O365, GoldG, and V3Det. Image resolution is 800 × 1333.

Decoder	Image Encoder	COCO-val	LVIS-minival	LVIS-val
✗	✗	42.6	31.1	25.4
✓	✗	43.7(+1.1)	33.6(+2.5)	27.4(+2.0)
✓	✓	**44.5**(+1.9)	**33.7**(+2.6)	**28.0**(+2.6)

Table 10. Ablation study for the initialization. Dynamic-DINO×16-Top2 is utilized. All models are trained on O365, GoldG, and V3Det. Image resolution is 640 × 640.

Method	COCO-val	LVIS-minival	LVIS-val
G-DINO 1.5 Edge	42.6	31.1	25.4
Dynamic-DINO w/o Initialization	43.1	32.5	26.2
Dynamic-DINO w/ Initialization	**43.7**	**33.6**	**27.4**

Table 11. Comparison of zero-shot performance on RefCOCO, RefCOCO+ and RefCOCOg. All models are trained on O365, GoldG, and V3Det. Image resolution is 640 × 640.

Method	RefCOCO			RefCOCO+			RefCOCOg	
	val	testA	testB	val	testA	testB	val	test
G-DINO 1.5 Edge	43.8	49.9	39.5	43.3	47.9	40.2	51.2	52.8
Dynamic-DINO (Ours)	**47.9**	**53.9**	**42.2**	**47.4**	**52.0**	**42.3**	**56.6**	**56.5**

FLOPs and -0.8 FPS over the baseline while achieving +1.87 AP on average.

Table 12. Performance comparisons on NVIDIA Orin NX. All models are trained on O365, GoldG, and V3Det. Image resolution is 640 × 640. Dynamic-DINO×16-Top2 is utilized. FLOPs are measured solely for the Decoder, which contains the MoE Layers in our method. FPS evaluates the full feed-forward pass.

Method	COCO-val	LVIS-minival	LVIS-val	FLOPs	FPS
G-DINO 1.5 Edge	42.6	31.1	25.4	2679.51M	10.2
Dynamic-DINO (Ours)	43.7	33.6	27.4	2679.75M	9.4

A.4. Visualizations

Fig. 12 provides a comparative visualization of the model's zero-shot object detection performance before and after the implementation of MoE-Tuning. The results demonstrate a significant improvement in the model's sensitivity to both object quantity and small-scale targets. Fig. 13 further visualizes the improvement in the model's ability to detect rare classes, indicating that MoE-Tuning effectively alleviates the long-tail problem.

A.5. More Statistical Analysis

Fig. 14 provides a detailed visualization of the expert collaboration statistics across each MoE layer of Dynamic-DINO, evaluated on the COCO, LVIS-minival, and ODinW13. The results reveal that Dynamic-DINO exhibits a nearly consistent pattern of expert collaboration across diverse datasets, which underscores the stability of expert collaboration and the sufficiency of training.

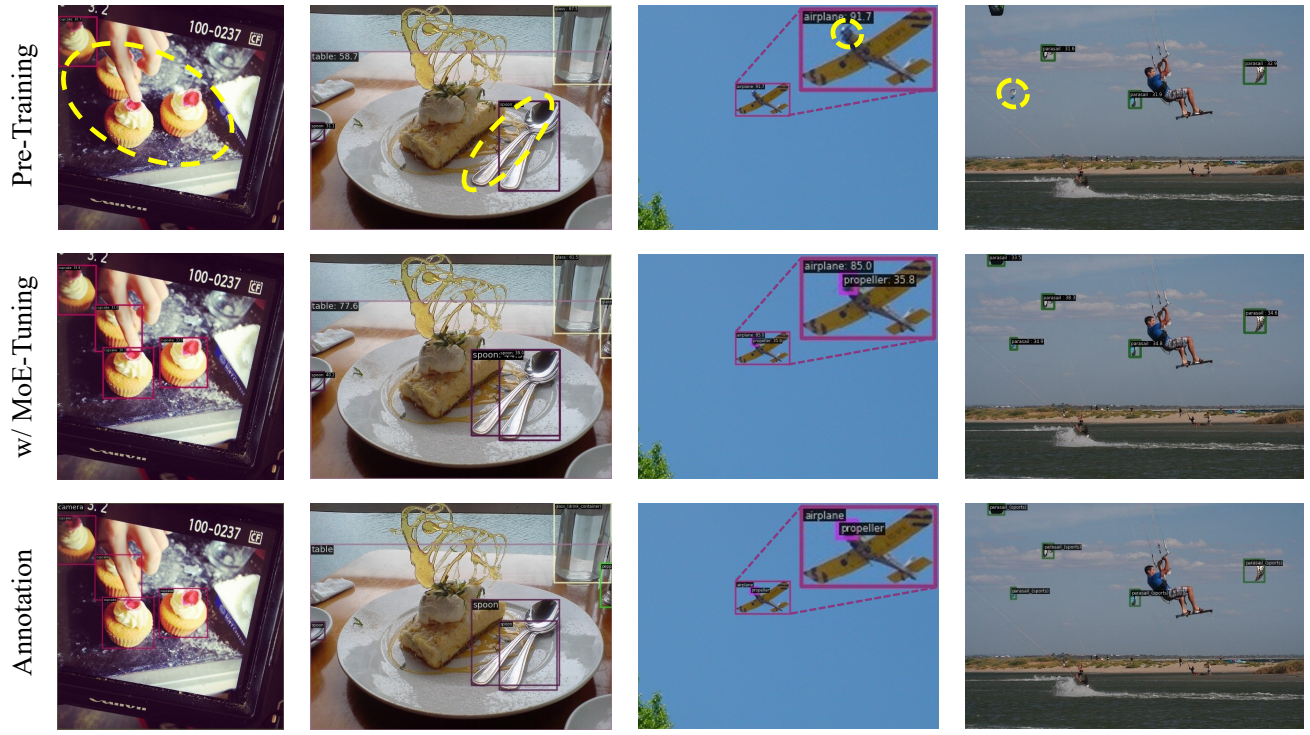

Figure 12. **Comparison of visualization results for zero-shot inference on LVIS.** We visualize the predictions of our pre-trained base model and Dynamic-DINO after MoE-Tuning. The failures are highlighted with a yellow circle.

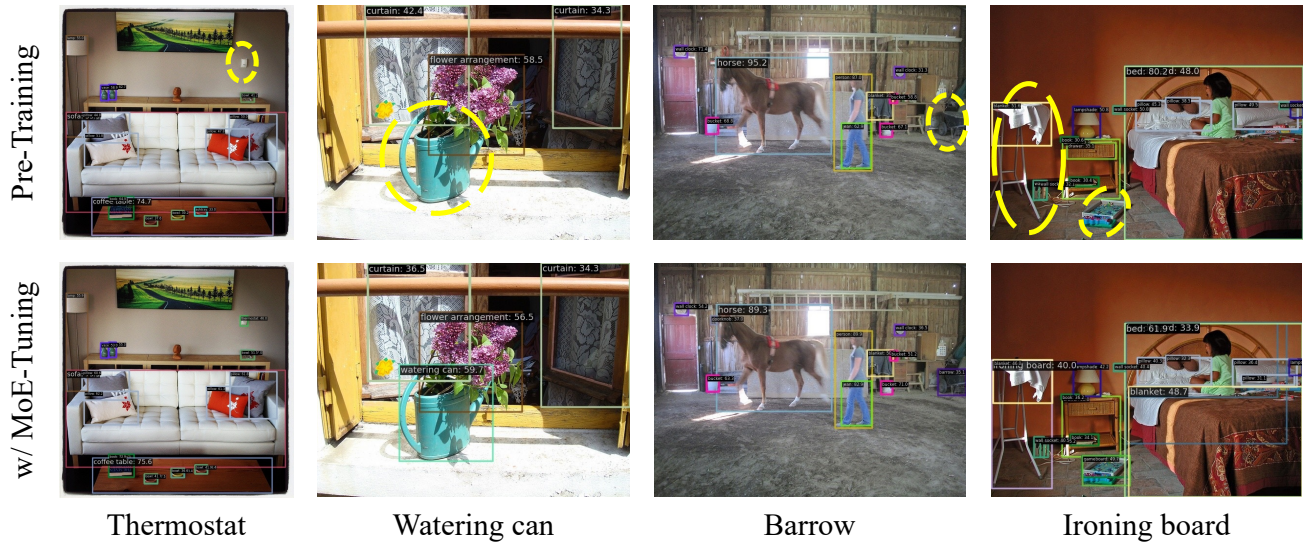

Figure 13. **Comparison of visualization results for zero-shot inference on rare classes of LVIS.** We visualize the predictions of our pre-trained base model and Dynamic-DINO after MoE-Tuning. The failures are highlighted with a yellow circle.

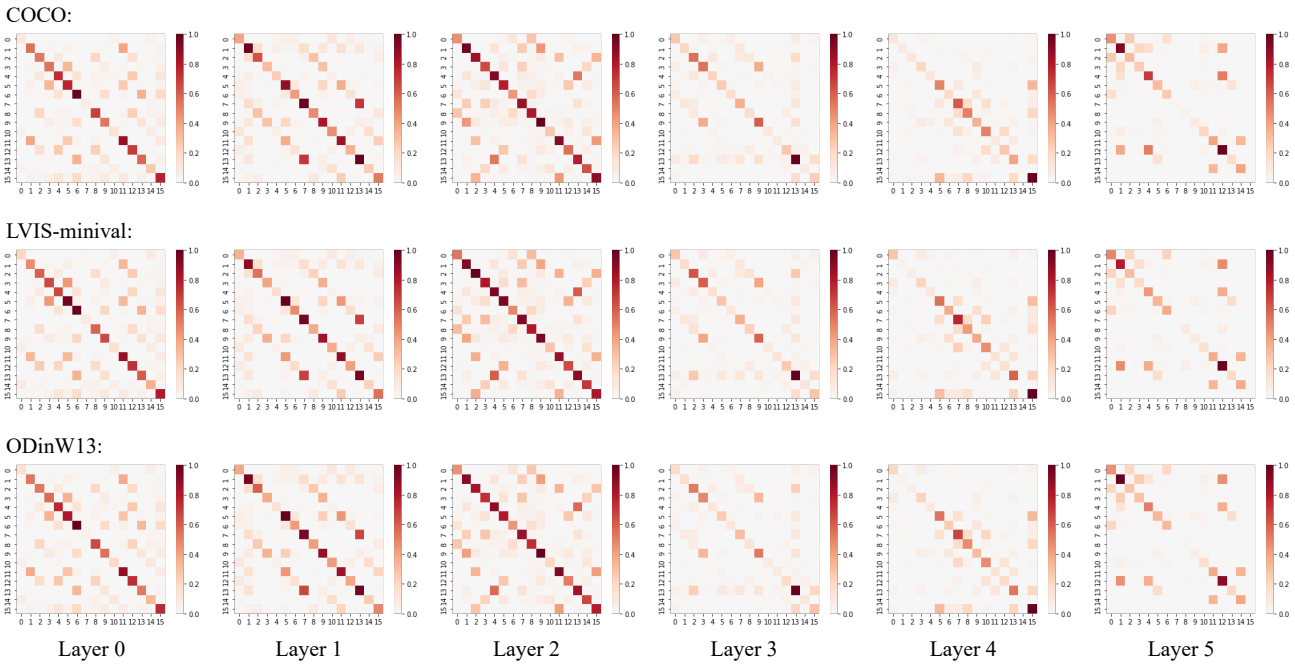

Figure 14. **Expert collaboration across 3 datasets.** The normalized co-selection frequencies are quantified for all expert pairs with Dynamic-DINO×16-Top2 model, which comprises 16 experts and activates 2 experts per inference.