Technical Appendix

Towards Efficient Use of Multi-Scale Features in Transformer-Based Object Detectors

Gongjie Zhang[†1,2] Zhipeng Luo[†1,3] Zichen Tian[1] Jingyi Zhang[1] Xiaoqin Zhang[4] Shijian Lu[*1]

[1]S-Lab, Nanyang Technological University [2]Black Sesame Technologies [3]SenseTime Research [4]Wenzhou University

gjz@ieee.org zhipeng001@e.ntu.edu.sg shijian.lu@ntu.edu.sg

A. Technical Appendix

This section provides more details of our proposed method and its experimental results, which are omitted in the main paper due to space limitation.

A.1. Training Objective of Iterative Multi-scale Feature Aggregation (IMFA)

As described in Section 4, all additional operations introduced by IMFA are fully differentiable, including the selection of top-K prior detection predictions, sparse feature sampling via bilinear interpolation, adaptive scale selection, Dynamic FFN, and iterative feature aggregation. Thus, the proposed IMFA can be trained end-to-end on top of the corresponding baselines [2, 7, 10, 12].

Besides, IMFA requires no additional training objectives. In other words, IMFA is trained purely with the supervision signals of the corresponding baselines' detection-related losses.

A.2. Additional Experiment Results

Table 3 in our manuscript has already demonstrated that our proposed IMFA can work well with stronger vision Transformer (ViT) backbones [8]. Here we present more results in Table 9. With Swin-Transformer-Tiny (Swin-T) [8] as the backbone, `DAB-DETR-Swin-T+IMFA` significantly outperforms `DAB-DETR-R50+IMFA` with comparable computational cost, which further demonstrates IMFA's excellent scalability.

Method	MS	SMS	#Epochs	#Params	FLOPs	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
DAB-DETR-R50 [7] + IMFA (Ours)		✓	50	53M	108G	45.5	65.0	49.3	27.3	48.3	61.6
DAB-DETR-Swin-T [7] + IMFA (Ours)		✓	50	57M	114G	**47.0**	**67.1**	**50.6**	**29.5**	**49.7**	**63.3**
Deformable-DETR-Swin-T [13]	✓		50	40M	180G	45.7	65.3	49.9	26.9	49.4	61.2
YOLOS-DeiT-B [3]			150	127M	538G	42.0	62.2	44.5	19.5	45.3	62.1
ViDT-Swin-T [11]	✓		50	38M	100G	44.8	64.5	48.7	25.9	47.6	62.1

Table 9. Comparison with state-of-the-art object detectors with ViT backbones on COCO val 2017. 'MS' denotes the use of multi-scale features. 'SMS' denotes the use of sparse multi-scale features with our proposed IMFA. '§' denotes two-stage Transformer-based object detector, with the encoder producing 'region proposals' to initialize object queries.

Besides, we also compare our method with other state-of-the-art Transformer-based object detectors using vision Transformers as backbones. As shown in Table 9, our `DAB-DETR-Swin-T+IMFA` still achieves the best overall performance. We notice that ViDT [11] has a lower FLOPs than ours, because it adopts an 'encoder-free neck architecture' based on deformable attention [13].

[*] marks corresponding author. [†] marks equal technical contribution.

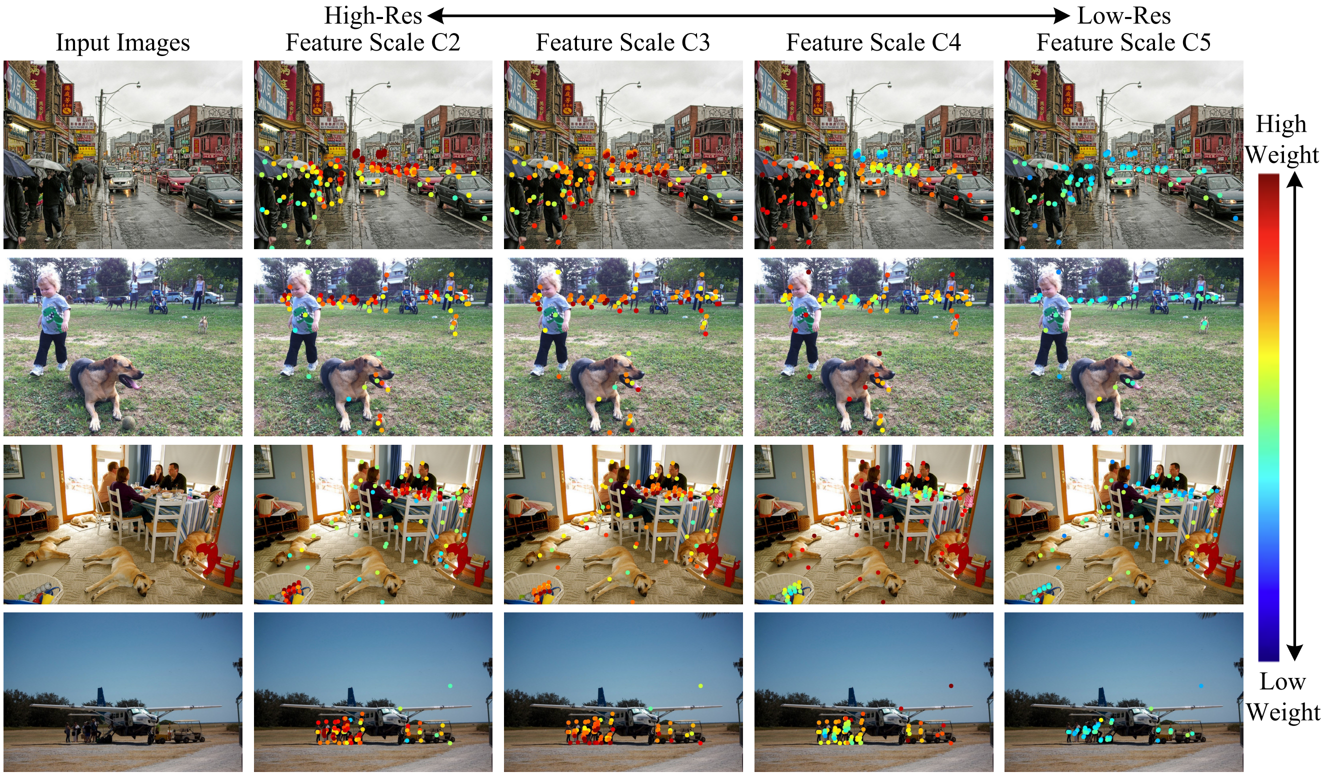

Figure 5. Additional visualization of IMFA's sampling locations and their adaptively selected feature scales. The searched sampling points mostly fall around the objects of interest, many of which are highly representative points with rich semantics, such as objects' extremities. Besides, IMFA adaptively selects appropriate feature scales for each sampling point, generating sparse yet informative scale-adaptive features for refined detection predictions. Best viewed in color.

A.3. Additional Visualization Results

For a more comprehensive understanding of the proposed IMFA, we provide more visualization results on IMFA's sampling locations and IMFA's adaptively selected feature scales in Fig. 5. These visualizations validate the effectiveness of IMFA in searching informative locations and appropriate scales for multi-scale feature sampling, even under very complex scenarios as shown in the first row. It is noteworthy that the sampling weights for C5 are generally low, even for large objects. This is because C5 has the same feature scale as the encoded image features, and thus IMFA tends to sample multi-scale features from C2-C4 for additional information.

A.4. Implementation Details for Human Pose Estimation

Section 5.4 investigates the generality of IMFA across various tasks by integrating it with PRTR (two-stage variant) [5]. Here we present the implementation details of this integration.

The implementation details align with PRTR (two-stage variant) [5]'s implementation. Concretely, we adopt the person detection results fine-tuned on COCO [6] to extract image patches that contain persons. These image patches are resized into a fixed shape of 384x288, then processed by data augmentations including random rotation, random scale, and horizontal flipping, and finally fed into the `PRTR+IMFA` model. We adopt the AdamW [9] optimizer for training, with the base learning rate for the ResNet backbone [4] as 1e-5 and 1e-4 for the rest, with a weight decay of 1e-4. The total number of training epochs is 200, and the learning rate is halved at the 120th and 140th epoch, respectively. For the Transformer part, the number of encoder and decoder layers are both set to 6. The number of keypoint queries is set to 100. During inference, we adopt the common practice of flip-test [5] and compute the keypoint coordinates by averaging the outputs of the original and flipped person image patches.

A.5. Further Discussions

Our differences with multi-scale feature fusion. Compared with existing multi-scale methods (e.g., FPN, DLA, Amulet, Deformable DETR, SMCA-DETR, etc.), the way we utilize multi-scale features is significantly different. Specifically, most existing methods use all the feature tokens from multi-scale features (typically 20x∼80x feature tokens compared to single-scale), whereas IMFA only adds less than 1x multi-scale feature tokens by aggregating multi-scale features from just a few informative keypoints. This is the key reason that IMFA can serve as a generic paradigm for efficient exploitation of multi-scale features in Transformer-based detectors. Our experiments show that, at very slight computational costs, IMFA is able to boost detection performance by large margins for multiple Transformer-based detectors.

It is noteworthy that Deformable DETR [13] also adopts sparse multi-scale feature computation. However, Deformable DETR still stores and uses all multi-scale feature tokens, which is different from IMFA. IMFA does not need to compute and store dense and high-resolution multi-scale features and is more efficient. Thus, IMFA introduces significantly smaller computational costs in processing multi-scale features.

Our relation to guided refinement. Guided refinement typically refers to the recursive update of predictions based on previous predictions. A typical example is Cascade R-CNN [1]. Our proposed IMFA falls under the umbrella of guided refinement. However, IMFA's guided refinement is inherited from its baseline methods (e.g., DETR, Conditional DETR, Anchor DETR, DAB-DETR, etc.) that involve multiple decoder layers as refinement stages. We highlight that IMFA does not introduce any additional refinement stages, and neither do we claim IMFA's guided refinement as a novelty or contribution. Our major contribution is that, on top of Transformer-based detectors' guided refinement patterns, we propose IMFA that efficiently and adaptively incorporates new information (sparse multi-scale features) at each detection stage to achieve superior detection performance at slight computational costs.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *CVPR*, 2018. 3

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with Transformers. In *ECCV*, 2020. 1

[3] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer in vision through object detection. In *NeurIPS*, 2021. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[5] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. In *CVPR*, 2021. 2

[6] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 2

[7] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. DAB-DETR: Dynamic anchor boxes are better queries for DETR. In *ICLR*, 2022. 1

[8] 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 *ICCV*, 2021. 1

[9] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 2

[10] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional DETR for fast training convergence. In *ICCV*, 2021. 1

[11] Hwanjun Song, Deqing Sun, Sanghyuk Chun, Varun Jampani, Dongyoon Han, Byeongho Heo, Wonjae Kim, and Ming-Hsuan Yang. ViDT: An efficient and effective fully transformer-based object detector. In *ICLR*, 2022. 1

[12] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor DETR: Query design for Transformer-based detector. In *AAAI*, 2022. 1

[13] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 1, 3