ASAG: Building Strong One-Decoder-Layer Sparse Detectors via Adaptive Sparse Anchor Generation

Shenghao Fu[1,3,4], Junkai Yan[1,4], Yipeng Gao[1,4], Xiaohua Xie[1,3,4*], Wei-Shi Zheng[1,2,3,4*]
[1]School of Computer Science and Engineering, Sun Yat-sen University, China, [2]Pengcheng Lab, China,
[3]Guangdong Province Key Laboratory of Information Security Technology, China,
[4]Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
{fushh7, yanjk3, gaoyp23}@mail2.sysu.edu.cn, xiexiaoh6@mail.sysu.edu.cn, wszheng@ieee.org

A. More Details about Loss Function

In this work, we use patches as the basic prediction units in Anchor Generator. We compute bipartite matching and losses for each patch independently and the targets for each patch are objects whose centers lie in the patch.

Further, we propose Query Weighting to stabilize the training process, which gives high-quality anchors with larger weights and vice versa. The `Norm` function is shown in Figure A-1. The variable x in the picture is the product of x_1 and x_2 in Equ. (1) of the main text. The monotonically increasing normalization function raises small values and keeps them smaller than 1.

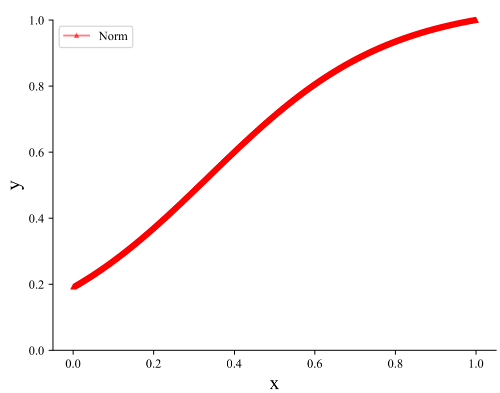

Figure A-1: Visualization of the normalization function in Query Weighting.

Following other DETR-like models, we use L1 loss and GIoU loss [17] with Query Weighting for box regression:

$$\begin{aligned}\mathcal{L}_{box}(\hat{b}, b) =& \lambda_1 \times w_{pos} \times \mathcal{L}_{L1}(\hat{b}, b) \\ &+ \lambda_2 \times w_{pos} \times \mathcal{L}_{GIoU}(\hat{b}, b),\end{aligned} \tag{A-1}$$

Denoising Training	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
	42.6	60.5	45.8	25.9	45.8	56.9
✓	43.1	60.2	46.7	25.1	45.8	58.4

Table A-1: Equipping ASAG-A with Denoising Training.

where $\hat{b}$ and b are the ground truth and the predicted box, respectively. The λ_1 and λ_2 are set to 5 and 2. w_{pos} is defined in Equ. (2) in the main text. The classification loss for negative samples is sigmoid focal loss [12] and the classification loss for positive samples is defined as follows:

$$\mathcal{L}_{cls}(s) = -\lambda_3 \times (w_{pos} \times \log s + w_{neg} \times \log(1-s)), \tag{A-2}$$

where s is the classification score with respect to the corresponding class and λ_3 is set to 2. w_{neg} is defined in Equ. (3) in the main text. In particular, w_{pos} in classification loss for Anchor Generator is set to IoU as dynamic anchors are class-agnostic and the location scores should be highly correlated to IoUs for selection. The overall losses are the sum of all components:

$$\mathcal{L}_{all} = \lambda_{an}\mathcal{L}_{anchor} + \mathcal{L}_{proposal} + \mathcal{L}_{final} + \sum_{i=0}^{2} \mathcal{L}^i_{auxiliary}, \tag{A-3}$$

Different from losses, the matching cost in bipartite matching does not use Query Weighting.

B. More Comparison with Other Well-Known Detectors

In this work, we aim to narrow the performance gap between one- and six-decoder-layer detectors and retain the fast speed by Adaptive Sparse Anchor Generation. Thus the performance of our models is highly related to baselines. However, ASAGs with only one decoder layer and fewer FLOPs still provide encouraging performance compared to well-known detectors, as shown in Table A-2.

* denotes the corresponding authors.

Detector	Backbone	#Layers	#Epochs	GFLOPs	AP	AP_{50}	AP_{75}	AP_s	AP_m	AP_l
DETR [1]	ResNet-50-DC5	6	500	187	43.3	63.1	45.9	22.5	47.3	61.1
SMCA [6]	ResNet-50	6	50	152	43.7	63.6	47.2	24.2	47.0	60.4
Deformable DETR [25]	ResNet-50	6	50	173	43.8	62.6	47.7	26.4	47.1	58.0
Sparse RCNN [18]	ResNet-50	6	36	152	45.0	63.4	48.2	26.9	47.2	59.5
Dynamic Sparse RCNN [8]	ResNet-50	6	36	-	47.2	66.5	51.2	30.1	50.4	61.7
Conditional DETR [15]	ResNet-50-DC5	6	108	195	45.1	65.4	48.5	25.3	49.0	62.2
Anchor DETR [19]	ResNet-50-DC5	6	50	151	44.2	64.7	47.5	24.7	48.2	60.6
DAB-DETR [13]	ResNet-50-DC5	6	50	202	44.5	65.1	47.7	25.3	48.2	62.3
DN-DETR [11]	ResNet-50-DC5	6	50	202	46.3	66.4	49.7	26.7	50.0	64.3
SAM-DETR-R50 w/ SMCA [21]	ResNet-50-DC5	6	50	210	45.0	65.4	47.9	26.2	49.0	63.3
DINO-4scale [23]	ResNet-50	6	24	279	49.9	67.4	54.5	31.8	53.3	64.3
AdaMixer [7]	ResNet-50	6	36	125	47.0	66.0	51.1	30.1	50.2	61.8
DAB-DETR-R50 + IMFA [22]	ResNet-50	6	50	108	45.5	65.0	49.3	27.3	48.3	61.6
REGO-Deformable DETR [5]	ResNet-50	12	50	190	47.6	66.8	51.6	29.6	50.6	62.3
SAP-DETR-DC5 [14]	ResNet-50-DC5	6	50	197	46.0	65.5	48.9	26.4	50.2	62.6
Efficient DETR [20]	ResNet-50	1	36	210	45.1	63.1	49.1	28.3	48.4	59.0
Cascade Featurized QRCNN [24]	ResNet-50	2	36	148	44.6	63.1	48.9	29.5	47.4	57.5
ASAG-S (Ours)	ResNet-50	1	36	136	45.0	64.1	49.1	29.5	47.4	57.8
ASAG-D (Ours)	ResNet-50	1	36	182	45.8	64.1	49.4	27.3	49.6	61.0
ASAG-A (Ours)	ResNet-50	1	36	139	46.3	65.1	50.3	29.9	49.2	59.6
DETR [1]	ResNet-101-DC5	6	500	253	44.9	64.7	47.7	23.7	49.5	62.3
SMCA [6]	ResNet-101	6	50	218	44.4	65.2	48.0	24.3	48.5	61.0
Sparse RCNN [18]	ResNet-101	6	36	250	46.4	64.6	49.5	28.3	48.3	61.6
Dynamic Sparse RCNN [8]	ResNet-101	6	36	-	47.8	67.0	52.0	31.0	51.1	62.2
Conditional DETR [15]	ResNet-101-DC5	6	108	262	45.9	66.8	49.5	27.2	50.3	63.3
DAB-DETR [13]	ResNet-101-DC5	6	50	282	45.8	65.9	49.3	27.0	49.8	63.8
DN-DETR [11]	ResNet-101-DC5	6	50	282	47.3	67.5	50.8	28.6	51.5	65.0
AdaMixer [7]	ResNet-101	6	36	201	48.0	67.0	52.4	30.0	51.2	63.7
REGO-Deformable DETR [5]	ResNet-101	12	50	257	48.5	67.0	52.4	29.5	52.0	64.4
SAP-DETR-DC5 [14]	ResNet-101-DC5	6	50	266	46.9	66.7	50.5	27.9	51.3	64.3
Efficient DETR [20]	ResNet-101	1	36	289	45.7	64.1	49.5	28.2	49.1	60.2
Cascade Featurized QRCNN [24]	ResNet-101	2	36	215	45.8	64.4	49.9	30.1	48.5	60.1
ASAG-A (Ours)	ResNet-101	1	36	206	47.5	66.1	51.2	30.4	50.6	62.6

Table A-2: Performance of different query-based detectors on COCO `minival` set with a 3$\times$ training schedule and single scale testing.

Note that some SOTA methods propose some advanced training techniques rather than novel decoder structures and these techniques can also boost the performance of ASAG, such as denoising training [11, 23], more positives [3, 10, 26, 16], knowledge distillation [9, 2, 4]. In Table A-1, we equip ASAG-A with 200 noised queries following DN-DETR [11]. The results show that Denoising Training can also benefit our methods.

C. More Visualization

In Figure C-2, we visualize all the bounding boxes appearing through the pipeline of ASAG-A. The anchors precisely cover the foreground objects and Adaptive Probing sparsely explores large feature maps. The number of patches and the location of patches vary according to different images. In particular, the last image does not use Adaptive Probing by the early-stop mechanism since there is no small object in the image. With precise anchors, the final predictions are as close as ground truth. For the first image, we can even predict more fine-grained bounding boxes for books on the shelf than ground truth.

In Figure C-3, we compare feature maps of our models with corresponding six-decoder-layer sparse detectors and dense-initialized ones. Different from dense ones that activate the whole object uniformly, ASAGs highlight the discriminative parts of objects and pay more attention to the background, similar to six-decoder-layer sparse detectors.

References

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

[2] Jiahao Chang, Shuo Wang, Guangkai Xu, Zehui Chen, Chenhongyi Yang, and Feng Zhao. Detrdistill: A universal knowledge distillation framework for detr-families. *arXiv preprint arXiv:2211.10156*, 2022. 2

[3] Qiang Chen, Xiaokang Chen, Gang Zeng, and Jingdong Wang. Group detr: Fast training convergence with decoupled one-to-many label assignment. *arXiv preprint arXiv:2207.13085*, 2022. 2

Figure C-2: More visualization of bounding boxes in our pipeline. All boxes without selection are drawn in the pictures. Patches and anchors are drawn in red and white, respectively. Different colors for dynamic proposals, final predictions, and ground truth are used to separate different classes in each image.

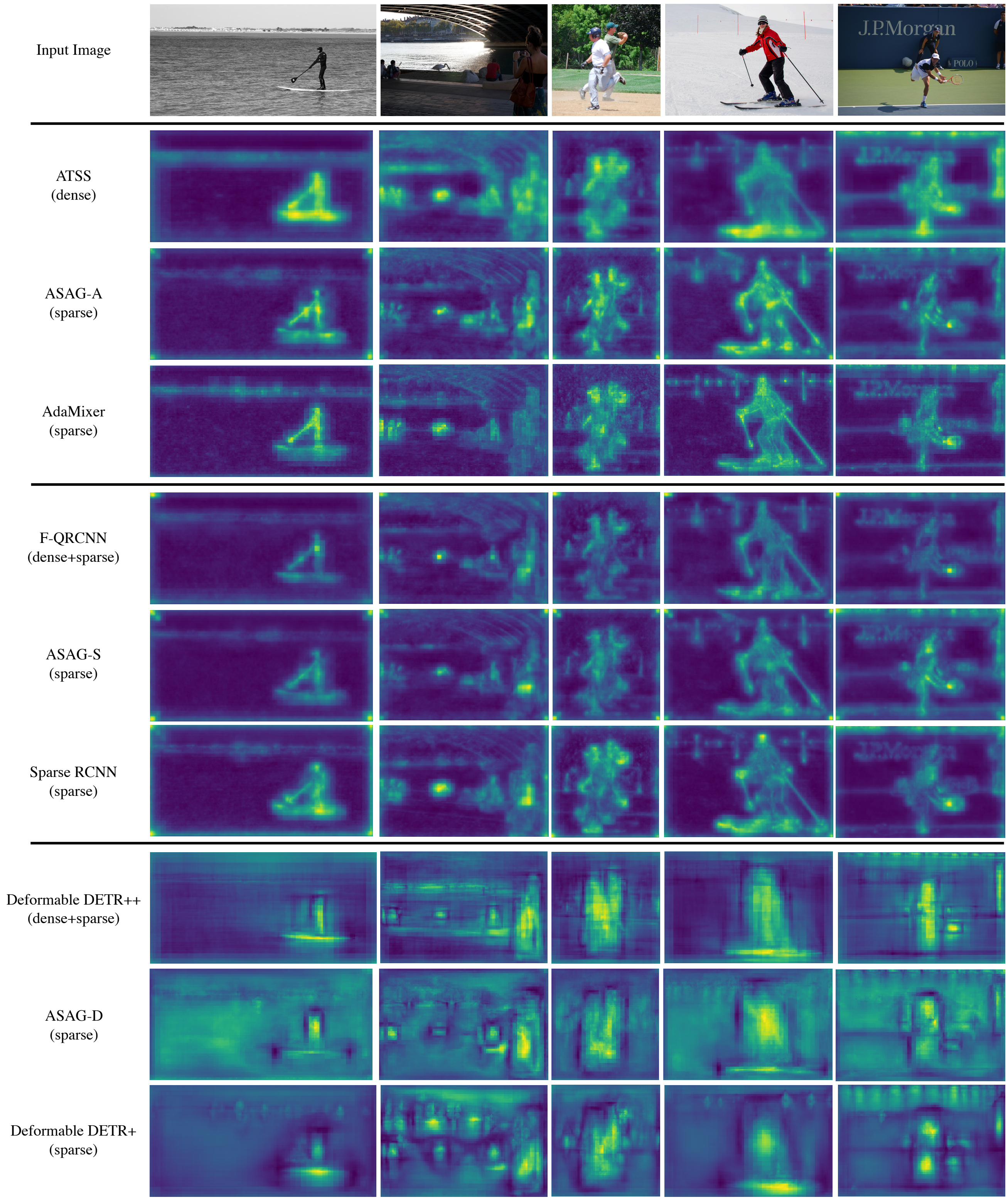

Figure C-3: More visualization of feature maps. Feature maps of ASAGs with sparse initialization are more similar to six-decoder-layer sparse detectors, which highlight the discriminative parts of foreground objects.

[4] Xiaokang Chen, Jiahui Chen, Yan Liu, and Gang Zeng. D 3 etr: Decoder distillation for detection transformer. *arXiv preprint arXiv:2211.09768*, 2022. 2

[5] Zhe Chen, Jing Zhang, and Dacheng Tao. Recurrent glimpse-based decoder for detection with transformer. In *CVPR*, 2022. 2

[6] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of detr with spatially modulated co-attention. In *ICCV*, 2021. 2

[7] Ziteng Gao, Limin Wang, Bing Han, and Sheng Guo. Adamixer: A fast-converging query-based object detector. In *CVPR*, 2022. 2

[8] Qinghang Hong, Fengming Liu, Dong Li, Ji Liu, Lu Tian, and Yi Shan. Dynamic sparse r-cnn. In *CVPR*, 2022. 2

[9] Linjiang Huang, Kaixin Lu, Guanglu Song, Liang Wang, Si Liu, Yu Liu, and Hongsheng Li. Teach-detr: Better training detr with teachers. *arXiv preprint arXiv:2211.11953*, 2022. 2

[10] Ding Jia, Yuhui Yuan, Haodi He, Xiaopei Wu, Haojun Yu, Weihong Lin, Lei Sun, Chao Zhang, and Han Hu. Detrs with hybrid matching. *arXiv preprint arXiv:2207.13080*, 2022. 2

[11] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *CVPR*, 2022. 2

[12] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 1

[13] 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. 2

[14] Yang Liu, Yao Zhang, Yixin Wang, Yang Zhang, Jiang Tian, Zhongchao Shi, Jianping Fan, and Zhiqiang He. Sap-detr: Bridging the gap between salient points and queries-based transformer detector for fast model convergency. *arXiv preprint arXiv:2211.02006*, 2022. 2

[15] 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. 2

[16] Jeffrey Ouyang-Zhang, Jang Hyun Cho, Xingyi Zhou, and Philipp Krähenbühl. Nms strikes back. *arXiv preprint arXiv:2212.06137*, 2022. 2

[17] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. 1

[18] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *CVPR*, 2021. 2

[19] Yingming Wang, Xiangyu Zhang, Tong Yang, and Jian Sun. Anchor detr: Query design for transformer-based detector. In *AAAI*, 2022. 2

[20] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021. 2

[21] Gongjie Zhang, Zhipeng Luo, Yingchen Yu, Kaiwen Cui, and Shijian Lu. Accelerating detr convergence via semantic-aligned matching. In *CVPR*, 2022. 2

[22] Gongjie Zhang, Zhipeng Luo, Yingchen Yu, Zichen Tian, Jingyi Zhang, and Shijian Lu. Towards efficient use of multi-scale features in transformer-based object detectors. *arXiv preprint arXiv:2208.11356*, 2022. 2

[23] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 2

[24] Wenqiang Zhang, Tianheng Cheng, Xinggang Wang, Qian Zhang, and Wenyu Liu. Featurized query r-cnn. *arXiv preprint arXiv:2206.06258*, 2022. 2

[25] 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. 2

[26] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. *arXiv preprint arXiv:2211.12860*, 2022. 2