DFormerv2: Geometry Self-Attention for RGBD Semantic Segmentation

Supplementary Material

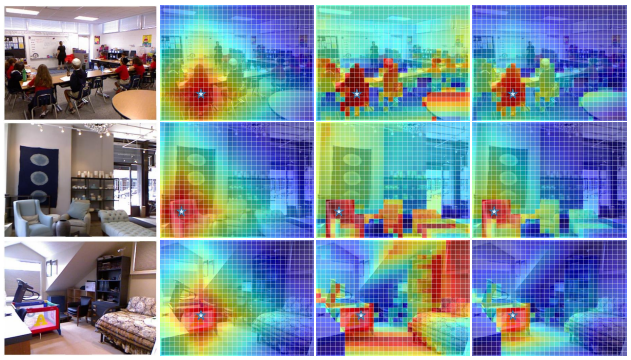


image spatial prior depth prior geometry prior

Figure 1. Visualization samples for spatial, depth, and geometry prior. The blue 'star' means the current query token.

pretraining config	DFormerv2-S/B/L
input size	224×224
weight init	trunc. normal (0.2)
optimizer	AdamW
base learning rate	1e-3
weight decay	0.05
optimizer momentum	β_1, β_2=0.9, 0.999
batch size	1024
training epochs	300
learning rate schedule	cosine decay
warmup epochs	5
warmup schedule	linear
layer-wise lr decay	F
randaugment	(9, 0.5)
mixup	0.8
cutmix	1.0
random erasing	0.25
label smoothing	0.1
stochastic depth	0.1/0.3/0.5
head init scale	F
gradient clip	F
token label	T
exp. mov. avg. (EMA)	T

Table 1. Pretraining settings. All the pretraining experiments are conducted on 8 3090 GPUs.

1. More insights about geometry prior.

To better understand the functions of depth and spatial priors, we present the visualization for them and geometry prior in Fig. 1. It is a supplement for the Fig.7 and Tab.3 of the main paper. Two objects that are far apart may not differ much in depth. Thus, spatial perception is also required. Combining the depth and spatial priors, our geometry priors can better reflect the 3D position relationships across the whole scene.

2. Configuration of our models

The detailed configurations for the three scales of our DFormerv2 are shown in Tab. 2.

3. Details about the decomposition

The detailed decomposed self-attention is shown in Fig. 2.

4. Experimental Details

4.1. Pretraining settings

The details for the pretraining in DFormerv2 is shown in Tab. 1.

4.2. Finetuning settings.

Details for the finetuning experiments are shown in Tab. 3.

4.3. Details of the experiments in Tab.7 of the main paper.

In Tab.7 of the main paper, we conduct experiments to see the effect of RGB and depth on classification and segmentation. To exclude the influence of factors outside the input modalities, we adopt similar architecture for the three input manners. For the architecture with RGB input, we adopt DFormerv2-S without the geometry prior part. For the architecture with depth input, we use the same architecture with RGB and change the input channel of stem layer from 3 to 1. For the architecture with RGB-D input, we employ DFormerv2-S. The training processes for classification and segmentation are separate. The classification training settings is the pretraining settings. The foreground segmentation adopt the common settings in foreground segmentation/ salient object detection [1, 3, 4]. For performance evaluation, we adopt two golden metrics of the binary segmentation, *i.e.,* mean absolute error (MAE) [6], weighted F-measure (wF) [5].

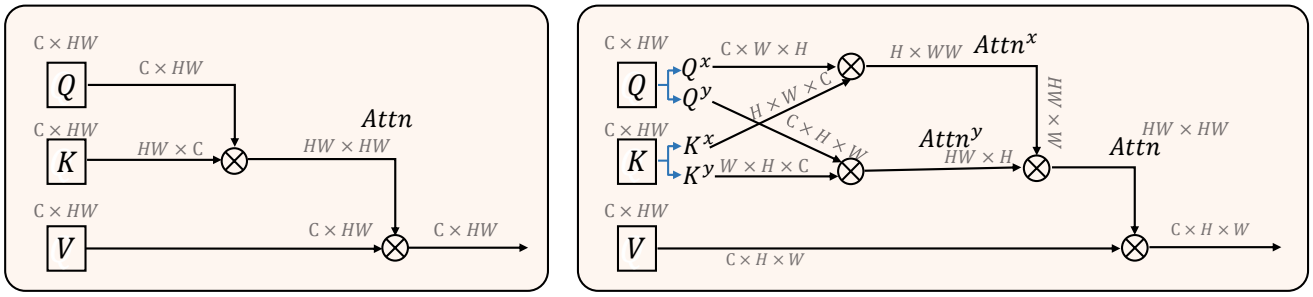
For the 50K samples and corresponding category label from ImageNet [7], we also incorporate the generated depth maps as same as DFormer [10] and the segmentation map from LUSS [2]. The segmentation annotations from LUSS provide the mask for the foreground in the scene, which is highly related to category. We split the 50K samples to 45K and 5K for training and validation respectively.

References

[1] Deng-Ping Fan, Ming-Ming Cheng, Jiang-Jiang Liu, Shang-Hua Gao, Qibin Hou, and Ali Borji. Salient objects in clut-

Stage	Output size	Expansion	DFormerv2-S	DFormerv2-B	DFormerv2-L
1	$\frac{H}{4} \times \frac{W}{4}$	4	$C_1 = 64, N_1 = 3$	$C_1 = 80, N_1 = 4$	$C_1 = 112, N_1 = 4$
2	$\frac{H}{8} \times \frac{W}{8}$	4	$C_2 = 128, N_2 = 4$	$C_2 = 160, N_2 = 8$	$C_2 = 224, N_2 = 8$
3	$\frac{H}{16} \times \frac{W}{16}$	3	$C_3 = 256, N_3 = 18$	$C_3 = 320, N_3 = 25$	$C_3 = 448, N_3 = 25$
4	$\frac{H}{32} \times \frac{W}{32}$	3	$C_4 = 512, N_4 = 4$	$C_4 = 512, N_4 = 8$	$C_4 = 640, N_4 = 8$
Decoder dimension			512	512	1024
Parameters (M)			26.7	53.9	95.5

Table 2. Detailed configurations of the proposed DFormerv2. 'C_i' represents the channel number in i-th stage. 'N_i' is the number of building blocks in i-th stage. 'Expansion' is the expand ratio for the number of channels in MLPs. 'Decoder dimension' denotes the channel dimension in the decoder.



(a) Self-Attention (b) Decomposed Self-Attention

Figure 2. Decomposition on the attention. Here we emphasize the decomposition operation and omit the geometry prior for simplicity.

Pretraining config	DFormerv2-S	DFormerv2-B	DFormerv2-L
input size	480×640 / 480^2	480×640 / 480^2	480×640 / 480^2
optimizer	AdamW	AdamW	AdamW
base learning rate	6e-5/8e-5	6e-5/8e-5	6e-5/8e-5
weight decay	0.01	0.01	0.01
batch size	8/16	8/16	8/16
epochs	500/300	500/300	500/300
optimizer momentum	β_1, β_2=0.9, 0.999	β_1, β_2=0.9, 0.999	β_1, β_2=0.9, 0.999
learning rate schedule	linear decay	linear decay	linear decay
warmup epochs	10	10	10
warmup schedule	linear	linear	linear
layer-wise lr decay	None	None	None
aux head	None	None	None
stochastic depth	0.1/0.1	0.1/0.1	0.2/0.3

Table 3. **DFormerv2 finetuning settings on NYUDepthv2 [8]/SUNRGBD [9]**. Multiple stochastic depth rates, input sizes and batch sizes are for NYUDepthv2 and SUNRGBD datasets respectively. All the finetuning experiments for RGB-D semantic segmenations are conducted on 4 NVIDIA 3090 GPUs.

ter: Bringing salient object detection to the foreground. In *ECCV*, 2018. 1

[2] Shanghua Gao, Zhong-Yu Li, Ming-Hsuan Yang, Ming-Ming Cheng, Junwei Han, and Philip Torr. Large-scale unsupervised semantic segmentation. *IEEE TPAMI*, 45(6):7457–7476, 2022. 1

[3] Jiang-Jiang Liu, Qibin Hou, Ming-Ming Cheng, Jiashi Feng, and Jianmin Jiang. A simple pooling-based design for real-time salient object detection. In *CVPR*, 2019. 1

[4] Nian Liu, Ziyang Luo, Ni Zhang, and Junwei Han. Vst++: Efficient and stronger visual saliency transformer. *IEEE TPAMI*, 2024. 1

[5] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *IEEE CVPR*, 2014. 1

[6] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *IEEE CVPR*, 2012. 1

[7] Olga Russakovsky et al. ImageNet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 1

[8] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob

Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012. 2

[9] Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *CVPR*, 2015. 2

[10] Bowen Yin, Xuying Zhang, Zhongyu Li, Li Liu, Ming-Ming Cheng, and Qibin Hou. Dformer: Rethinking rgbd representation learning for semantic segmentation. *ICLR*, 2024. 1