SegMAN: Omni-scale Context Modeling with State Space Models and Local Attention for Semantic Segmentation

Supplementary Material

Yunxiang Fu* Meng Lou* Yizhou Yu†
School of Computing and Data Science, The University of Hong Kong
yunxiang@connect.hku.hk, loumeng@connect.hku.hk, yizhouy@acm.org

A. Additional ablation studies

We present additional ablation studies on the MMSCopE module (refer to Figure 3 (d) in the main paper) using the ADE20K [65] dataset. Specifically, we investigate the effect of diffusion feature fusion strategies, the effect of feature map resolutions on simultaneous SS2D scans, and then verify the impact of Pixel Shuffle and Pixel Unshuffle operations on preserving feature map details.

Effect of feature fusion in the decoder. Table 8 illustrates the importance of our feature fusion approach (Figure 3(c)). Direct prediction from multi-scale context feature F' results in significant performance degradation (-1.0%). While incorporating the feature map F improves performance to 50.9%, our fusion strategy with stage-specific features F_2', F_{up3} F_{up4} achieves the optimal performance of 51.3% mIoU. Removal of average-pooled features and not adding F' to stage-specific features reduces effectiveness. For a fair comparison, average-pooled features are included in experiments without stage-specific features.

Input to classifier	Params (M)	GFLOPs	mIoU
SegMAN-S	29.4	25.3	51.3
w/o stage-specific features			
Concat (F,F')	29.5	24.6	50.9 (-0.4)
$F + F'$	29.2	25.2	50.8 (-0.5)
F'	29.2	24.4	50.3 (-1.0)
with stage-specific features			
w/o avg pool	29.3	25.2	50.9 (-0.4)
w/o addition	29.3	25.2	51.1 (-0.2)

Table 8. Effect of fusion strategies in the SegMAN decoder.

Effect of feature map resolution on SS2D scans. We examine how different spatial resolutions for simultaneous SS2D scans influence performance. In our proposed MMSCopE module, feature maps are rescaled and concatenated at a resolution of $\frac{H}{32} \times \frac{W}{32}$ (i.e., 16×16 for 512×512 input images in ADE20K). To assess the impact of higher resolutions, we experiment with rescaling feature maps to $\frac{H}{16} \times \frac{W}{16}$ (32×32) and $\frac{H}{8} \times \frac{W}{8}$ (64×64) resolutions. To gather feature maps at the $\frac{H}{8} \times \frac{W}{8}$ resolution, we upsample the feature maps using the Pixel Shuffle operation [42], which rearranges elements from the channel dimension into the spatial dimension, effectively increasing spatial resolution while preserving feature information. For the $\frac{H}{16} \times \frac{W}{16}$ resolution, we apply Pixel Unshuffle to the $\frac{H}{8} \times \frac{W}{8}$ feature maps, and Pixel Shuffle to the $\frac{H}{32} \times \frac{W}{32}$.

As shown in Table 9, scanning at the proposed $\frac{H}{32} \times \frac{W}{32}$ resolution achieves the best performance. Scanning at higher resolutions results in decreased mIoU scores. This decline occurs because higher resolutions lead to reduced channel dimensions in the feature maps after applying Pixel Shuffle operations. Specifically, the Pixel Shuffle operation decreases the channel dimension by factors of 4 and 16 when upsampling by factors of 2 and 4, respectively. This significant reduction in channel dimensions limits the SS2D's learning capacity, thereby negatively impacting performance.

Impact of Pixel Unshuffle Operation. We evaluate replacing the Pixel Unshuffle operation with bilinear interpolation when preparing feature maps for SS2D scanning at the $\frac{H}{32} \times \frac{W}{32}$ resolution. Pixel Unshuffle downscales feature maps without information loss, ensuring the downsampled maps fully represent the original features despite reduced spatial resolution. Processing these maps together enables simultaneous handling of multiple scales, effectively modeling multi-scale information.

As shown in Table 9, substituting Pixel Unshuffle with bilinear interpolation reduces mIoU from 50.0% to 49.0%. This confirms that preserving the full representational capacity during downsampling is crucial. Bilinear interpolation, a smoothing operation, loses fine-grained spatial information, leading to diminished segmentation accuracy. Therefore, the Pixel Unshuffle operation is vital for maintaining multi-scale contextual information.

Encoder hyperparameter ablations. Table 10 compares the effect of different window sizes and SS2D parameter settings on SegMAN-T. Default setting (window size [11,9,7,7]) yields best performance.

B. Detailed backbone comparison

Table 11 presents a detailed comparison of ImageNet-1k classification accuracy. Additional representative backbones (PVTv2 [50], MaxViT [47], MambaTree [55]) are

Model Variant	Params (M)	GFLOPs	mIoU
SegMAN-S	29.9	24.6	50.0
Scan Resolution			
$\frac{W}{16} \times \frac{W}{16}$ (32×32)	30.5	25.2	49.5 (-0.5)
$\frac{W}{8} \times \frac{W}{8}$ (64×64)	29.7	26.6	49.2 (-0.8)
Downsample Method			
Bilinear Interpolation	29.9	24.5	49.0 (-1.0)

Table 9. Additional ablation studies on the MMSCopE module in SegMAN decoder.

Encoder config	Param (M)	GFLOP	ImageNet-1k	ADE20k
Window size for each stage [13,11,9,7]	5.1 (+0.0)	4.8 (+0.2)	76.4 (+0.2)	43.2 (-0.1)
Window size for each stage [9,7,7,7]	5.1 (+0.0)	4.4 (-0.2)	76.1 (-0.1)	43.1 (-0.2)
SSM expansion ratio 1 → 2	5.3 (+0.2)	5.2 (+0.6)	76.3 (+0.1)	43.0 (-0.3)
SSM state dimension N 1 → 16	5.3 (+0.2)	6.5 (+1.9)	76.5 (+0.3)	43.1 (-0.2)

Table 10. Effect of Encoder window size and SSM configurations.

included for comparison.

C. Generalization

Table 12 empirically demonstrates the modular compatibility of SegMAN components across two representative frameworks: SegNeXt and CGRSeg. Replacing SegNeXt's encoder with our SegMAN-S Encoder reduces parameters by 9% and GFLOPs by 15% while improving ADE20K mIoU by +0.7%; substituting its decoder achieves +0.4% mIoU at 14% lower computation. Similarly, integrating our encoder into CGRSeg yields +1.7% mIoU, while our decoder enhances its performance by +0.8% mIoU. These results quantify the efficacy of our encoder and decoder in balancing accuracy-efficiency trade-offs, validating that either component can independently upgrade existing pipelines. The bidirectional improvements underscore SegMAN's plug-and-play adaptability, where each module achieves an optimal balance of performance gains (up to +1.7% mIoU) and computational pragmatism across diverse architectures.

D. Panoptic and instance segmentation

To demonstrate task-agnostic capabilities, we deploy our SegMAN-S Encoder in Mask DINO [24] for panoptic and instance segmentation. Replacing its default ResNet50 backbone with our ImageNet-1k pretrained encoder as well as the MiT-B2 [56] backbone in SegFormer. We maintain Mask DINO's architecture while increasing batch size from 16 to 48 for training efficiency.

As shown in Table 13, SegMAN-S achieves 49.6 instance AP (+3.3 over ResNet50, +2.0 over MiT-B2) and 56.8 panoptic PQ (+3.8/+2.1) while operating at 283 GFLOPs, which is 6% fewer than ResNet50 (286 GFLOPs) and 10% fewer than MiT-B2 (315 GFLOPs). Despite comparable parameter counts (48.3M vs. 48.5M MiT-B2), our encoder delivers superior multi-task performance, validating its effectiveness beyond semantic segmentation.

Models	Params (M)	GFLOPs	Acc
MiT-B0 [56]	3.8	0.60	70.5
EFT-T [60]	3.7	0.60	72.3
MSCAN-T [18]	4.2	0.89	75.9
SegMAN-T Encoder	3.5	0.65	**76.2**
MiT-B2 [56]	24	4.0	81.6
EFT-B [60]	26	4.2	82.4
MSCAN-B [18]	27	4.4	83.0
Swin-T [29]	28	4.5	81.2
PVTv2-B2 [50]	25	4.0	79.8
ConvNeXt-T [30]	28	4.5	82.1
InternImage-T [51]	30	5.0	83.5
MaxViT-T [47]	31	5.6	83.7
ViM-S [67]	26	-	81.6
VMamba-T [28]	29	4.9	82.6
MambaTreev-T [55]	30	4.8	83.4
SparX-Mamba-T [32]	27	5.2	83.5
SegMAN-S Encoder	26	4.1	**84.0**
MiT-B3 [56]	45	6.9	83.1
MSCAN-L [18]	45	9.1	83.9
Swin-S [29]	50	8.7	83.2
PVTv2-B3 [50]	45	6.9	83.2
ConvNeXt-S [30]	50	8.7	83.1
InternImage-S [51]	50	8.0	84.2
MaxViT-S [47]	69	11.7	84.5
VMamba-S [28]	50	8.7	83.6
MambaTreeV-S [55]	51	8.5	84.2
SparX-Mamba-S [32]	47	9.3	84.2
SegMAN-B Encoder	45	9.9	**85.1**
MiT-B5 [56]	82	11.8	83.8
Swin-B [29]	88	15.4	83.5
PVTv2-B5 [50]	82	11.8	83.8
ConvNeXt-B [30]	89	15.4	83.8
InternImage-B [51]	97	16.7	84.9
MaxViT-B [47]	120	24.0	84.9
ViM-B [67]	98	-	83.2
VMamba-B [28]	89	15.9	84.5
MambaTreeV-B [55]	91	15.1	84.8
SparX-Mamba-B [32]	84	15.9	84.5
SegMAN-L Encoder	81	16.8	**85.5**

Table 11. Detailed comparison of classification accuracy and computational complexity (FLOPs at 224×224 resolution) of encoder architectures on ImageNet-1K.

Configuration	Feature Encoder	Decoder	Param	GFLOP	ADE20k
SegNeXt	MSCAN-B	HAM	27.7	34.9	48.5
SegNeXt + our encoder	SegMAN-S	HAM	25.3	29.5	**49.2** (+0.7)
SegNeXt + our decoder	MSCAN-B	Ours	30.6	29.9	**48.9** (+0.4)
CGRSeg	EfficientFormerV2-L	CGRHead	35.7	16.5	47.3
CGRSeg + our encoder	SegMAN-S	CGRHead	44.2	24.5	**49.0** (+1.7)
CGRSeg + our decoder	EfficientFormerV2-L	Ours	29.9	17.9	**48.1** (+0.8)

Table 12. Encoder and Decoder generalization results.

E. Qualitative examples

We present qualitative examples of SegMAN's segmentation results on ADE20K Figures 4, Cityscapes 5, and

Encoder	Param (M)	GFLOP	Instance AP	Panoptic PQ
ResNet50	52	286	46.3	53.0
MiT-B2	48.5	315	47.6	54.7
SegMAN-S Encoder	48.3	283	**49.6** (+3.3)	**56.8** (+3.8)

Table 13. Panoptic and instance segmentation using Mask DINO.

COCO-Stuff-164K 6. For COCO-Stuff, comparisons are made with VWFormer and EDAFormer only, since the checkpoints for other segmentation models are not released. These figures illustrate SegMAN's capability to capture both fine-grained local dependencies and long-range contextual information. Compared to other segmentation methods, SegMAN yields more precise boundaries and accurately identifies intricate details within the scenes. These qualitative results verify our quantitative findings, highlighting the benefits of SegMAN's ability to capture fine-grained details while maintaining global context, which is unattainable by existing approaches.

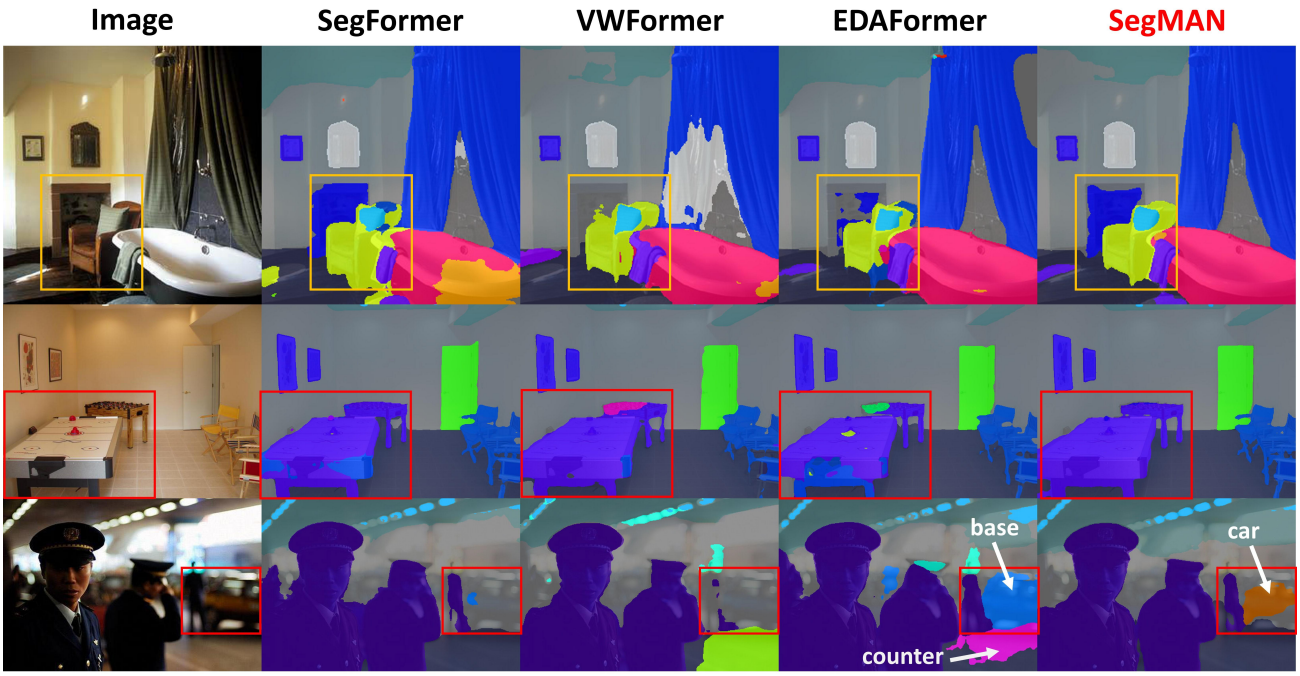

Figure 4. **Qualitative results on ADE20K.** Zoom in for best view.

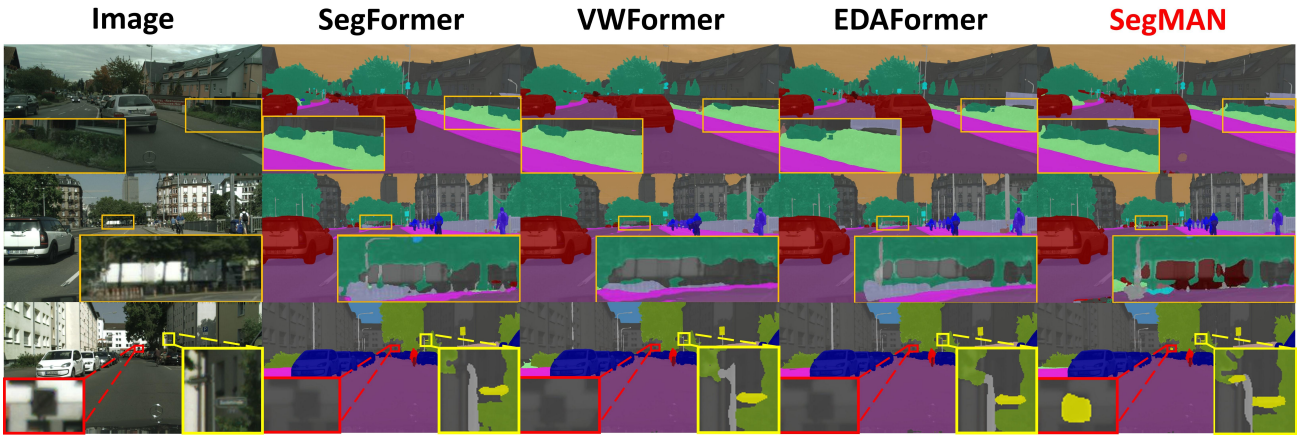

Figure 5. **Qualitative results on Cityscapes.** Zoom in for best view.

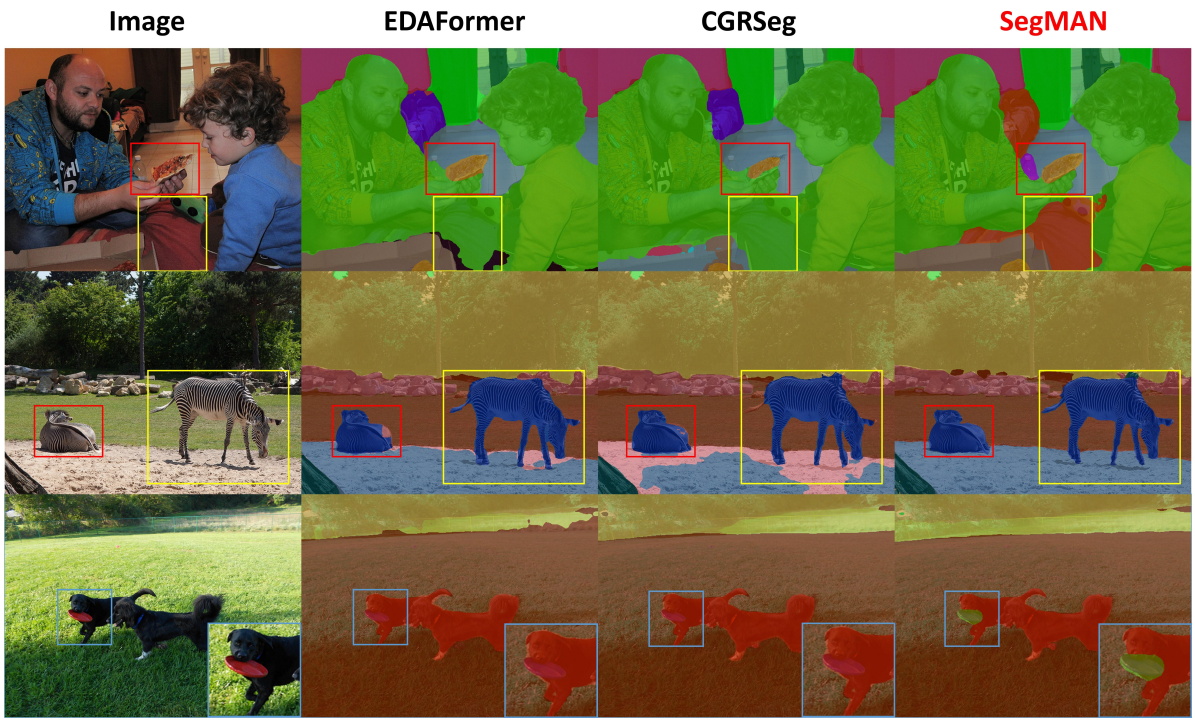

Figure 6. **Qualitative results on COCO-Stuff-164K.** We do not compare with SegFormer as its COCO-Stuff checkpoints are not released. Zoom in for best view.