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

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

Abstract

High-quality semantic segmentation relies on three key capabilities: global context modeling, local detail encoding, and multi-scale feature extraction. However, recent methods struggle to possess all these capabilities simultaneously. Hence, we aim to empower segmentation networks to simultaneously carry out efficient global context modeling, high-quality local detail encoding, and rich multi-scale feature representation for varying input resolutions. In this paper, we introduce SegMAN, a novel linear-time model comprising a hybrid feature encoder dubbed SegMAN Encoder, and a decoder based on state space models. Specifically, the SegMAN Encoder synergistically integrates sliding local attention with dynamic state space models, enabling highly efficient global context modeling while preserving fine-grained local details. Meanwhile, the MM-SCopE module in our decoder enhances multi-scale context feature extraction and adaptively scales with the input resolution. We comprehensively evaluate SegMAN on three challenging datasets: ADE20K, Cityscapes, and COCO-Stuff. For instance, SegMAN-B achieves 52.6% mIoU on ADE20K, outperforming SegNeXt-L by 1.6% mIoU while reducing computational complexity by over 15% GFLOPs. On Cityscapes, SegMAN-B attains 83.8% mIoU, surpassing SegFormer-B3 by 2.1% mIoU with approximately half the GFLOPs. Similarly, SegMAN-B improves upon VWFormer-B3 by 1.6% mIoU with lower GFLOPs on the COCO-Stuff dataset. Our code is available at https://github.com/yunxiangfu2001/SegMAN

1. Introduction

As a core computer vision task, semantic segmentation needs to assign a categorical label to every pixel within an image [31]. Accurate semantic segmentation demands three crucial capabilities. First, **global context modeling** establishes rich contextual dependencies regardless of

---

*Equal contribution

spatial distance, enabling overall scene understanding [13]. Second, **local detail encoding** learns fine-grained feature and boundary representations, crucial for differentiating semantic categories and localizing boundaries between adjacent regions with different semantic meanings [51]. Third, **context modeling based on multiple intermediate scales** learns semantic representations across multiple scales, addressing intra-class scale variations while enhancing inter-class discrimination [2, 49, 55].

Recent methods of semantic segmentation have been unable to simultaneously encapsulate all three of these capabilities effectively. For instance, VWFormer [49] introduces a varying window attention (VWA) mechanism that effectively captures multi-scale information through cross-attention between local windows and multiple windows with predefined scales. However, at higher input resolutions, the global context modeling capability of VWA diminishes because the predefined window sizes fail to maintain full feature map coverage. In addition, larger windows significantly increase the computational cost as self-attention exhibits quadratic complexity. Likewise, EDAFormer [52] proposes an embedding-free spatial reduction attention (SRA) mechanism to implement global attention efficiently and an all-attention decoder for global and multi-scale context modeling. However, fine-grained details are lost due to its reliance on downsampled features for token-to-region attention. Moreover, the absence of dedicated local operators in the feature encoder limits fine-grained feature learning [48, 52].

To present a more intuitive understanding of the aforementioned issues, we visualize the effective receptive field (ERF) maps and segmentation maps of recent state-of-the-art models [48, 49, 52] with 27M-29M parameters. As depicted in Figure 1 (*Left*), the ERF of VWFormer [49] and EDAFormer [52] has limited feature map coverage when processing high-resolution Cityscapes images. Meanwhile, the segmentation maps in Figure 1 (*Right*) reveal that existing methods have a limited ability to recognize fine-grained local details, which we attribute to insufficient local feature modeling in these architectures.
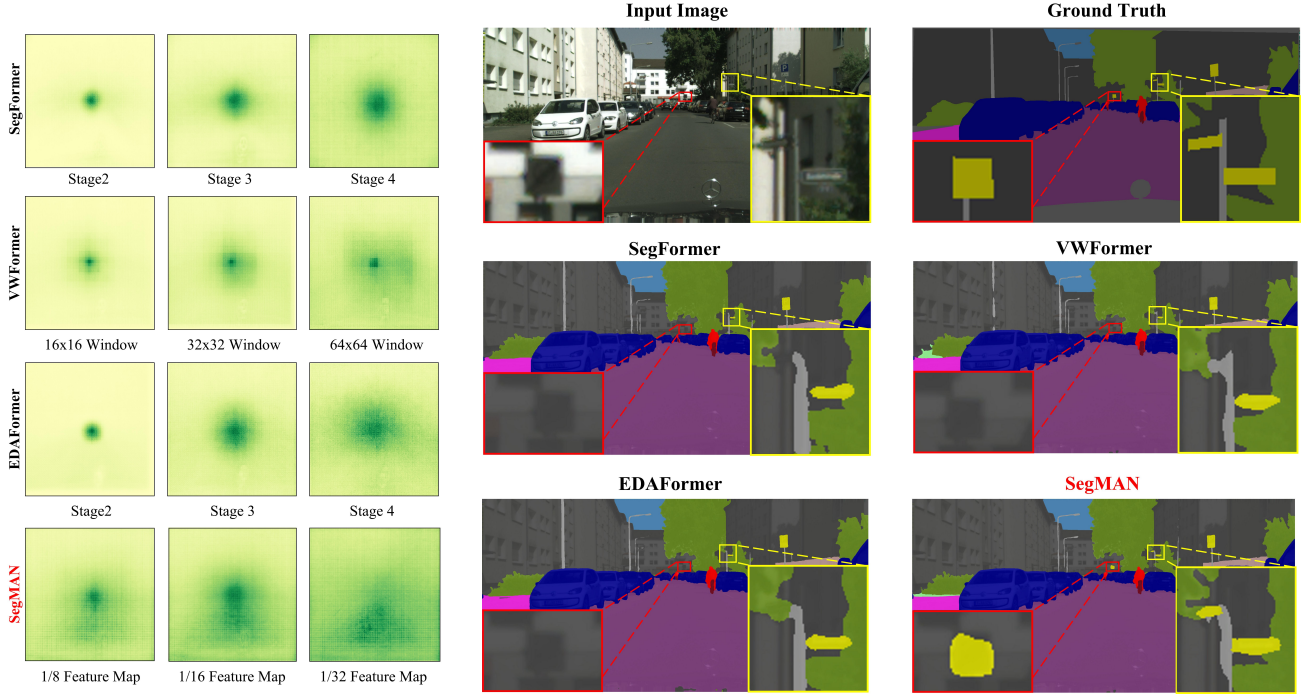
Figure 1. Qualitative analysis of receptive field patterns and segmentation performance for small-sized models (27M-29M parameters). *Left*: Visualization of effective receptive fields (ERF) on the Cityscapes validation set at 1024×1024 resolution, illustrating SegMAN's stronger global context modeling capacity in comparison to existing state-of-the-art models. *Right*: Segmentation maps highlighting SegMAN's superior capacity to encode fine-grained local details that are often missed by existing approaches.

In this work, we aim to encapsulate omni-scale context modeling for a varying input resolution within a semantic segmentation network. As a result, we propose SegMAN, a novel segmentation network capable of carrying out efficient global context modeling, high-quality local detail encoding, and rich context representation at diverse scales simultaneously. Our SegMAN introduces two novel sub-networks: (1) a hybrid feature encoder that integrates Local Attention and State Space models (LASS) in the token mixer, and (2) a decoder with a Mamba-based Multi-Scale Context Extraction (MMSCopE) module. In the SegMAN Encoder, LASS leverages a two-dimensional state space model (SS2D) [27] for global context modeling and neighborhood attention [19] (Natten) for local detail encoding, both functioning in linear time. The dynamic global scanning in our method always covers the full feature map, enabling robust global context modeling across varying input resolutions and all encoder layers. Meanwhile, by performing local attention within a sliding window, Natten exhibits an outstanding capability in fine-grained local detail encoding. On the other hand, leveraging the linear time complexity of Mamba, our decoder complements the encoder with robust feature learning across multiple scales. It first generates multiple versions of an aggregated feature map in different resolutions including the full resolution, each version

representing a different scale. Then it employs SS2D to scan these feature maps to learn multi-scale contexts. For higher efficiency, the feature maps at different scales are reshaped to the same resolution and concatenated so that scanning only needs to be performed on a single feature map. Note that our method differs from existing methods in two aspects. First, unlike multi-kernel convolution [2] and multi-scale window attention [49], our design adaptively scales with the input resolution. Second, by performing SS2D-based scanning over entire multi-scale feature maps, our method better preserves fine-grained details that are typically compromised by pooling operations [32, 55] or spatial reduction attention [52] in existing methods.

Our SegMAN achieves state-of-the-art performance on three widely used datasets: ADE20K [57], Cityscapes [8], and COCO-Stuff-164k [1]. Notably, our Base variant (51.8M parameters, 58.1 GFLOPs for ADE20K/COCO-Stuff, 479.0 GFLOPs for Cityscapes) achieves 52.6%, 83.8%, and 48.4% mIoU on these datasets respectively while maintaining competitive inference speed.

To summarize, our contributions are threefold:
- We introduce a novel encoder architecture featuring our LASS token mixer that synergistically combines local attention with state space models for efficient global context modeling and local detail encoding, achieving linear

2

time complexity.

- We propose MMSCopE, a novel decoder module that operates on multi-scale feature maps that adaptively scale with the input resolution, surpassing previous approaches in both fine-grained detail preservation and omni-scale context learning.
- We demonstrate through comprehensive experiments that SegMAN, powered by LASS and MMSCopE, establishes new state-of-the-art performance while maintaining competitive computational efficiency across multiple challenging semantic segmentation benchmarks.

2. Related work

Segmentation encoders. Prior research in semantic segmentation architectures typically comprises two key components: an encoder for feature extraction from input images and a decoder for multi-scale context learning and pixel-wise classification. While conventional approaches often adapt popular classification networks such as ResNet [20], ConvNeXt [29], PVT [44], and Swin-Transformer [28] as encoders, improvements in classification performance do not necessarily translate to enhanced segmentation capabilities [21], primarily due to the requirement for fine-grained spatial information in segmentation tasks. This observation has motivated the development of specialized segmentation-oriented encoders [18, 35, 40, 46, 48, 50, 52, 56]. In this paper, we introduce a novel segmentation encoder that integrates local attention with dynamic State Space Models (Mamba) [16]. In contrast to previous works, our design enables the simultaneous modeling of fine-grained details and global contexts while maintaining linear time complexity.

Segmentation decoders. Multi-scale feature learning plays a fundamental role in semantic segmentation, leading to numerous innovations in decoder architectures. Seminal works established various approaches to multi-scale feature aggregation, including the adaptive pooling strategy of PSPNet [55], atrous convolutions in the DeepLab series [2–4], and feature pyramid network (FPN) based decoders such as UPerNet [47] and Semantic FPN [23]. Contemporary architectures have advanced both efficiency and performance through diverse approaches to multi-scale feature fusion, including MLP-based methods (Segformer [48]), Matrix decomposition modules (SegNext [15, 18]), Transformers (FeedFormer [38], VWFormer [49]), and convolutions (CGRSeg [32]). An approach relevant to ours is VWFormer [49], which employs cross-attention between local windows and multiple larger windows with predefined scales to capture multi-scale contexts. However, its reliance on fixed window scales limits its global context modeling capability: as the input resolution increases, the predefined windows fail to maintain complete feature map coverage. In contrast, our proposed decoder adaptively extracts con-

text features at different scales while maintaining full feature map coverage regardless of the input resolution.

Vision Mamba. Dynamic state space models represented by Mamba [16] have demonstrated promising performance in sequence modeling. Mamba combines structured state space models [17, 39] with selective scanning, enabling global context modeling with dynamic weights and linear time complexity. These favorable properties have led to promising results on vision tasks. For example, VMamba [27] and ViM [59] are Mamba-based vision backbone networks. Other works have explored Mamba for medical imaging [25, 36], 3D point clouds [26, 54], and image/video generation [12, 14, 41]. In this paper, we leverage Mamba to learn global contexts in the encoder and multi-scale contexts in the decoder for segmentation.

3. Method

3.1. Overall Architecture

As shown in Figure 2, our proposed SegMAN consists of a newly designed feature encoder and decoder. Specifically, we propose a novel hybrid feature encoder based on core mechanisms from both Transformer and dynamic State Space Models [16], and a simple yet effective Mamba-based Multi-Scale Context Extraction module (MMSCopE) in the decoder. The hierarchical SegMAN Encoder integrates Neighborhood Attention (Natten) [19] and the 2D-Selective-Scan Block (SS2D in VMamba) [27] within the token mixer termed Local Attention and State Space (LASS), enabling comprehensive feature learning at both global and local scales across all layers while maintaining linear computational complexity. Meanwhile, MMSCoPE dynamically and adaptively extracts multi-scale semantic information by processing feature maps at varying levels of granularity using SS2D [27], with the scales adaptively adjusted according to the input resolution. Collectively, the SegMAN Encoder injects robust global contexts and local details into a feature pyramid, where feature maps are progressively downsampled and transformed to produce omni-scale features at different pyramid levels. This feature pyramid is then fed into MMSCoPE, where features are further aggregated spatially as well as across scales, resulting in comprehensive multi-scale representations that can be used for dense prediction.

3.2. Feature encoder

Overview. As shown in Figure 2, our SegMAN Encoder is a standard four-stage network [28, 29], where each stage begins with a strided 3×3 convolution for spatial reduction, followed by a series of LASS Blocks. Each LASS Block includes pre-layer normalization, a novel LASS module, and a feedforward network (FFN) [10, 19, 28]. Our proposed LASS represents the first integration of local self-attention
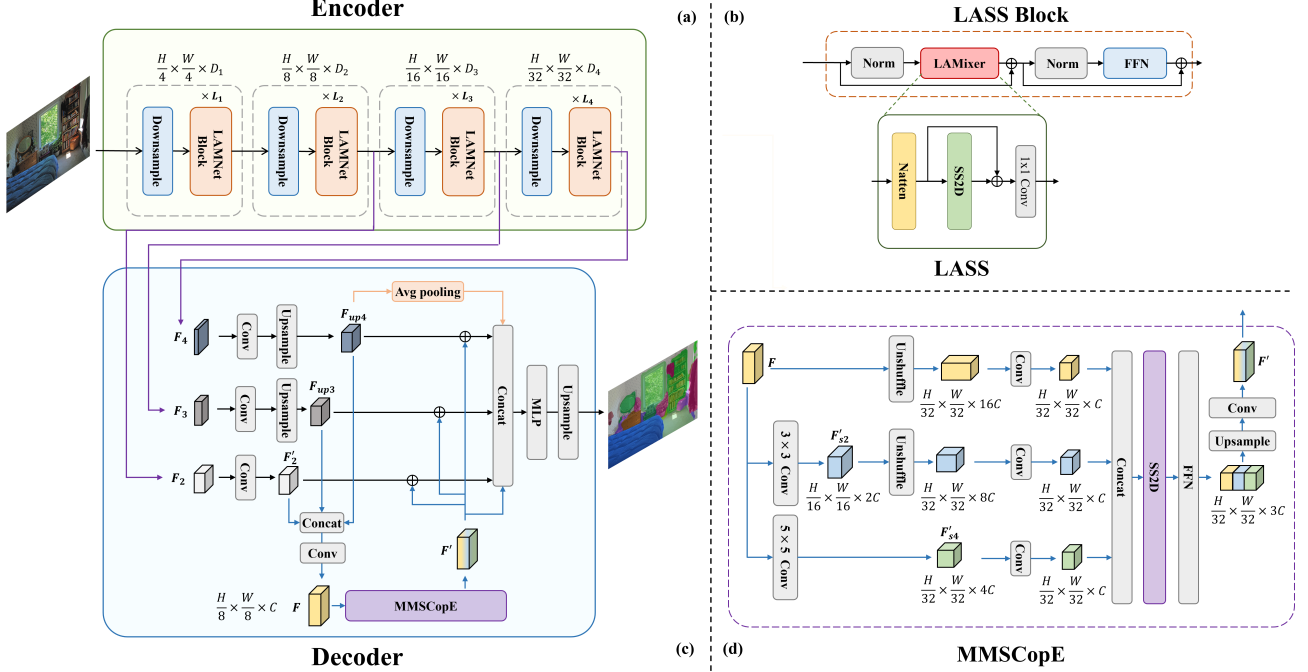
Figure 2. Overall Architecture of SegMAN. (a) Hierarchical SegMAN Encoder. (b) LASS for modeling global contexts and local details with linear complexity. (c) The SegMAN Decoder. (d) The MMSCopE module, which leverages Mamba-based 2D-selective-scan (SS2D) to extract contexts at multiple scales.

(Natten [19]) and state space models (SS2D [27]) for semantic segmentation, capturing local details and global contexts simultaneously with linear time complexity. Ablation studies (Table 8) confirm the necessity of both components.

Local attention and state space module. Long-range dependency modeling is crucial for semantic segmentation as they enable comprehensive context learning and scene understanding, as demonstrated by the success of Transformer-based semantic segmentation models [30, 35, 48, 52, 56]. To efficiently learn global contexts across all network layers, we leverage the SS2D block in VMamba [27]. This variant of a dynamic state space model named Mamba [16] adopts four distinct and complementary scanning paths, enabling each token to integrate information from all other tokens in four different directions.

However, while Mamba [16] achieves linear time complexity by compressing the global context of each channel into a fixed-dimensional hidden state [16], this compression inherently leads to information loss, in particular, loss of fine-grained local spatial details, that are crucial in segmentation to accurately localize region boundaries. To this end, we utilize Neighborhood Attention (Natten) [19], where a sliding-window attention mechanism localizes every pixel's attention span to its immediate neighborhood. This approach retains translational equivalence, effectively captures local dependencies, and has linear time complexity. In practice, we serially stack Natten and SS2D, and

SegMAN Encoder	Channels	Blocks	Params (M)	FLOPs (G)
Tiny	[32, 64, 144, 192]	[2, 2, 4, 2]	3.5	0.65
Small	[64, 144, 288, 512]	[2, 2, 10, 4]	25.5	4.05
Base	[80, 160, 364, 560]	[4, 4, 18, 4]	45.0	9.94

Table 1. Configurations of the three SegMAN Encoder variants. FLOPs were measured at the 224×224 resolution.

a shortcut around SS2D is employed to merge local and global information. The merged output is then fed into a 1×1 convolution layer for further global-local feature fusion.

Network architecture. Our SegMAN Encoder produces a feature pyramid. The spatial resolution of F_i is $\frac{H}{2^{1+i}} \times \frac{W}{2^{1+i}}$, where H and W are the height and width of the input image, respectively. We design three Seg-MAN Encoder variants with different model sizes, each with computational complexity comparable to some existing semantic segmentation encoders [18, 48, 52]. As the feature map resolution at Stage 4 is ($\frac{H}{32} \times \frac{W}{32}$), global self-attention [43] becomes computationally feasible. Therefore we replace SS2D with this more powerful global context modeling mechanism. The configurations of these models are presented in Table 1.

4

3.3. Feature Decoder

Overview. As shown in Figure 2, our proposed decoder first aggregates features at various levels of abstraction (from low-level F_2 to high-level F_4) to obtain a feature map F of size $\frac{H}{8} \times \frac{W}{8}$, as in prior works [48, 49]. Specifically, each feature map F_i is projected to a lower dimension using a 1×1 convolution followed by batch normalization and ReLU, collectively denoted as 'Conv' in Figure 2. Subsequently, F_3 and F_4 are upsampled using bilinear interpolation to match the spatial dimensions of F_2. The upsampled features F_{up3} and F_{up4} are concatenated together with F_2 and passed through a Conv layer for channel dimension reduction, resulting in F with size $\frac{H}{8} \times \frac{W}{8} \times C$, where C is the number of channels. The feature map F is then processed by our proposed MMSCopE module to extract rich multi-scale context features, producing a new feature map F'. We then add the feature map F' to F_2, F_{up3}, and F_{up4} to enrich their multi-scale contexts. The final prediction pathway concatenates F' with these context-enhanced features and the global pooling result on F_{up4}, passes the combined features through a two-layer MLP, and the result is upsampled to the input resolution.

Mamba-based multi-scale context extraction. To effectively extract contexts at different scales, we propose a Mamba-based Multi-Scale Context Extraction (MMSCoPE) module. As shown in Figure 2, we apply strided convolutions to downsample the feature map F, generating F_{s2} and F_{s4} with a resolution equal to 1/2 and 1/4 of that of F, respectively. The motivation behind this is to obtain larger-scale representations through two derived feature maps: F_{s2}, which aggregates features from 3×3 neighborhoods using a convolution with stride 2, and F_{s4}, which aggregates features from 5×5 neighborhoods with stride 4. Then, motivated by the observation that Mamba scans each channel independently, we use a single Mamba scan to simultaneously extract contexts from the three feature maps F, F_{s2}, and F_{s4}. The main idea is to concatenate these feature maps along the channel dimension and perform a single Mamba scan, which is more efficient on the GPU than processing each feature map separately. However, the spatial dimensions of these feature maps are not consistent, which prevents direct concatenation. To resolve this, we employ lossless downsampling via the *Pixel Unshuffle* [37] operation to reduce the spatial dimensions of F and F_{s2} to match that of F_{s4}, which is $\frac{H}{32} \times \frac{W}{32}$. Specifically, this operation rearranges non-overlapping 4×4 patches from F and 2×2 patches from F_{s2} into their respective channel dimension, increasing the channel depth by a factor of 16 and 4 respectively, while preserving complete spatial information. The transformed feature maps retain information at their original scales. To reduce computational costs, we project each feature map to a fixed channel dimension C through 1×1 convolutions. This step assigns equal importance to the con-

text information from each scale. Then we concatenate the projected feature maps along the channel dimension, and pass the combined feature map to SS2D [27] to achieve multi-scale context extraction in a single scan. The resulting feature map is of size $\frac{H}{8} \times \frac{W}{8} \times 3C$, where the three segments along the channel dimension correspond to earlier feature maps at three distinct scales (F, F_{s2}, and F_{s4}). To facilitate context mixing across scales, we append a 1×1 convolution layer after the SS2D scan. Finally, we use bilinear interpolation to upsample the mixed features to 1/8 of the input resolution, and project the channel dimension down to C using another 1×1 convolution. This yields the new feature map F' with mixed multi-scale contexts.

Multi-scale fusion. Instead of directly using the feature map F' to predict pixel labels, we further exploit stage-specific representations from the encoder by adding F' to F_2, F_{up3}, and F_{up4}. This effectively injects multi-scale contexts into stage-specific feature maps, which inherits information from various levels of abstraction. Next, we concatenate the resulting context-enhanced feature maps from each stage, the multi-scale context feature F', and the global average pooling result on F_{up4}. For pixel-wise label prediction, the diverse concatenated features are fused through a two-layer MLP, and further bilinearly interpolated to restore the original input resolution.

4. Experiments

4.1. Setting

Datasets. Following previous works [18, 48], we firstly pre-train our SegMAN Encoder on the ImageNet-1K dataset [9]. Then, we evaluate the semantic segmentation performance on three commonly used datasets: ADE20K [57], a challenging scene parsing benchmark consisting of 20,210 images annotated with 150 semantic categories; Cityscapes [8], an urban driving dataset comprising 5,000 high-resolution images with 19 semantic categories; and COCO-Stuff-164K [1], which contains 164,000 images labeled across 172 semantic categories.

Implementation details. We conducted ImageNet pre-training using the Timm library [45] with hyperparameters and data augmentation settings identical to those of Swin Transformer [28]. Regarding semantic segmentation, we trained all models using the MMSegmentation library [7], combining the pre-trained SegMAN Encoder with a randomly initialized decoder. Following the Segformer training protocol [48], we employed standard data augmentation techniques, including random horizontal flipping and random scaling (ratio of 0.5 to 2.0). We implemented dataset-specific random cropping with dimensions of 512×512 for ADE20K and COCO-Stuff, and 1024×1024 for Cityscapes. We trained the models using the AdamW optimizer for 160K iterations with batch sizes of 16 for ADE20K and

Method	Params(M)	ADE20K		Cityscapes		COCO-Stuff	
		GFLOP	mIoU	GFLOP	mIoU	GFLOP	mIoU
Segformer-B0 [48]	3.8	8.4	37.4	125.5	76.2	8.4	35.6
SegNeXt-T [18]	4.3	7.7	41.1	61.6	78.9	7.7	38.7
VWFormer-B0 [49]	3.7	5.8	38.9	112.4	77.2	5.8	36.2
EDAFormer-T [52]	4.9	5.8	42.3	151.7	78.7	5.8	40.3
CGRSeg-T† [32]	9.4	4.8	42.5	65.5	78.3	4.8	40.4
SegMAN-T	6.4	6.2	**43.0**	52.5	**80.3**	6.2	**41.3**
Swin UperNet-T [28]	60.0	236.0	44.4	-	-	-	-
ViT-CoMer-S [46]	61.4	296.1	46.5	-	-	-	-
OCRNet [53]	70.5	164.8	45.6	-	-	-	-
Segformer-B2 [48]	27.5	62.4	46.5	717.1	81.0	62.4	44.6
MaskFormer [5]	42.0	55.0	46.7	-	-	-	-
Mask2Former [6]	47.0	74.0	47.7	-	-	-	-
SegNeXt-B [18]	27.6	34.9	48.5	279.1	82.6	34.9	45.8
FeedFormer-B2 [38]	29.1	42.7	48.0	522.7	81.5	-	-
VWFormer-B2 [49]	27.4	46.6	48.1	415.1	81.7	46.6	45.2
EDAFormer-B [52]	29.4	32.1	49.0	605.9	81.6	32.1	45.9
CGRSeg-T† [32]	35.7	16.5	47.3	199.7	80.2	16.5	45.2
SegMAN-S	29.4	25.3	**51.3**	218.4	**83.2**	25.3	**47.5**
Swin UperNet-B [28]	121.0	261.0	48.1	-	-	-	-
ViT-CoMer-B [46]	144.7	455.4	48.8	-	-	-	-
Segformer-B3 [48]	47.3	79.0	49.4	962.9	81.7	79.0	45.5
SegNeXt-L [18]	48.9	70.0	51.0	577.5	83.2	70.0	46.5
VWFormer-B3 [49]	47.3	63.3	50.3	637.1	82.4	63.3	46.8
SegMAN-B	51.8	58.1	**52.6**	479.0	**83.8**	58.1	**48.4**

Table 2. Comparison with state-of-the-art semantic segmentation models on ADE20K, Cityscapes, and COCO-Stuff-164K. GFLOPs is calculated at 512x512 resolution for ADE20K and COCO-Stuff, and at 2048x1024 resolution for Cityscapes.

COCO-Stuff, and 8 for Cityscapes. We evaluated segmentation performance using mean intersection over union (mIoU) and computed FLOPs using the fvcore library [11]. All models were trained on 8 NVIDIA H100 GPUs.

4.2. Comparison with state-of-the-art methods

This section presents a comprehensive comparison of our proposed SegMAN with state-of-the-art segmentation methods on three widely used datasets.

Baselines. We compare our method with other strong competitors including EDAFormer [52], VWFormer [49] and CGRSeg [32], Segformer [48], and SegNeXt [18]. Note that the Efficientformer-v2 [24] used in CGRSeg [32] significantly benefits from knowledge distillation from RegNetY-16GF [34] (83.6M parameters, 15.9 GFLOPs) and extended training duration of 450 epochs compared to the conventional 300 epochs, providing a stronger encoder network compared to other methods. To ensure a fair comparison, we retrained Efficientformer-v2 on ImageNet-1K using the same training configurations as Swin [28]. We report results for CGRSeg by fine-tuning this retrained encoder following official implementations.

Results. Table 2 shows the superior performance of Seg-MAN across all three datasets: ADE20K, Cityscapes, and COCO-Stuff-164K. Among lightweight models, SegMAN-T achieves the best performance, with 43.0% mIoU on ADE20K, 80.3% mIoU on Cityscapes, and 41.3% mIoU on COCO-Stuff, while maintaining comparable computational costs. Our SegMAN-S model achieves significant performance improvements compared to state-of-the-art methods.

On the ADE20K benchmark, it surpasses the recently introduced EDAFormer-B [52] by 2.3% mIoU, and even outperforms the larger SegNeXt-L [18] model, which has 48.9M parameters, by 0.3% mIoU. The performance gains extend to the COCO-Stuff dataset, where SegMAN-S demonstrates improvements of 1.6% mIoU over EDAFormer-B and 1.0% mIoU over SegNeXt-L. Notably, SegMAN-S achieves these superior results while reducing computational complexity by over 20%, requiring only 25.3 GFLOPs compared to EDAFormer-B. On Cityscapes, SegMAN-S achieves 83.2% mIoU, surpassing EDAFormer-B by 1.6%. Moreover, SegMAN-B surpasses previous best performing models by 1.6% on both ADE20K and COCO-Stuff while incurring lower computational cost. On Cityscapes, SegMAN-B improves SOTA by 0.6% mIoU using 17% less GFLOPs. The consistent performance gains across different datasets and model scales validate the effectiveness of our proposed Seg-MAN, which can simultaneously capture global contexts, local details, and multi-scale information through its LASS-based encoder and MMSCopE-based decoder.

Speed Analysis. We benchmark inference speed on Cityscapes [8] (2048×1024 resolution) using an NVIDIA L40S GPU, averaging Frames Per Second (FPS) over 128 steps with batch size 2. As shown in Table 3, SegMAN-T achieves +1.6% mIoU at approximately 3 times the speed of EDAFormer-T, highlighting an excellent balance between performance and speed.

Model	Params (M)	GFLOPs	FPS	mIoU
Segformer-B0 [48]	3.8	125.5	21.8	76.2
SegNext-T [18]	4.3	61.6	25.8	78.9
VWFormer-B0 [49]	3.7	112.4	21.1	77.2
EDAFormer-T [52]	4.9	151.7	12.7	78.7
CGRSeg-T [32]	9.3	65.5	27.6	78.3
SegMAN-T	6.4	52.5	34.9	**80.3**
Segformer-B2 [48]	27.5	717.1	10.2	81.0
SegNext-B [18]	27.6	279.1	14.1	82.6
VWFormer-B2 [49]	27.4	415.1	9.87	81.7
EDAFormer-B [52]	29.4	605.9	6.6	81.6
CGRSeg-B [32]	35.6	199.0	11.8	80.2
SegMAN-S	29.4	218.4	13.8	**83.2**
Segformer-B3 [48]	46.3	962.9	6.7	81.7
SegNext-L [18]	48.9	577.5	10.1	83.2
VWFormer-B3 [49]	47.3	637.1	6.3	82.4
SegMAN-B	51.8	479.0	7.2	**83.8**

Table 3. Comparison of model complexity and inference speed.

4.3. Comparison of different feature encoders

In this section, we evaluate the classification performance of our proposed SegMAN Encoder on ImageNet-1K [9], comparing against two categories of architectures: state-of-the-art segmentation encoders (MiT [48], MSCAN [48], EFT [52]) and representative vision backbones designed for dense prediction tasks (Swin Transformer [28], PVT [44], ConvNeXt [29]). As demonstrated in Table 4, SegMAN Encoder variants consistently achieve superior classification accuracy in comparison to architectures of similar computational complexity. Specifically, SegMAN-S Encoder significantly surpasses ConvNeXt-T and MSCAN-B by 1.9% and 1.0% in Top-1 accuracy, respectively. The larger SegMAN-B Encoder variant demonstrates similar improvements over Swin-S and MSCAN-L by 1.9% and 1.2%, respectively. The promising performance across model variants demonstrates the benefits of integrating local attention with state space models to effectively capture both fine-grained details and global contextual information.

4.4. Ablation studies on SegMAN decoder

To validate our architectural design choices, we conduct ablation studies on the ADE20K dataset [57], examining both the overall effectiveness of our proposed decoder and the performance of individual components of the MMSCopE module. GFLOPs and FPS are measured at 512×512 resolution. For a fair comparison, we ensure that different model variants have comparable computational complexity by adjusting the number of channels. Here, all ablations are conducted using the SegMAN-S decoder.

Comparison of decoders. Table 5 shows a performance comparison of segmentation decoders. For a fair comparison, all models here share the same feature encoder, SegMAN-S Encoder, and we adopt decoders from segmentation model variants [18, 32, 48, 49, 52] with model size

Models	Params (M)	GFLOPs	Acc
MiT-B0 [48]	3.8	0.60	70.5
EFT-T [52]	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 [48]	24.2	4.0	81.6
EFT-B [52]	26.4	4.2	82.4
MSCAN-B [18]	26.8	4.4	83.0
Swin-T [28]	28.0	4.5	81.2
PVT-S [44]	24.5	3.8	79.8
ConvNeXt-T [29]	28.0	4.5	82.1
SegMAN-S Encoder	25.5	4.1	**84.0**
MiT-B3 [48]	45.2	6.9	83.1
MSCAN-L [18]	45.2	9.1	83.9
Swin-S [28]	50.0	8.7	83.2
PVT-M [44]	44.2	6.7	81.2
ConvNeXt-S [29]	50.0	8.7	83.1
SegMAN-B Encoder	45.0	9.9	**85.1**

Table 4. Comparison of classification accuracy and computational complexity (GFLOPs at 224×224 resolution) of encoder architectures on ImageNet-1K.

Model	Params (M)	GFLOPs	mIoU (%)
UPerNet [47]	27.1	38.7	50.4
MLP Decoder [48]	25.0	32.7	50.3
Ham Decoder [18]	25.3	29.5	50.4
VWFormer Decoder [49]	28.2	40.7	50.5
CGRHead [32]	44.2	24.5	50.3
EDAFormer Decoder [52]	28.5	26.2	50.2
SegMAN-S Decoder	29.4	**25.3**	**51.3**

Table 5. Comparison of complexity and accuracy of segmentation decoders. FLOPS and FPS are measured at 512×512 resolution.

similar to that of SegMAN-S, and adjusted UPerNet's channel dimension to match comparable GFLOPs. Notably, our proposed decoder demonstrates superior segmentation accuracy.

Ablation studies on the MMSCopE module. Here, we investigate the design choices of the MMSCopE module in our decoder. Table 6 presents the impact of various components within the MMSCopE module: (1) Switching to independently scanning each feature map instead of our proposed simultaneous scanning strategy significantly reduces FPS from 128 to 65; (2) The inclusion of multi-scale features yields consistent performance gains. Specifically, removing the feature map at 16x16, 32x32, or 64x64 resolution inside the MMSCopE module results in sub-optimal mIoU, with the removal of the 64x64 feature map resulting in the largest performance drop of 0.7%. Additionally, using the 64x64 feature map alone without pixel unshuffling and compression leads to a 0.4% decline. (3) Channel compression before SS2D proves beneficial, as its removal reduces mIoU by 0.3%. Alternative channel scaling configuration (1C, 2C, 4C) for the three feature scales yields suboptimal results; (4) The FFN following SS2D provides a marginal benefit of 0.1%.

Model Variant	Params (M)	GFLOPs	FPS	mIoU
SegMAN-S	29.4	25.3	128	51.3
Scan Strategy				
Independent scans	28.7	25.9	65 (-63)	51.5 (+0.2)
Multi-scale feature				
w/o 16×16	28.9	24.8	130 (+2)	51.0 (-0.3)
w/o 32×32	28.5	24.5	132 (+4)	50.8 (-0.5)
w/o 64×64	28.3	24.3	131 (+3)	50.6 (-0.7)
64×64 only	26.2	23.6	137 (+9)	50.9 (-0.4)
Compression				
w/o compression	30.3	26.7	98 (-30)	51.0 (-0.3)
Channels 1C:2C:4C	31.2	25.3	124 (-4)	50.8 (-0.5)
Architecture Components				
w/o FFN	28.4	24.8	129 (+1)	51.2 (-0.1)

Table 6. Ablation studies on the MMSCopE module in SegMAN decoder. Computational complexity (GFLOPs) and speed (FPS) measured at 512×512 resolution.

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 7. Effect of fusion strategies in the SegMAN decoder.

Model Variant	Params (M)	GFLOPs	Acc	mIoU
SegMAN-S Encoder	25.5	4.0	84.0	51.3
Replace LASS				
MaxViT [42]	24.6	3.9	83.5 (-0.5)	47.2 (-4.1)
ACMix [33]	24.9	3.8	83.1 (-0.9)	48.6 (-2.7)
PVT [44]	29.5	3.7	82.8 (-1.2)	49.1 (-2.2)
BiFormer [58]	25.2	4.1	82.9 (-1.1)	48.8 (-2.5)
LASS Design				
Parallel Natten and SS2D	25.5	4.1	83.8 (-0.2)	48.9 (-2.4)
w/o SS2D residual	26.1	4.1	83.7 (-0.3)	50.2 (-1.1)
Stage 4 Attn → SS2D	30.0	4.1	83.7 (-0.3)	50.6 (-0.7)
SS2D within LASS				
Replace SS2D w SRA [44]	26.0	4.2	83.7 (-0.3)	50.6 (-0.7)
Replace SS2D w L-Attn [22]	27.6	4.1	83.6 (-0.4)	49.5 (-1.8)
Remove SS2D	25.0	3.9	83.6 (-0.4)	47.4 (-3.9)
Natten within LASS				
Replace Natten w Conv	25.8	3.8	83.1 (-0.9)	49.5 (-1.8)
Replace Natten w Swin [28]	30.0	4.1	83.5 (-0.5)	50.3 (-1.0)
Remove Natten [19]	24.9	4.1	83.6 (-0.4)	49.8 (-1.5)

Table 8. Ablation studies and comparison of token mixers for SegMAN encoder. Parameters, FLOPs and FPS are calculated or measured at 224×224 resolution using the encoder.

Effect of feature fusion in the decoder. Table 7 illustrates the importance of our feature fusion approach (Figure 2(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.

4.5. Ablation studies on SegMAN encoder

This section presents a systematic investigation of SegMAN Encoder's architectural design choices through experiments on both ImageNet-1K [9] and ADE20K [57] using the SegMAN-S Encoder. The LASS token mixer stacks Neighborhood Attention (Natten) [19] and 2D-Selective-Scan (SS2D) in series with a residual connection added to the SS2D module to merge global and local contexts. And global attention instead of SS2D is employed in Stage 4 blocks.

To validate these architectural decisions, we conduct comprehensive ablation studies examining: (1) compara-tive performance against alternative token mixers, (2) design choices of LASS, and (3) the impact of individual components within the token mixer. In all experiments, including token mixer replacements, we maintain consistent model complexity by adjusting channel dimensions to achieve comparable parameter counts and computational costs. Table 8 presents our findings.

Alternative token mixers. We evaluate the effectiveness of LASS by replacing it with recent token mixers (MaxViT [42], ACMix [33], PVT [44], and BiFormer [58]). Note that we implement MaxViT's Block-SA and Grid-SA modules in series to serve as a token mixer. As shown in Table 8, our LASS achieves consistent and substantial improvements across both classification and segmentation tasks while maintaining competitive computational costs. The performance gains stem from the complementary nature between Natten, which models fine-grained local features, and SS2D, which efficiently models global contexts.

Alternative LASS mixer designs. We explore alternative methods to integrate SS2D and Natten together in LASS. Results indicate that parallel arrangement of Natten and SS2D and removing the residual connections for SS2D lead to substantial performance drops in both classification accuracy and segmentation mIoU.

SS2D and Natten. We analyze the two key components of LASS through systematic ablations. When SS2D is replaced with alternative global attention mechanisms, performance consistently decreases: SRA [44] (-0.3% accuracy, -0.7% mIoU) and Linear Attention [22] (-0.4% accuracy, -1.8% mIoU). Likewise, replacing Natten with either convolutions (-0.9% accuracy, -1.8% mIoU) or Shifted-Window Attention [28] (-0.5% accuracy, -1.0% mIoU) leads to sig-

nificant performance degradation, demonstrating the effectiveness of our component choices.

5. Conclusion

This work presents SegMAN, a novel architecture for semantic segmentation that advances the state-of-the-art through two key innovations: (1) SegMAN Encoder, a hierarchical encoder using LASS modules that synergistically integrates local attention with state space models, and (2) SegMAN Decoder, which incorporates MMSCopE for efficient multi-scale context extraction using state space models. The encoder captures both local details and global context, while the decoder processes multi-scale features with adaptive scaling based on input resolution. Comprehensive evaluations across multiple benchmarks demonstrates that our approach achieves state-of-the-art performance in semantic segmentation.

A. Appendix

A.1. Additional ablation studies

We present additional ablation studies on the MMSCopE module (refer to Figure 2 (d) in the main paper) using the ADE20K [57] dataset. Specifically, we investigate the effects 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 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 [37], 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 SS2D's learning capacity, thereby negatively impacting performance.

formance.

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 51.3% to 50.3%. 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.

Model Variant	Params (M)	GFLOPs	mIoU
SegMAN-S	29.4	25.3	51.3
Scan Resolution			
$\frac{W}{16} \times \frac{W}{16}$ (32×32)	30.1	25.8	50.5 (-0.8)
$\frac{W}{8} \times \frac{W}{8}$ (64×64)	29.4	27.0	50.0 (-1.3)
Downsample Method			
Bilinear Interpolation	29.4	25.2	50.3 (-1.0)

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

A.2. Qualitative examples

We present a qualitative analysis comparing the semantic segmentation results of our SegMAN-S model against similarly-sized state-of-the-art baseline models on ADE20K in Figure 3, Cityscapes in Figure 4, and COCO-Stuff-164K in Figure 5. For COCO-Stuff, comparisons are made with VWFormer and EDAFormer only, since the checkpoints for other segmentation models have not been 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 contexts, which is unattainable by existing approaches.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 2, 5
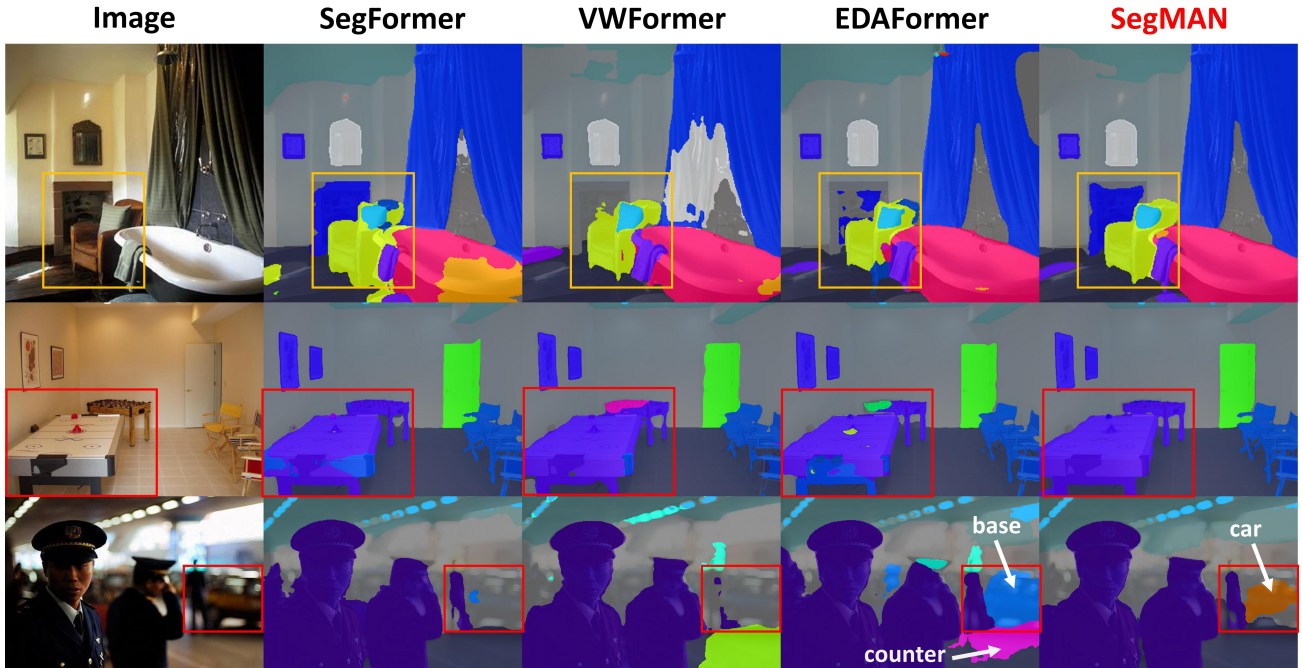
Figure 3. **Qualitative results on ADE20K.** All models have a similar size as SegMAN-S. Zoom in for best view.
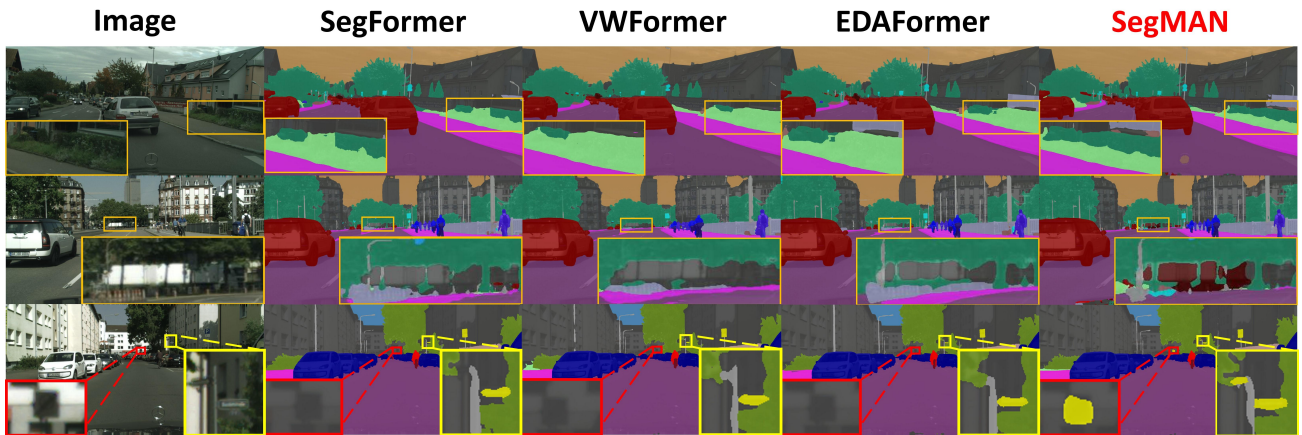

Figure 4. **Qualitative results on Cityscapes.** All models have a similar size as SegMAN-S. Zoom in for best view.

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 1, 2, 3

[3] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *ArXiv*, abs/1706.05587, 2017.

[4] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 3

[5] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. *Advances in neural information processing systems*, 34:17864–17875, 2021. 6

[6] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022. 6

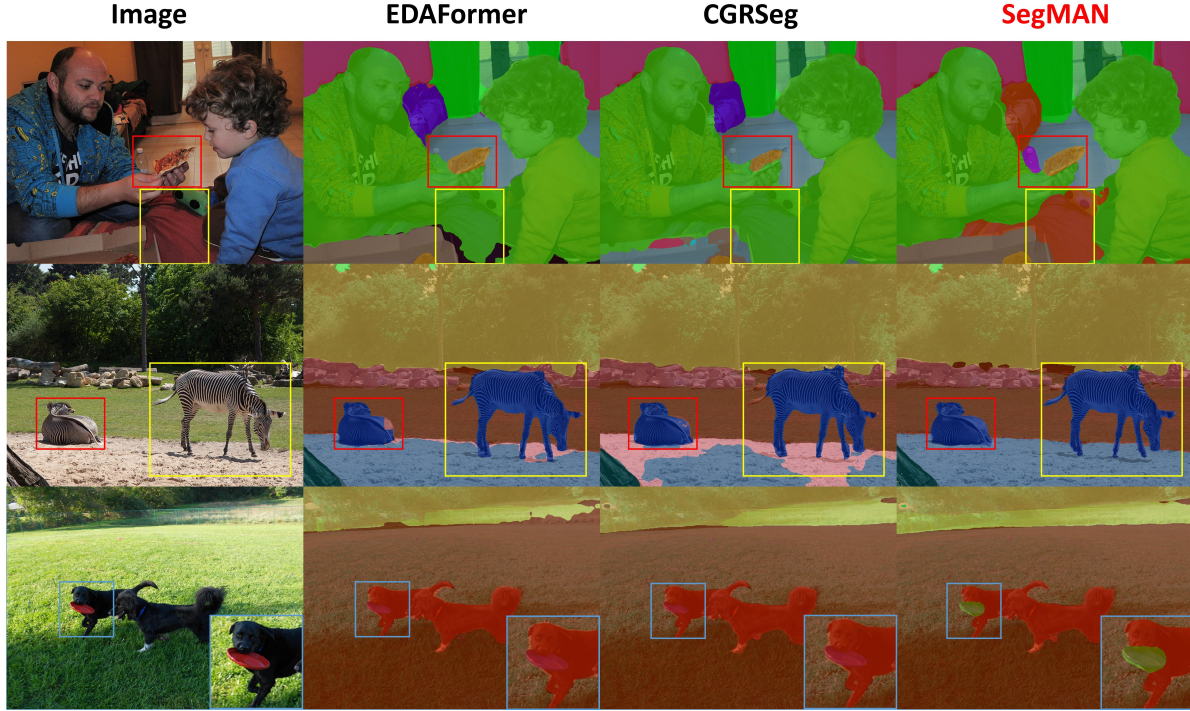[7] MMSegmentation Contributors. MMSegmentation:

Figure 5. **Qualitative results on COCO-Stuff-164K.** All models have a similar size as SegMAN-S. We do not compare with SegFormer and VWFormer as their COCO-Stuff checkpoints are not released. Zoom in for best view.

Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 5

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 5, 6

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5, 7, 8

[10] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[11] FAIR. fvcore. https://github.com/facebookresearch/fvcore, 2021. 6

[12] Zhengcong Fei, Mingyuan Fan, Changqian Yu, and Junshi Huang. Scalable diffusion models with state space backbone. *arXiv preprint arXiv:2402.05608*, 2024. 3

[13] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3146–3154, 2019. 1

[14] Yunxiang Fu, Chaoqi Chen, and Yizhou Yu. Lamamba-diff: Linear-time high-fidelity diffusion models based on local at-tention and mamba. *arXiv preprint arXiv:2408.02615*, 2024. 3

[15] Zhengyang Geng, Meng-Hao Guo, Hongxu Chen, Xia Li, Ke Wei, and Zhouchen Lin. Is attention better than matrix decomposition? *arXiv preprint arXiv:2109.04553*, 2021. 3

[16] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 3, 4

[17] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021. 3

[18] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. *Advances in Neural Information Processing Systems*, 35:1140–1156, 2022. 3, 4, 5, 6, 7

[19] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6185–6194, 2023. 2, 3, 4, 8

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[21] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classifica-

tion with convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 558–567, 2019. 3

[22] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020. 8

[23] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6399–6408, 2019. 3

[24] Yanyu Li, Ju Hu, Yang Wen, Georgios Evangelidis, Kamyar Salahi, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Rethinking vision transformers for mobilenet size and speed. In *Proceedings of the IEEE international conference on computer vision*, 2023. 6

[25] Jiarun Liu, Hao Yang, Hong-Yu Zhou, Yan Xi, Lequan Yu, Yizhou Yu, Yong Liang, Guangming Shi, Shaoting Zhang, Hairong Zheng, et al. Swin-umamba: Mamba-based unet with imagenet-based pretraining. *arXiv preprint arXiv:2402.03302*, 2024. 3

[26] Jiuming Liu, Ruiji Yu, Yian Wang, Yu Zheng, Tianchen Deng, Weicai Ye, and Hesheng Wang. Point mamba: A novel point cloud backbone based on state space model with octree-based ordering strategy. *arXiv preprint arXiv:2403.06467*, 2024. 3

[27] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 2, 3, 4, 5

[28] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 5, 6, 7, 8

[29] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 3, 7

[30] Chenyang Lu, Daan de Geus, and Gijs Dubbelman. Content-aware token sharing for efficient semantic segmentation with vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23631–23640, 2023. 4

[31] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542, 2021. 1

[32] Zhenliang Ni, Xinghao Chen, Yingjie Zhai, Yehui Tang, and Yunhe Wang. Context-guided spatial feature reconstruction for efficient semantic segmentation. *arXiv preprint arXiv:2405.06228*, 2024. 2, 3, 6, 7

[33] Xuran Pan, Chunjiang Ge, Rui Lu, Shiji Song, Guanfu Chen, Zeyi Huang, and Gao Huang. On the integration of self-attention and convolution. In *Proceedings of the IEEE/CVF*

conference on computer vision and pattern recognition, pages 815–825, 2022. 8

[34] Ilija Radosavovic, Raj Prateek Kosaraju, Ross Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10428–10436, 2020. 6

[35] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 3, 4

[36] Jiacheng Ruan and Suncheng Xiang. Vm-unet: Vision mamba unet for medical image segmentation. *arXiv preprint arXiv:2402.02491*, 2024. 3

[37] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016. 5, 9

[38] Jae-hun Shim, Hyunwoo Yu, Kyeongbo Kong, and Suk-Ju Kang. Feedformer: Revisiting transformer decoder for efficient semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2263–2271, 2023. 3, 6

[39] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022. 3

[40] Quan Tang, Bowen Zhang, Jiajun Liu, Fagui Liu, and Yifan Liu. Dynamic token pruning in plain vision transformers for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 777–786, 2023. 3

[41] Yao Teng, Yue Wu, Han Shi, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Dim: Diffusion mamba for efficient high-resolution image synthesis. *arXiv preprint arXiv:2405.14224*, 2024. 3

[42] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxvit: Multi-axis vision transformer. In *European conference on computer vision*, pages 459–479. Springer, 2022. 8

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[44] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021. 3, 7, 8

[45] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. 5

[46] Chunlong Xia, Xinliang Wang, Feng Lv, Xin Hao, and Yifeng Shi. Vit-comer: Vision transformer with convolutional multi-scale feature interaction for dense predictions.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5493–5502, 2024. 3, 6

[47] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. 3, 7

[48] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090, 2021. 1, 3, 4, 5, 6, 7

[49] Haotian Yan, Ming Wu, and Chuang Zhang. Multi-scale representations by varying window attention for semantic segmentation. *ICLR*, 2024. 1, 2, 3, 5, 6, 7

[50] Chenglin Yang, Yilin Wang, Jianming Zhang, He Zhang, Zijun Wei, Zhe Lin, and Alan Yuille. Lite vision transformer with enhanced self-attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11998–12008, 2022. 3

[51] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Learning a discriminative feature network for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1857–1866, 2018. 1

[52] Hyunwoo Yu, Yubin Cho, Beoungwoo Kang, Seunghun Moon, Kyeongbo Kong, and Suk-Ju Kang. Embedding-free transformer with inference spatial reduction for efficient semantic segmentation. In *European Conference on Computer Vision*, pages 92–110. Springer, 2024. 1, 2, 3, 4, 6, 7

[53] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 173–190. Springer, 2020. 6

[54] Tao Zhang, Xiangtai Li, Haobo Yuan, Shunping Ji, and Shuicheng Yan. Point could mamba: Point cloud learning via state space model. *arXiv preprint arXiv:2403.00762*, 2024. 3

[55] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1, 2, 3

[56] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021. 3, 4

[57] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 2, 5, 7, 8, 9

[58] Lei Zhu, Xinjiang Wang, Zhanghan Ke, Wayne Zhang, and Rynson WH Lau. Biformer: Vision transformer with bi-level routing attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10323–10333, 2023. 8

[59] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024. 3