# Appendix for SeMask: Semantically Masked Transformers for Semantic Segmentation

Jitesh Jain[1*] Anukriti Singh[1] Nikita Orlov[2]
Zilong Huang[1] Jiachen Li[1] Steven Walton[1] Humphrey Shi[1,2]
[1]SHI Labs @ Georgia Tech & UIUC & Oregon [2]Picsart AI Research (PAIR)

In this appendix, we first share our experimental settings used in the main paper in Appendix A and the results with SeMask on the COCO-Stuff 10k dataset in Appendix B. Then, we present additional ablation studies in Appendix C. We also provide an analysis on the SeMask's effect on the feature maps in Appendix D. Appendix E provides a qualitative comparison of SeMask-L FPN to Swin-L FPN on the COCO-Stuff 10k [7] and ADE20K [4] datasets.

## A. Experimental Settings

**Training Settings.** To fine-tune the pre-trained models on the semantic segmentation task, we employ the AdamW [8] optimizer with a base learning rate $\gamma_0$. Following the seminal work of DeepLab [1] we adopt the *poly* learning rate decay $\gamma = \gamma_0\,(1-\frac{N_{iter}}{N_{total}})^{0.9}$ where $N_{iter}$ and $N_{total}$ represent the current iteration number and the total iteration number. We use a linear warmup strategy for 1,500 iterations.

For ADE20K, we set the base learning rate $\gamma_0$ to $10^{-4}$, weight decay to $10^{-4}$ and train for $80K$ iterations with a batch size of 16.

For Cityscapes, we set $\gamma_0$ to $10^{-3}$, a weight decay of $5 \times 10^{-2}$ and train for $80K$ iterations with a batch size of 8.

**Inference.** To handle varying image sizes during inference, we keep the aspect ratio intact and resize the image to a resolution with the smaller edge resized to the training resolution and consequently rescaled to the original dimensions before calculating the metric score. For multi-scale inference, following standard practice [2] we use rescaled versions of the image with scaling factors of $(0.5, 0.75, 1.0, 1.25, 1.5, 1.75)$.

## B. Experiments on COCO-Stuff 10k

COCO-Stuff 10k comprises of a total of 10k images with dense pixel-level annotations, selected from the COCO [7] dataset. The training set contains 9k images with 171 semantic classes and the test set contains 1k images.

We set the base learning rate $\gamma_0$ to $10^{-4}$, weight decay to $10^{-4}$ and train for 80K iterations with a batch size of 16.

We provide our experimental results in Tab. I. Our SeMask framework shows impressive improvement on the COCO-Stuff 10k dataset proving its dataset-agnostic ability.

| Method | Backbone | Crop Size | #Param. (M) | s.s. mIoU (%) | m.s. mIoU (%) |
|---|---|---|---|---|---|
| Swin-T FPN | Swin-T | 512×512 | 33 | 37.14 | 38.37 |
| SeMask-T FPN | SeMask Swin-T | 512×512 | 35 | **37.53** (+0.39) | **38.88** (+0.55) |
| Swin-S FPN | Swin-S | 512×512 | 54 | 40.53 | 41.91 |
| SeMask-S FPN | SeMask Swin-S | 512×512 | 56 | **40.72** (+0.19) | **42.27** (+0.36) |
| Swin-B FPN | Swin-B$^{\dagger}$ | 512×512 | 54 | 44.18 | 45.79 |
| SeMask-B FPN | SeMask Swin-B$^{\dagger}$ | 512×512 | 56 | **44.68** (+0.50) | **46.30** (+0.51) |
| Swin-L FPN | Swin-L$^{\dagger}$ | 640×640 | 204 | 46.42 | 48.13 |
| SeMask-L FPN | SeMask Swin-L$^{\dagger}$ | 640×640 | 211 | **47.47** (+1.05) | **48.54** (+0.41) |

Table I: **Experiments with COCO-Stuff 10k.** We provide a comparison of using SeMask Swin with Semantic-FPN [6] decoder on the COCO Stuff-10k test set. We evaluate the models using both, the *single scale (s.s)* and *multi-scale (m.s.)* mIoU (↑).

## C. Additional Ablations

**Tuning the hyperparameter** $\alpha$We weigh the loss ($\mathcal{L}_2$) calculated on the semantic-prior prediction with a hyperparameter $\alpha$ as formulated in Eq. (1). Using weighted supervision for the semantic-prior maps is critical so that the model treats the semantic context as an additional signal for feature modeling and not as the main prediction.

$$\mathcal{L}_{\mathcal{T}} = \mathcal{L}_1 + \alpha\mathcal{L}_2 \tag{1}$$

We study the impact of $\alpha$ on performance in Tab. II by changing the values of $\alpha$ on the Swin-Tiny variant. $\alpha = 0.4$ is the optimum setting for modeling the network's image feature level and semantic level context.

**Pretraining Dataset.** We study the impact of the pretraining dataset (ImageNet-1k v/s ImageNet-22k) on performance in Tab. III by training and evaluating the *Base* variant pretrained on various settings. Our framework is agnostic to the pretraining setting showing improvement for all combinations mainly used for the ImageNet pretraining: *(i)* ImageNet-1k and 224×224 image resolution; *(ii)* ImageNet-

[*] Work partially done during JJ's internship at PAIR.

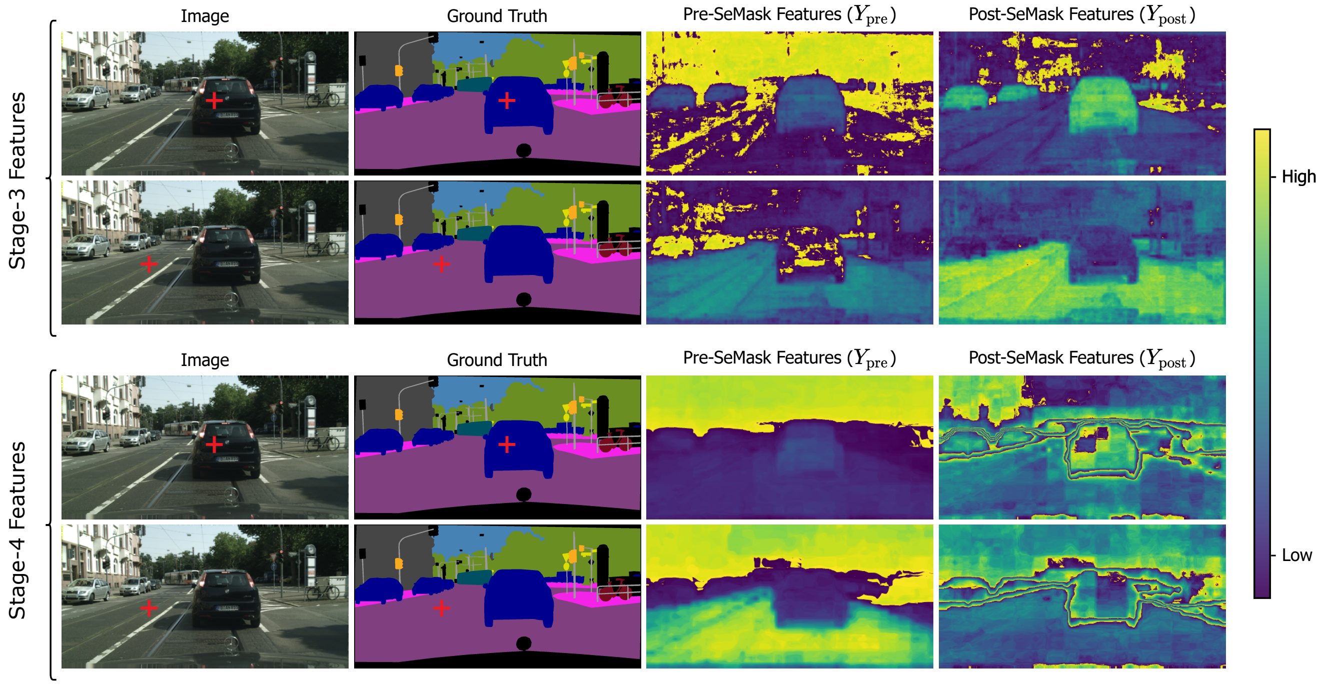
Image
Ground Truth
Pre-SeMask Features ($Y_{\text{pre}}$)
Post-SeMask Features ($Y_{\text{post}}$)
Stage-3 Features
Stage-4 Features
High
Low

Figure I: **Analysis of features on the Cityscapes val set.** We analyze pixel-wise attention maps for the $Y_{\text{pre}}$ and $Y_{\text{post}}$ features from Stage-3 and Stage-4 of our SeMask-T FPN network. The post-SeMask ($Y_{\text{post}}$) features are richer in clear boundaries and pixel similarity than pre-SeMask ($Y_{\text{pre}}$) features.

| **Method** | **Backbone** | $\alpha$ | **mIoU (%)** | **#Param (M)** |
|---|---|---|---|---|
| SeMask-T FPN | SeMask Swin-T | 0.0 | 41.72 | 35 |
| SeMask-T FPN | SeMask Swin-T | 0.4 | **42.06** | 35 |
| SeMask-T FPN | SeMask Swin-T | 0.7 | 41.87 | 35 |
| SeMask-T FPN | SeMask Swin-T | 1 | 41.67 | 35 |

Table II: **Ablation on $\alpha$.** We experiment with different values of $\alpha$ on the SeMask-Tiny variant and report single-scale mIoU (↑). $\alpha = 0.4$ is the best setting.

| **Method** | **Backbone** | **Pre** | **Res** | **mIoU (%)** | **#Param (M)** |
|---|---|---|---|---|---|
| Swin-B FPN | Swin-B | 1k | 224 | 45.47 | 93 |
| SeMask-B FPN | SeMask Swin-B | 1k | 224 | **45.63** | 96 |
| Swin-B FPN | Swin-B | 22k | 224 | 47.65 | 93 |
| SeMask-B FPN | SeMask Swin-B | 22k | 224 | **48.29** | 96 |
| Swin-B FPN | Swin-B | 22k | 384 | 48.80 | 93 |
| SeMask-B FPN | SeMask Swin-B | 22k | 384 | **49.06** | 96 |

Table III: **Ablation on Pretraining dataset.** We compare the improvement when using the SeMask-Base variant with different pretraining settings: *ImageNet-1k $v/s$ ImageNet-22k* and 224×224 $v/s$ 384×384 and show that it is agnostic to the pretraining setting.

22k and 224×224 image resolution; and *(iii)* ImageNet-22k and 384×384 image resolution.

**Number of SeMask Blocks ($N_S$).** In Tab. IV we study the impact of number of SeMask attention blocks on performance by changing the values of $N_S$ inside each semantic layer on the Swin-Tiny variant. We observe that $N_S = [1, 1, 1, 1]$ is the best setting. Interestingly, when stacking multiple blocks in a layer, we observe that inputting the $S_Q$ from the previous SeMask block into the later one gives better performance than obtaining $S_Q$ from the features. This shows that extracting semantic features using a single semantic attention operation is the optimum setting.

| **Method** | **Backbone** | $N_S$ | **mIoU (%)** | **#Param (M)** |
|---|---|---|---|---|
| SeMask-T FPN | SeMask Swin-T | [1, 1, 1, 1] | **42.06** | 35 |
| SeMask-T FPN | SeMask Swin-T | [1, 2, 2, 2] | 40.60 | 37 |
| SeMask-T FPN | SeMask Swin-T | [2, 2, 2, 2] | 40.09 | 37 |

Table IV: **Ablation on $N_S$.** We experiment with different combinations of $N_S$ on the SeMask-Tiny variant and report mIoU (↑). $N_S = [1, 1, 1, 1]$ is the best setting.

## D. Analysis on SeMask

In order to confirm our hypothesis that adding semantic context inside the encoder with the help of the semantic attention operation helps in improving the semantic quality of the features, we analyze the pixel-wise attention quality of the intermediate features of our SeMask-T FPN model on the Cityscapes [3] val dataset as shown in Fig. I.

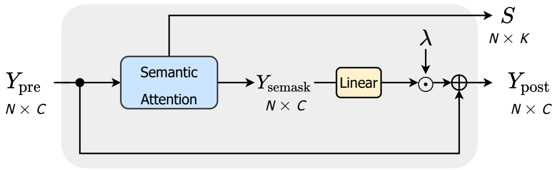

Figure II: **SeMask Block.** The semantic attention outputs the semask features ($Y_{\text{semask}}$) using the features from the transformer layer ($Y_{\text{pre}}$). We use a residual connection from $Y_{\text{pre}}$ to obtain the final output ($Y_{\text{post}}$). $S$ is the semantic-prior map used to semantically mask the features ($Y_{\text{pre}}$).

Specifically, we analyze pixel-wise attention for the pre-SeMask ($Y_{\text{pre}}$) and post-SeMask ($Y_{\text{post}}$) features (Fig. II) for Stage-3 and Stage-4 which are downsampled by $\times 16$ and $\times 32$, respectively. We calculate the pixel-wise attention maps corresponding to the target pixel (red cross sign), and we observe that post-SeMask features have more similar features for the same semantic category region with better boundaries than the pre-SeMask features. It reflects that the semantic prior maps help increase similarity between the pixels belonging to the same semantic category and improve the semantic segmentation performance.

## E. Qualitative results

We provide qualitative results on the COCO-Stuff 10k test set in Fig. III where SeMask-L FPN produces better per-pixel predictions compared to Swin-L FPN. It is evident in $(b)$ as the Swin-L FPN network fails to label the pole correctly and completely mislabels the sky region in $(c)$.

We show more qualitative results on the ADE20K validation set in Fig. IV. Swin-L FPN mislabels *mirror* as *curtain* in $(b)$ due to the reflection of the curtain. On the other hand, SeMask-L FPN classifies the regions accurately.

## F. Discussion on SeMask with CNNs

We formulate SeMask keeping in mind the feature modeling inside hierarchical vision transformers. Thus, directly incorporating SeMask into CNN backbones is unfair. This is proved by our experiments where we achieve 26.60% (s.s. mIoU) with SeMask ResNet-50 FPN compared to mIoU=37.49% (s.s. mIoU) with ResNet-50 FPN [5].

## References

[1] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. 1

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, 2018. 1

[3] 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 *CVPR*, 2016. 2

[4] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. Semantic understanding of scenes through the ade20k dataset. In *CVPR*, 2017. 1

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[6] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 1

[7] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[8] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 1

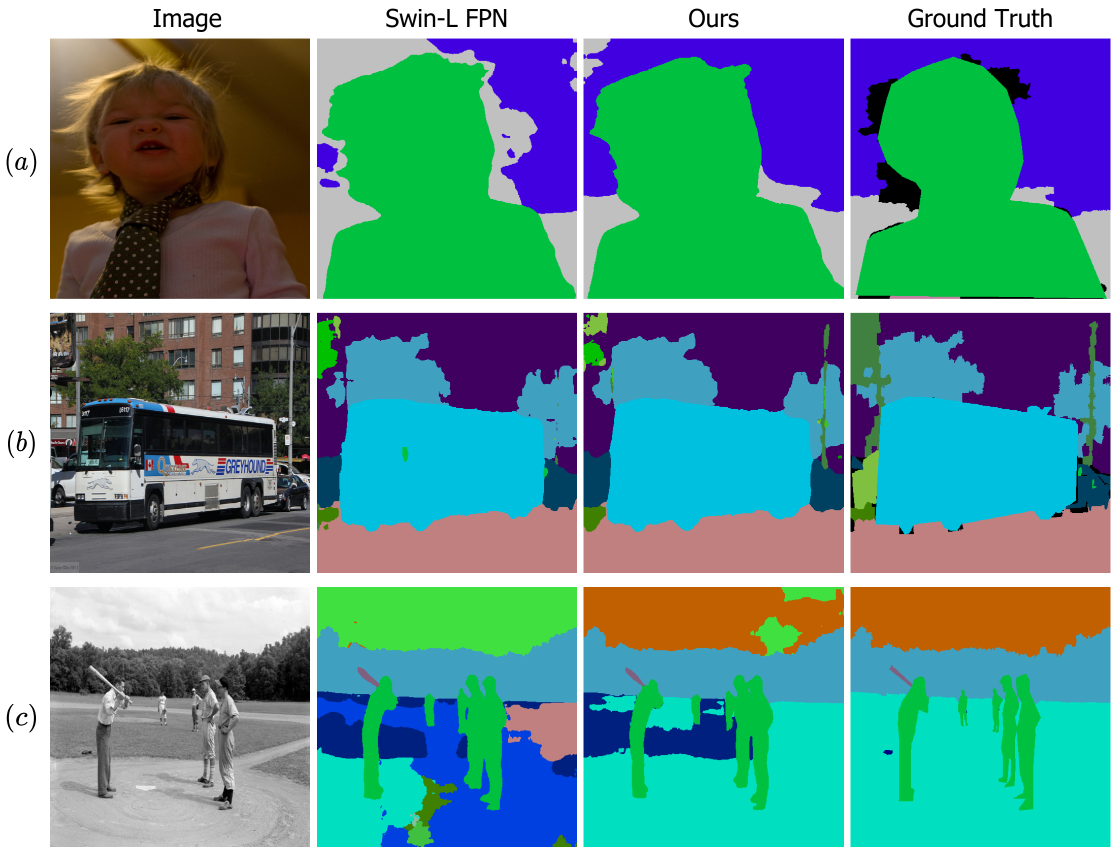

Figure III: **Qualitative results on the COCO-Stuff 10k test set.** Swin-L FPN completely mislabels the sky region and a significant part of the ground in $(c)$, and our **SeMask-L FPN** shows better accuracy in classifying the regions.

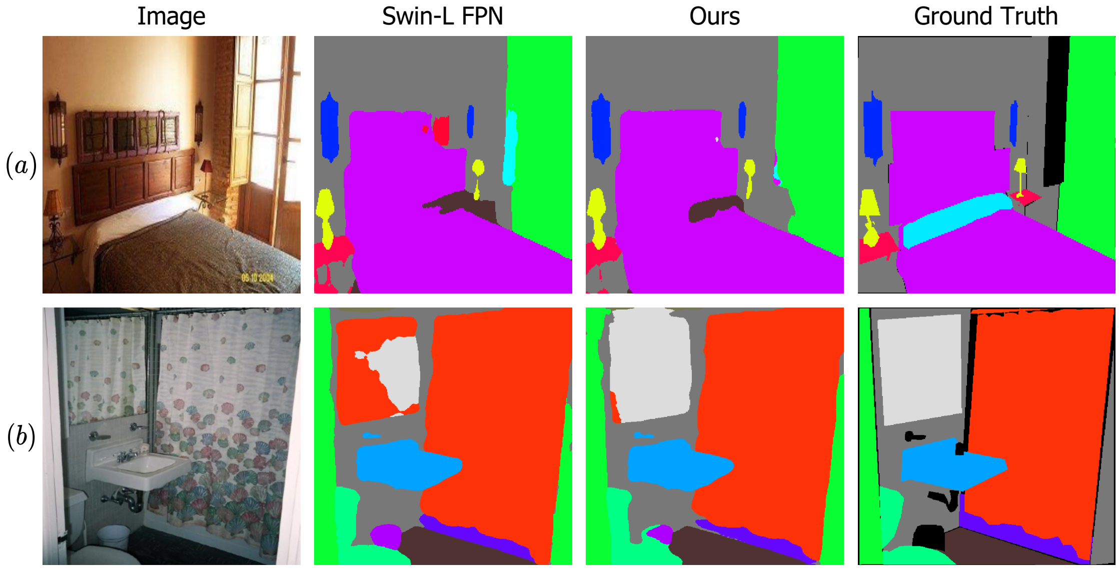

Figure IV: **Qualitative results on the ADE20K validation set.** Our **SeMask-L FPN** can correctly classify the mirror region in $(b)$, whereas Swin-L FPN mislabels a significant part of the mirror as curtain owing to the reflection.