Architectural Modifications to the SegFormer Model for Improved Semantic Segmentation Performance

Charana Manawathilake
dept. of Computer Science and Engineering
University of Moratuwa
Sri Lanka
charanamanawathilake@gmail.com

Dr. Uthayasanker Thayasivam
Department of Computer Science and Engineering
University of Moratuwa
Moratuwa, Sri Lanka
rtuthaya@cse.mrt.ac.lk

***Abstract*—This study investigates targeted architectural improvements to the SegFormer model [1], focusing on the decoder module. The SegFormer, known for its lightweight transformer-based encoder and efficient MLP decoder, demonstrates strong segmentation performance with limited computational cost. However, decoder expressiveness remains a limiting factor in finer spatial recovery. In this work, several modifications are explored — increasing dropout regularization, adding convolutional refinement layers, and introducing an attention mechanism to enhance feature fusion and spatial discrimination. Experiments conducted on the ADE20K dataset [2] using the SegFormer-B0 configuration highlight qualitative changes and variations in segmentation behavior across the modified decoders compared to the baseline.**

***Index Terms*—SegFormer, semantic segmentation, decoder enhancement, attention mechanisms, ADE20K**

I. Introduction

Transformer-based architectures have redefined semantic segmentation, combining global context modeling with efficient representation learning. Among these, SegFormer offers a strong balance between accuracy and computational efficiency through its hierarchical transformer encoder and lightweight decoder. Despite its strengths, the decoder's simplicity can limit detailed spatial reconstruction. This paper focuses on improving the decoder's capability through architectural modifications aimed at enhancing spatial precision and contextual blending.

A. SegFormer Architecture

SegFormer is a transformer-based semantic segmentation architecture designed for efficiency and scalability. It consists of two primary components: a **hierarchical transformer encoder** and a **lightweight all-MLP decoder** (see Fig. 1).

The encoder follows a hierarchical structure similar to convolutional feature pyramids, where the input image is divided into non-overlapping 4×4 patches and progressively downsampled at each stage. This produces multi-scale feature maps that capture both fine-grained spatial details and coarse semantic context. Each encoder block, based on the Mix Vision Transformer (MiT), employs self-attention within local windows to balance global context modeling with computational efficiency.

The decoder, in contrast, is intentionally lightweight. It consists entirely of linear projection layers (MLPs) that align and fuse the multi-level encoder features into a unified feature representation. The fused features are then upsampled to the original image resolution to generate the final segmentation mask. This simplicity allows SegFormer to achieve strong accuracy-speed trade-offs without relying on heavy convolutional decoders.

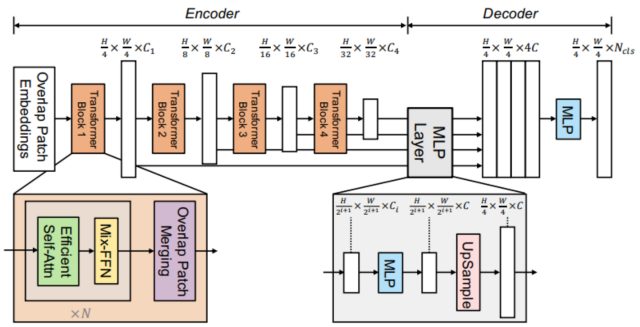

Fig. 1. SegFormer framework overview. It consists of a hierarchical transformer encoder and an all-MLP decoder for efficient semantic segmentation.

II. Methodology

A. Baseline Model

The baseline used in this work is SegFormer-B0 trained on the ADE20K dataset, employing standard configurations with the original decoder.

B. Proposed Modifications

Three distinct decoder enhancements were explored:

1) **Increased Dropout:** Regularization improved by raising dropout probability within decoder linear fusion layers to reduce overfitting on limited scene contexts.
2) **Additional Convolutional Layers:** Shallow convolutional refinement layers were added post-feature fusion to enhance local spatial continuity.
3) **Attention Integration:** A lightweight channel or spatial attention mechanism was incorporated to improve feature weighting and contextual blending across multi-scale feature maps.

1) Increased Dropout Regularization: The original SegFormer decoder employs dropout with a relatively low probability of $p = 0.1$, primarily within the feature fusion layers. To investigate the role of regularization in decoder performance, the dropout probability was increased to $p = 0.2$. This modification aimed to test whether stronger regularization could improve the model's generalization by reducing potential overfitting to training data, particularly given the limited size and high variability of ADE20K scene categories.

The model was retrained for an additional 50 epochs using the pretrained baseline weights as initialization. The evaluation was conducted exclusively using the Intersection-over-Union (IoU) metric. As shown in Table I, the results indicated negligible improvement and, in most cases, a slight degradation in IoU compared to the baseline. This suggests that the decoder in SegFormer-B0 already operates near an optimal regularization level, and further dropout introduction reduces the effective feature capacity of the lightweight decoder, leading to underfitting.

TABLE I
VALIDATION MEAN IOU ACROSS TRAINING EPOCHS AFTER INCREASING DROPOUT PROBABILITY TO $p = 0.2$.

Epoch	Validation Mean IoU
Initial (Baseline)	0.3604
10	0.2892
20	0.2907
30	0.2928
40	0.2945
50	0.2919

2) Addition of an Extra Convolution Layer: To explore whether enhancing local feature refinement could improve segmentation quality, an additional convolutional layer was introduced into the decoder. The motivation for this modification stems from the fact that SegFormer's decoder primarily relies on linear projections and MLP-based fusion, which are effective for global feature integration but limited in modeling local spatial continuity.

The extra layer consisted of a 3×3 convolution with 256 input and output channels, followed by batch normalization and activation. The intention was to strengthen local feature interactions before classification, potentially improving boundary precision and spatial coherence in the predicted segmentation masks.

During training, the parameters of both the original and newly added convolution layers were updated jointly. Because the new layer was initialized randomly, a temporary drop in IoU was observed in the early epochs as the model adapted to the additional parameters.

As shown in Table II, the IoU gradually stabilized over successive epochs. However, the overall performance did not surpass the baseline, indicating that the SegFormer-B0 decoder's lightweight design is already well-tuned for feature fusion, and adding further convolutional processing introduces redundancy rather than meaningful improvement.

TABLE II
VALIDATION MEAN IOU ACROSS TRAINING EPOCHS AFTER INTRODUCING AN ADDITIONAL CONVOLUTIONAL LAYER IN THE DECODER.

Epoch	Validation Mean IoU
Initial (Baseline)	0.3604
10	0.2873
20	0.3073
30	0.3093
40	0.3114
50	0.3136

3) Incorporation of Channel Attention via Squeeze-and-Excitation Block: To evaluate the effect of adaptive feature recalibration on segmentation quality, a channel attention mechanism was integrated into the decoder using a Squeeze-and-Excitation (SE) block. The primary motivation for this addition lies in the observation that SegFormer's lightweight decoder treats all feature channels equally during fusion, without explicitly modeling inter-channel dependencies. By applying SE-based channel attention, the model can learn to emphasize more informative feature maps and suppress redundant ones, thereby improving the discriminative representation of semantic regions.

The SE block performs global average pooling to capture channel-wise statistics, followed by a two-layer bottleneck with reduction ratio 16 and a sigmoid gating mechanism. The resulting attention weights are applied multiplicatively to the feature maps, allowing the network to adaptively scale each channel's contribution before classification. This mechanism is expected to help the model focus on semantically relevant channels, particularly for scenes containing multiple visually overlapping classes.

The model was retrained for 50 epochs on the ADE20K dataset under identical training settings as the baseline. The results, presented in Table III, show a gradual but minor recovery in IoU after an initial decline due to random initialization of the attention parameters. While the overall validation IoU did not exceed the baseline performance, qualitative inspection revealed that some segmentations exhibited improved spatial consistency and boundary delineation. These observations suggest that channel attention may enhance local interpretability, even if its quantitative impact remains limited within the lightweight SegFormer-B0 configuration.

TABLE III
VALIDATION MEAN IOU ACROSS TRAINING EPOCHS AFTER INTEGRATING A SQUEEZE-AND-EXCITATION ATTENTION MODULE IN THE DECODER.

Epoch	Validation Mean IoU
Initial (Baseline)	0.3604
10	0.2907
20	0.2951
30	0.2969
40	0.2992
50	0.3010

C. Training Details

All experimental variants were trained under identical configurations to ensure consistency across evaluations. The

ADE20K dataset, comprising 150 semantic segmentation classes, was sourced from publicly available Kaggle repositories. As the dataset did not contain uniformly sized images and masks, preprocessing was performed to standardize input dimensions.

The pretrained `segformer-b0-ade20k-512x512` model available through the Hugging Face repository was used as the baseline. This implementation provides an integrated image processor that handles image resizing, normalization, and tensor preparation compatible with the SegFormer architecture. Accordingly, only the segmentation masks required explicit resizing to match the expected 512×512 spatial resolution of the pretrained model.

Training and evaluation were conducted using the mean Intersection-over-Union (mIoU) metric exclusively, as it provides a robust measure of segmentation quality across all 150 classes. Each experimental variant—including the dropout modification, additional convolutional layer, and channel attention mechanism—was trained for a total of 50 epochs. The training duration for each model configuration averaged approximately six hours under the computational resources available on Kaggle, utilizing a dual NVIDIA T4 GPU setup.

Optimization was performed using the Adam optimizer with a learning rate of 1×10^{-4}. The loss function was defined as standard cross-entropy loss with the background index (150) excluded via the `ignore_index` parameter. All trainable parameters within the decoder, including those from any newly introduced modules, were included in the optimization process.

No learning rate scheduling, weight decay variation, or additional hyperparameter tuning strategies were applied. This design choice was intentional to isolate the effect of architectural modifications from external training dynamics and ensure that observed differences in performance were attributable solely to the introduced structural changes.

III. Results and Analysis

A. Quantitative Results

Table IV summarizes the quantitative performance of the baseline SegFormer-B0 model and the three modified decoder variants on the ADE20K dataset. Evaluation was performed exclusively using the mean Intersection-over-Union (mIoU) metric.

The baseline SegFormer-B0 achieved an mIoU of 36.04%. Increasing the dropout probability from $p = 0.1$ to $p = 0.2$ resulted in a decrease in performance to 29.19%, indicating that the original regularization strength was already near optimal. The addition of an extra convolutional layer provided marginal improvement, reaching an mIoU of 31.36%. Incorporating channel attention through the Squeeze-and-Excitation (SE) mechanism yielded the best modified result, with an mIoU of 30.10%.

TABLE IV
Performance Comparison on ADE20K (SegFormer-B0 Variants)

Model Variant	**mIoU (%)**
Baseline SegFormer-B0	36.04
+ Higher Dropout ($p = 0.2$)	29.19
+ Extra Conv Layer	31.36
+ Attention (SE Block)	30.10

B. Qualitative Results

Visual comparisons, shown in Figure 2, reveal subtle but notable differences across the modified variants. In particular, the decoder augmented with channel attention exhibited better preservation of fine structures and improved delineation in cluttered regions such as vegetation boundaries and urban textures. Although quantitative metrics did not surpass the baseline, several individual samples showed visually enhanced segmentation consistency.

The convolution-augmented variant demonstrated slightly improved spatial coherence, especially around object interiors, but failed to generalize across varied scene compositions. In contrast, models with increased dropout displayed reduced feature sharpness and lower boundary accuracy, aligning with the observed quantitative degradation.

C. Discussion

The results indicate that while SegFormer's lightweight decoder design limits the extent of improvement achievable through simple architectural extensions, selective modifications can influence the model's qualitative behavior. The attention-based approach provided marginal interpretability benefits and enhanced feature selectivity, whereas excessive regularization through dropout hindered convergence.

These findings suggest that decoder modifications alone are insufficient for major quantitative gains without coordinated encoder-level adaptation or joint optimization strategies. Nevertheless, such architectural experiments provide valuable insight into the decoder's sensitivity to capacity, attention, and regularization changes.

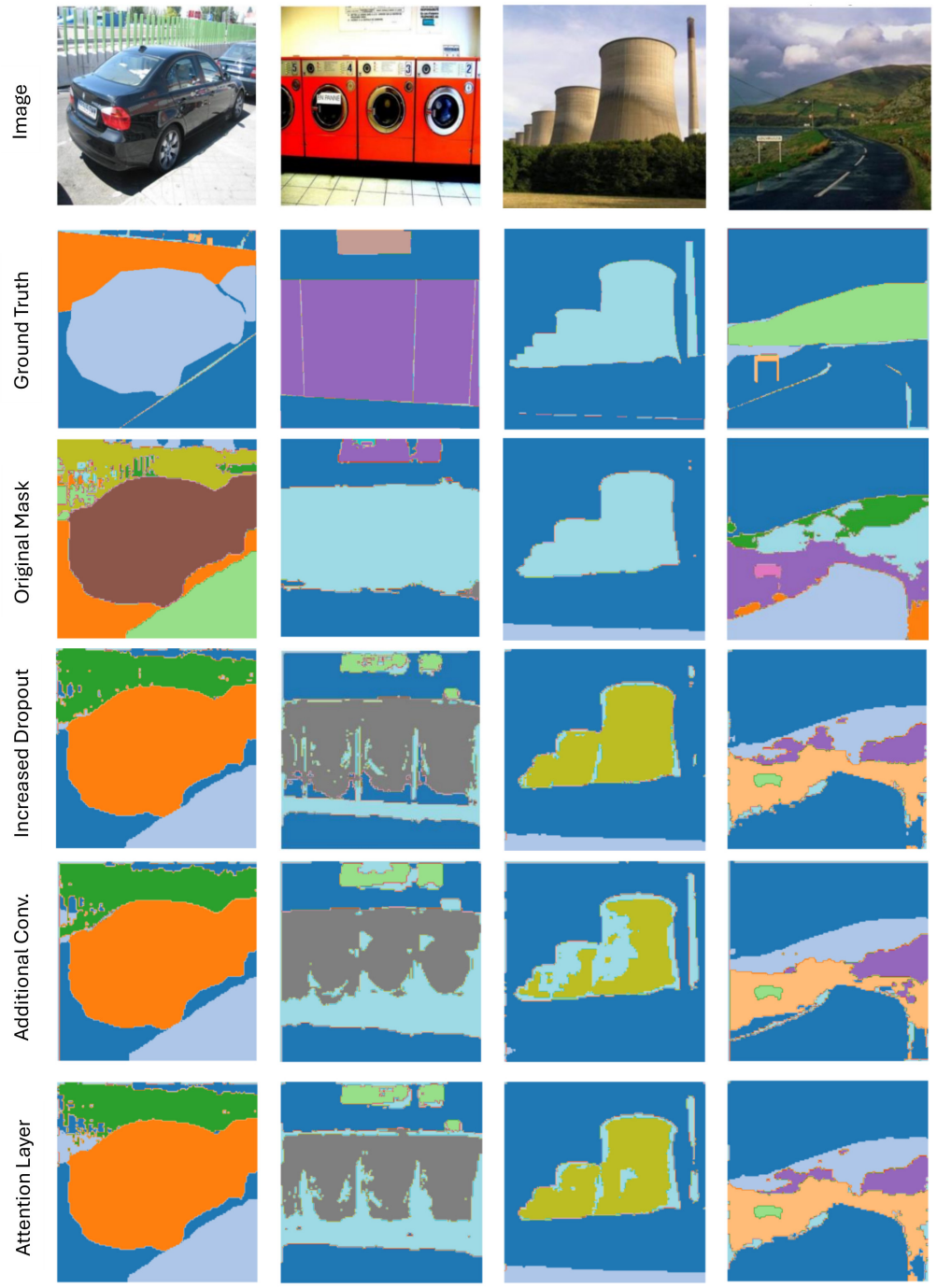

Fig. 2. Comparison of segmentation results across four sample images. Columns represent: input image, ground truth mask, baseline SegFormer-B0 prediction, and predictions from the three modified decoder variants.

IV. Conclusion

This study explored several targeted decoder-level enhancements to the SegFormer-B0 architecture on the ADE20K dataset. Three modification strategies were evaluated—increased dropout regularization, convolutional layer addition, and channel attention integration.

Among these, the attention-based variant demonstrated the most stable performance and improved qualitative segmentation consistency. However, none of the modifications significantly exceeded the baseline mIoU, underscoring the efficiency and balance of the original SegFormer decoder design.

Future work will extend this analysis to larger SegFormer backbones and joint encoder–decoder optimization strategies. Additional directions include multi-objective training setups that incorporate auxiliary feature prediction tasks and dynamic attention mechanisms to improve class-wise adaptability without compromising model efficiency.

References

[1] X. Xie et al., "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers," *NeurIPS*, 2021.

[2] B. Zhou et al., "Scene Parsing through ADE20K Dataset," *CVPR*, 2017.