SEGFORMER - FPN : BRIDGING TRANSFORMERS AND PYRAMID FOR SEGMENTATION

CONTENTS

INTRODUCTION

Semantic segmentation is a fundamental task in computer vision problems, the goal is to assign semantic labels to each of the pixels in an image, further upgrading to videos. This enables a detailed understanding of scenes, making it critical for different applications like autonomous driving, medical imaging and urban planning. Traditional semantic segmentation approaches have been relied on convolutional neural networks with encoder-decoder architectures. These methods progressively down-sample the input images in the encoder to extract the features and then up-sample them in the decoder to recover spatial details. While this is an effective method, CNN-based approaches rely on local receptive fields and struggle with long-range dependencies and global contexts, which are crucial for accurate segmentation of complex objects and multi-scale structures in real-world images.

During the advancements in the NLP field, the transformers have also shown powerful performance for computer vision tasks, leveraging their self-attention mechanism to model global dependencies efficiently. Transformer based segmentation models, such as SETR and Segmenter takes in images as a sequence of patches capturing the global context modeling from the very first layer. These models have shown state-of-the-art performance on various benchmarks like ADE20K and Pascal VOC datasets, overcoming the limitations of CNN's fixed receptive fields. Subsequently, hybrid models like Segformer combine hierarchical transformer encoder with light-weight Multi-layer perceptron decoder, which brings a balance between efficiency and performance. Although these models capture global information, they struggle with fine-grained details due to the simple decoder designed. This research project seeks to address this limitation by exploring other CNN based decoders for better segmentation and increasing the performance of the model.

Figure 1 : Semantic Segmentation

BASELINE MODEL

Segformer is a hybrid architecture with a transformer backbone and a light weight MLP for the decoder, where the transformer acts as a encoder due to its hierarchical architecture. Segformer is popular for its simplicity, scalability, and exceptional performance with various datasets. Unlike conventional model structure that rely heavily on CNN, Segformer integrates the strengths of transformer and feature extraction techniques for delivering robust multi-scale contextual understanding.

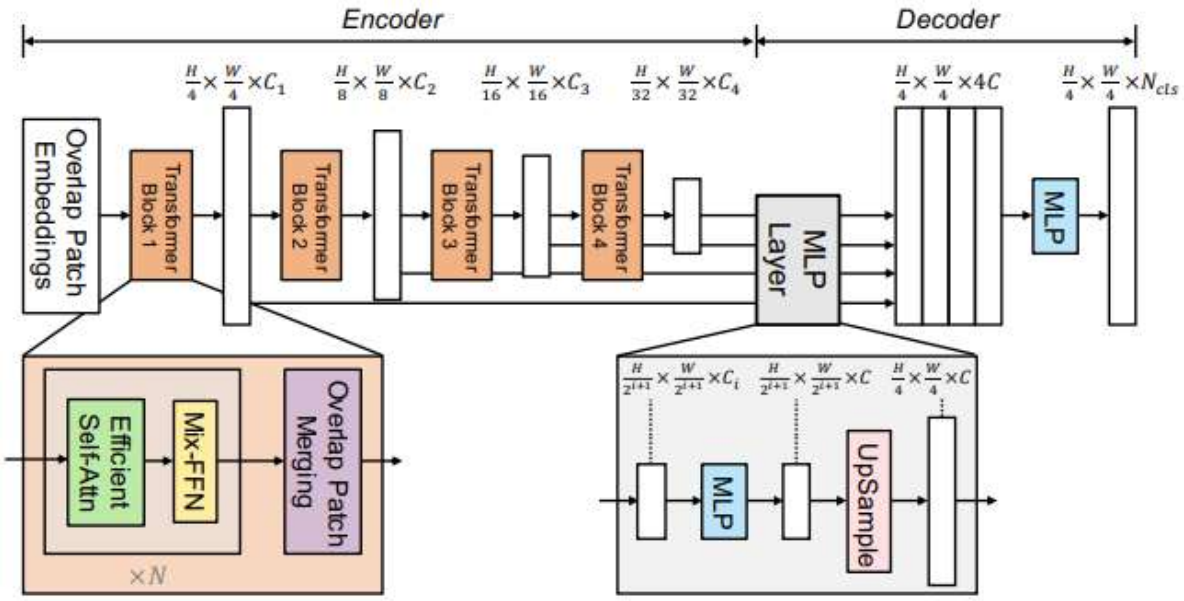

Figure 2 : Block diagram of Segformer[1]

As mentioned, the encoder is the backbone of the segformer, which extracts multi-scale features with high efficiency. It is due to the hierarchical transformer, with 4 stages in encoder, generating increasing scales of feature maps. Concatenating these feature maps and further providing these features to the decoder to segment the outputs.

The encoder mainly consists of patch embedddings, Transformer block and feature maps or channels. The patch embeddings divide the inputs into non-overlapping patches and feed it it to the 1st transformer block. The transformer block consist of Efficient Self-Attention module, a Mix-FFN and patch embeddings module, which reduces the dimensions by a factor of 4. This is repeated in 4 stages to gain the features in 4 different feature dimensions and feature maps.

In the decoder, these feature maps of different sizes are resized and concatenated and aggregated into a single representation for the segmentation head to classify each pixel to its respective class. A final linear projection transforms the aggregated features into pixel-wise class probabilities, generating the segmentation mask.

The model is trained using specific hyperparameters to achieve optimal performance. Key training hyperparameters include:

- **Learning rate:** 5e-5
- **Weight decay:** 0.01
- **Training iterations:** 20,000
- **Batch size:** 4

Additionally, the process of generating of patches introduces crucial hyperparameters. In our case, the baseline model utilizes stride of 4, patch size of 7. These parameters influence the division of images into smaller segments for processing in .

Class	IoU	Acc
background	90.64	93.93
aeroplane	81.25	92.82
bicycle	59.07	81.06
bird	77.53	93.67
boat	65.29	77.04
bottle	59.43	85.44
bus	77.94	92.54
car	75.62	85.66
cat	81.77	94.82
chair	23.07	38.42
cow	69.7	80.54
diningtable	42.83	62.19
dog	71.89	86.42
horse	63.75	83.95
motorbike	71.43	88.2
person	74.4	87.21
pottedplant	45.06	57.26
sheep	70.02	86.84
sofa	40.26	57.03
train	70.71	86.16
tvmonitor	50.8	69.75

Summary:

Scope	mIoU	mAcc	aAcc
global	64.88	80.05	90.83

The performance results reveal the mIoU of the baseline model is 64.88%, mean Accuracy of 80.05 and overall Accuracy of 90.83. Observing the IoU of each class, we can conclude that the model is unable to segment small, thin objects effectively such as chair and table with IoU of 23.07 and 42.83.

RESEARCH WORK

The baseline model lacked few key features which reduced the performance of the model, they include simple MLP, no positional encoding (skipped as patch overlapping mechanism is equipped) and simpler Efficient Self-Attention module is used, hindering the feature extraction. While there are several things affecting the performance, the decoder has a major role. To address this, several techniques were experimented, the best technique involved using Feature Pyramid Network for the decoder and enhancing it.

Feature Pyramid Network is used to aggregate features from different resolutions levels like C1, C2, C3, and C4 to create a single high resolution, multi-scale feature map. FPN works on 4 main techniques.

1. Lateral Connections : The features from each stage of the encoder os processed through a lateral 1 X 1 convolution to align the channel dimensions.
2. Top-Down Pathway : The features from the deepest layer (low-resolution; C4) is upsampled and added to higher-resolution features.
3. FPN Convolutions : Finally 3 X 3 convolutions refine the fused features
4. Output Fusion : The refined outputs from all levels are concatenated, i.e summated.

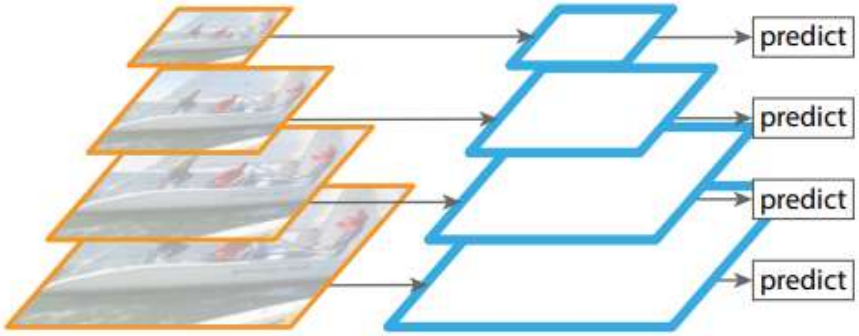

Figure 3 : Feature Pyramid Network [4]

A vanilla FPN acting as a decoder performed better than the baseline model, as it considers the features from each stage and concatenates periodically and systematically. This showed good decoding technique compared to the MLP. However, there was potential for upgrading the FPN network using different techniques such channel attention and Spatial Attention.

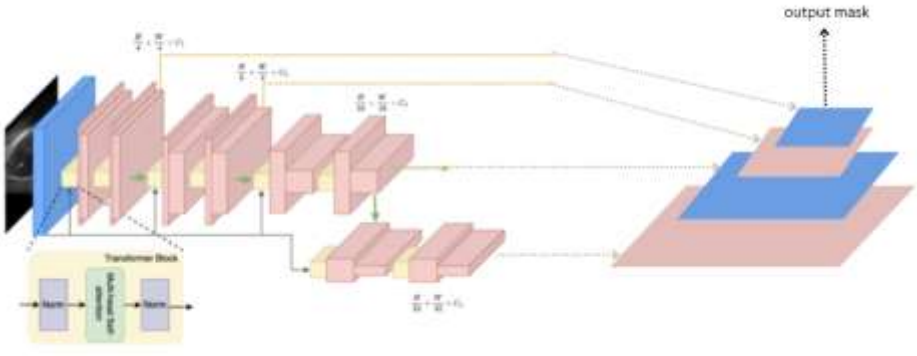

Figure 4 : SegFormer with FPN

Channel Attention is applied using the SE-block (Squeeze and Extraction Block), which is a powerful mechanism for channel-wise attention. The mechanism works by first applying global average pooling to the input feature map to get a squeezed feature, which has a global context. This feature is passed through couple of fully connected layers with ReLU activation and sigmoid function to obtain a set of channel-wise scaling factors. These scaling factors are then used to re-weight the input feature maps, which emphazes important channels and suppressing the lesser useful channels.

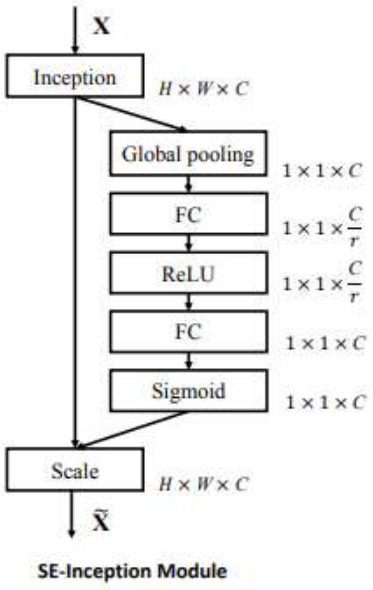

Figure 5 : SE – Block [5]

Convolutional Block Attention module (CBAM) is another technique used in our project which mainly focuses on the spatial attention mechanism while also working with channel attention. CBAM focuses on the important spatial locations within each later across all the feature maps to capture spatial dependencies, following a sigmoid activation to generate spatial attention map. CBAM enhances the model's ability to capture both global and local contextual information.

SE blocks are incorporated for each of the levels in the feature pyramid, which helps in prioritizing the important feature maps and ignoring other feature maps.This recalibration helps the model focus on informative features, segmenting the important parts of the images at each scale. The FPN becomes more effective at discriminating between relevant and irrelevant features leading to improved segmentation results. Similarly, CBAM is applied after the SE block , which generates a spatial attention map highlighting relevant spatial regions in the feature maps, enabling the segmentation of finer objects, contributing to effective segmentation.

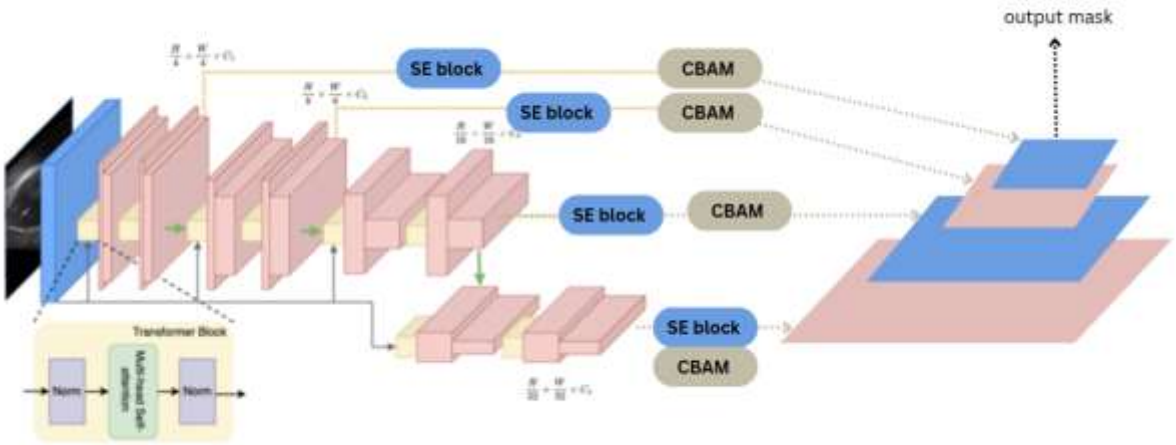

Figure 6 : SegFormer - FPN

RESULT

The baseline model performed well for larger objects, where as they were not accurate in segmenting smaller objects. There were several experiments conducted to understand which decoder to be used and which techniques to be applied over the decoder for better performance.

In the first experiment, we employed a traditional Feature Pyramid network for 4 stages, concatenating the features from corresponding level from the encoder. This showed slightly better performance than the baseline model.

The second experiment involved channel attention and spatial attention for enhancing the performance – SE block and CBAM. This showed a significant increase in the performance compared to the baseline model.

Metrics	**Segformer** **%**	**Segformer with FPN** **%**	**Segformer with FPN (channel and spatial attention)** **%**
mIoU	64.88	65.63	67.19
mAcc	80.05	75.46	76.74
aAcc	90.83	91.5	92.08

Table 1 : Performance metrics

Class	IoU	Acc
background	90.64	93.93
aeroplane	81.25	92.82
bicycle	59.07	81.06
bird	77.53	93.67
boat	65.29	77.04
bottle	59.43	85.44
bus	77.94	92.54
car	75.62	85.66
cat	81.77	94.82
chair	23.07	38.42
cow	69.7	80.54
diningtable	42.83	62.19
dog	71.89	86.42
horse	63.75	83.95
motorbike	71.43	88.2
person	74.4	87.21
pottedplant	45.06	57.26
sheep	70.02	86.84
sofa	40.26	57.03
train	70.71	86.16
tvmonitor	50.8	69.75

Summary:

Scope	mIoU	mAcc	aAcc
global	64.88	80.05	90.83

Table 2 : Metrics of SegFormer

Class	IoU	Acc
background	91.38	96.25
aeroplane	83.48	89.01
bicycle	59.11	74.28
bird	79.93	89.08
boat	65.1	72.73
bottle	64.96	74.91
bus	78.69	85.8
car	76.11	80.48
cat	82.12	90.99
chair	20.2	30.57
cow	67.34	73.09
diningtable	45.92	55.38
dog	68.05	85.85
horse	61.78	74.55
motorbike	75.44	84.39
person	75.37	87.4
pottedplant	40.83	45.19
sheep	76.17	86.03
sofa	39.78	55.79
train	70.39	79.29
tvmonitor	55.97	73.62

2024-11-27 21:06:34,827 - mmseg -
2024-11-27 21:06:34,827 - mmseg -

Scope	mIoU	mAcc	aAcc
global	65.63	75.46	91.5

Table 3: Metrics of SegFormer with FPN

Class	IoU	Acc
background	91.99	96.75
aeroplane	85.06	90.5
bicycle	60.77	72.55
bird	79.62	89.5
boat	66.6	76.63
bottle	68.59	81.76
bus	78.66	84.8
car	76.78	82.59
cat	81.45	92.01
chair	22.66	32.71
cow	72.66	84.07
diningtable	47.27	56.05
dog	71.31	82.63
horse	65.37	77.0
motorbike	76.57	84.78
person	76.52	87.72
pottedplant	44.3	52.14
sheep	74.72	84.58
sofa	40.57	51.92
train	71.4	78.15
tvmonitor	58.16	72.67

Summary:

Scope	mIoU	mAcc	aAcc
global	67.19	76.74	92.08

Table 4: SegFormer-FPN

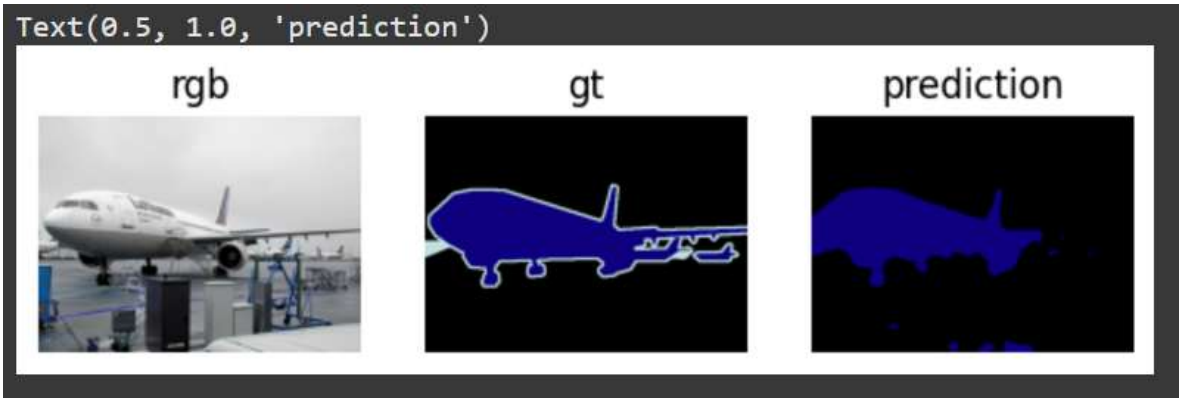

Figure 7 : Inference 1

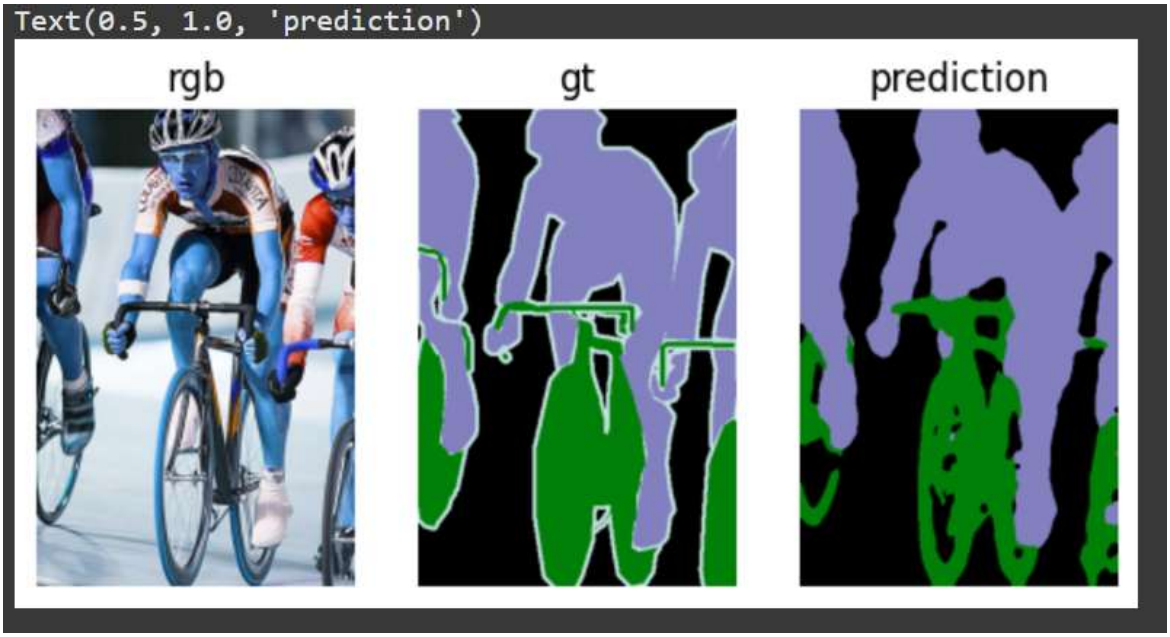

Figure 8 : Inference 2

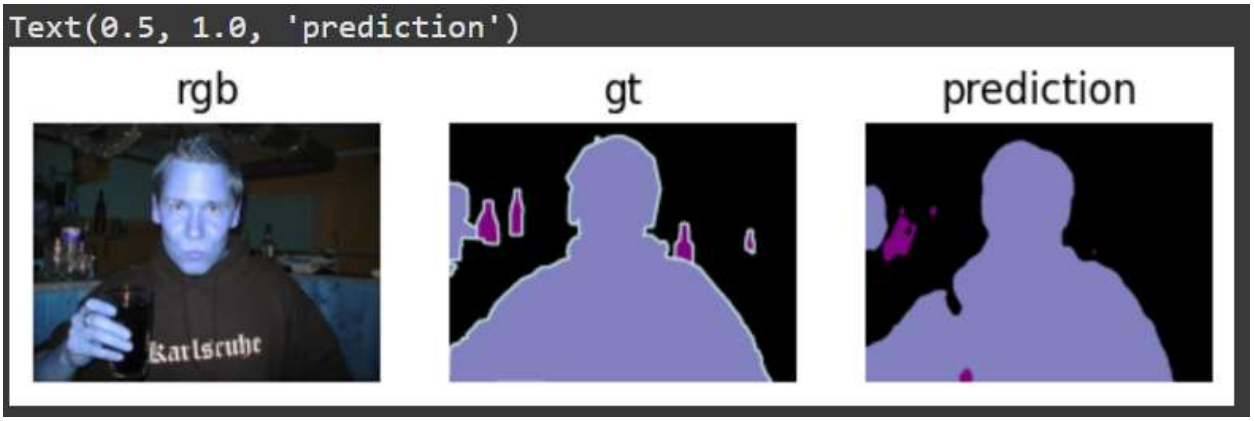

Figure 9 : Inference 3

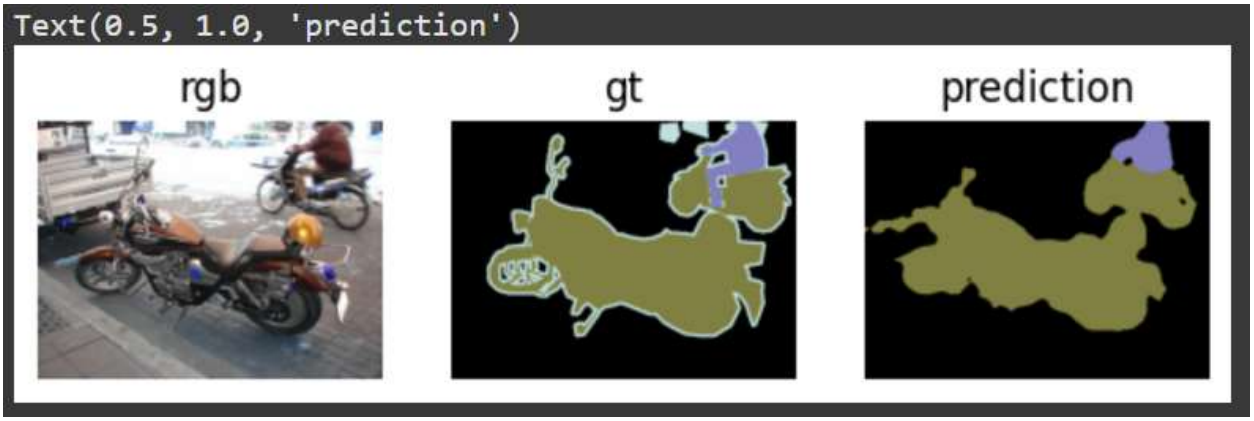

Figure 10 : Inference 4

CONCLUSION

This project presented a semantic segmentation model built with Segformer as the backbone and Feature pyramid Network as the decoder. The Segformer, with the transformer-based architecture, effectively captured long-range dependencies and global information, while the FPN improved the feature aggregation across different scales for accurate segmentation results.

Additionally, the integration of SE block and CBAM enchanced the model's performance. The SE block enhanced the channel attention to find the relevant channels or features and discard others, while CBAM focused in relevant spatial regions improving the segmentation accuracy.

Overall, the combination of SegFormer, FPN, SE block and CBAM resulted in a robust and efficient semantic segmentation model, showcasing the effectiveness of transformer-based architecture in computer vision tasks.

REFERENCES

[1] E Xie, et al., “SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers”

[2] R. Strudel et al., “Segmenter: Transformer for Semantic Segmentation,” in CVPR, 2021

[3] Ronneberger et al.,”U-Net: Convolutional Networks for Biomedical Image Segmentation”

[4]T Y Lin, et al., ” Feature Pyramid Networks for Object Detection”, 2017

[5]Jie Hu, et al., “Squeeze-and-Excitation Networks”, 2019

[6] S Woo, et al.,” CBAM: Convolutional Block Attention Module” 2018

Contents