



[논문 리뷰] SETR [2021]

☀ 본 논문은 Transformer기반으로 Segmentation을 시행하는 SETR Model을 소개합니다.

최근 Semantic Segmentation 모델들은

FCN기반의 Encoder Decoder Architecture로 구성되어 있습니다.

Segmentation내에서 Context Modeling이 중요하게 됨으로서, 최근에는 **dilated/atrous convolutions or inserting attention module**을 사용함으로써, **receptive field**를 증가시키는데, 많은 노력을 기울였습니다.

본 논문에서는 Image를 sequence of patch로 인코딩하여, Semantic Segmentation을 Sequence-to-Sequence의 관점으로 해석하는 대안을 제안합니다.

Transformer의 모든 layer를 global context model내에 도입하였습니다.

이러한 encoder는 decoder는 simple decoder와 결합하여, 강력한 Segmentation model을 생성할 수 있습니다. 이를 **SEgmentation TRansformer (SETR)**라고 부릅니다.

Introduction

기존의 Segmentation Model들은 FCN(Fully Convolutional Network)에 기반을 두었으며, 일반적으로 Encoder & Decoder architecture를 갖추고 있습니다.

일반적으로 Encoder는 feature representation을 training합니다. 반면에, Decoder는 Encoder가 추출한 feature representation에 Pixel-level내에서 Classification을 시행합니다.

Computational cost에 관한 염려 때문에, feature map의 resolution은 점진적으로 감소합니다.

그리고 Encoder는 increased receptive field에서 abstract/semantic visual concept을 더 학습하게 됩니다.

구조적인 관점에서 receptive field는 Network의 Depth에 선형적인 관계를 가집니다.

그러므로, Semantic Segmentation에서 넓은 receptive field를 갖기 위해, depth를 늘려야 하지만, receptive field가 제한이 있어, FCN은 information에 의존하면서, long-range 학습하는 것은 상당히 제한됩니다.

앞서, 언급한 문제들을 해결하기 위해, 여러가지 연구를 시행해왔습니다.

1. large kernel size, atrous convolutions, image/feature pyramids 등 Convolution operation을 조작하는 것입니다.
2. FCN architecture내에 attention module을 통합시키는 방법이다.



본 논문에서는 점진적으로 해상도를 감소시키는 encoder기반의 layer를 쌓은 SETR이라고 불리는 pure transformer를 제안합니다.

Transformer의 Encoder는 Patch Embedding을 학습하기 위해,

Input Image를 sequence of Image Patch로 여긴다. Feature representation을 식별하는 것을 학습하기 위해 global self-attention Modeling Sequence로 변환합니다.

Model Process

구체적으로는, Sequence of Patch를 생성하기 위해, Image를 고정된 patch로 분해합니다.

이후, Image Patch들을 flatten한 후, linear embedding layer에 적용합니다. 이후, 우리는 Transformer에 Input으로 넣어, Sequence of feature Embedding Vector를 얻을 수 있습니다.

Transformer의 Encoder에서 feature들을 학습한 후, Decoder는 original Image Resolution으로 recover하기 위해 사용됩니다. 이러한 과정에서는 Spatial Resolution을 Downsampling을 하지 않으며,

그렇게 됨으로써 Semantic Segmentation에 새로운 관점을 제시합니다.

해당 논문은 3가지의 Contribute로 인하여 작성되었습니다.

1. 이전의 FCN의 관점에서 NLP의 **Sequence-to-Sequence**의 관점으로 reformulation 하였습니다.
2. Transformer를 사용함으로써, fully attentive feature를 구현하였습니다.
3. 3가지의 다양한 decoder를 소개합니다.

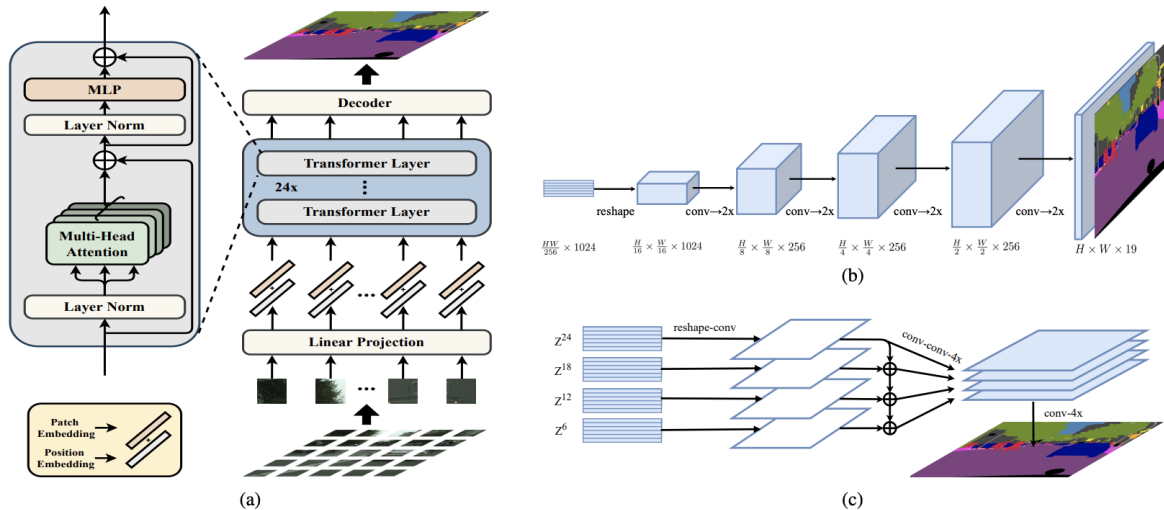


Figure 1. Schematic illustration of the proposed *Segmentation Transformer* (SETR) (a). We first split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. To perform pixel-wise segmentation, we introduce different decoder designs: (b) progressive upsampling (resulting in a variant called SETR-PUP); and (c) multi-level feature aggregation (a variant called SETR-MLA).

Method

FCM-based Semantic Segmentation

최근 연구에서는, FCN과 attention mechanism을 같이 활용하는 것이 long-range contextual information을 학습하는데 더 효율적인 전략이라고 하였다.

이러한 방법은 smaller input size를 higher layer로 learning하는데, 제한을 합니다.

그리하여, Self-Attention 기반의 Encoder를 지닌, Segmentation Transformers (SETR)을 제안합니다.

Sequentialization and Position Embedding

- Transformer 모델을 사용하기 위해, $H \times W \times 3$ resolution의 이미지를 C hidden channel size를 가지는 L개의 Sequential Vector형태로 re-representation합니다.
- 논문에서 ViT와 동일하게 입력 영상을 16×16 개의 patch로 분할합니다. 각각의 patch를 flatten 한 후 linear projection하여, C차원으로 축소하여, 결과적으로 1차원 patch Embedding의 Sequence로 변환합니다.
- 추가로, 순서 정보를 주기 위해 각 Patch에 해당하는 Position Embedding을 따로 추가합니다.

Transformer Encoder

- 기존, Transformer와 다른 점은 Multi-Head Self Attention을 적용하기 전, layer norm을 적용합니다. [MLP(Multi-Layer Perceptron) layer를 따로 추가합니다.]
- 24개의 Transformer Block Layer로 모델을 구성합니다.
- 모델의 구조는 이미지의 Patch형태로, Re-representation되어, Transformer Input으로 활용되어, 모든 레이어는 “Global Reception Field”를 가지게 됩니다.

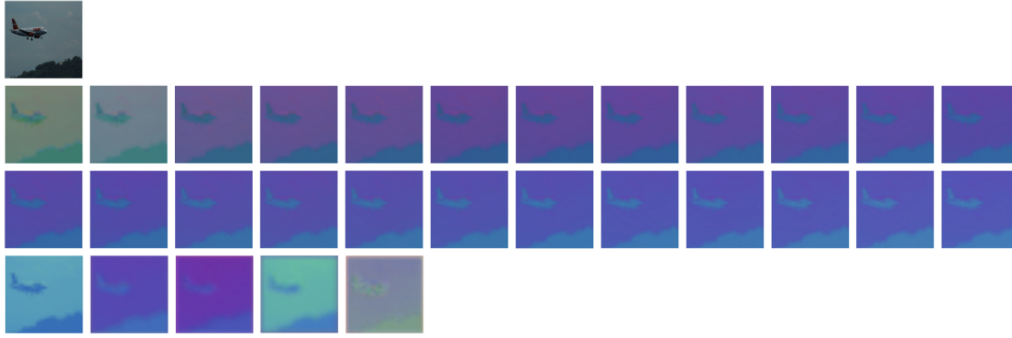
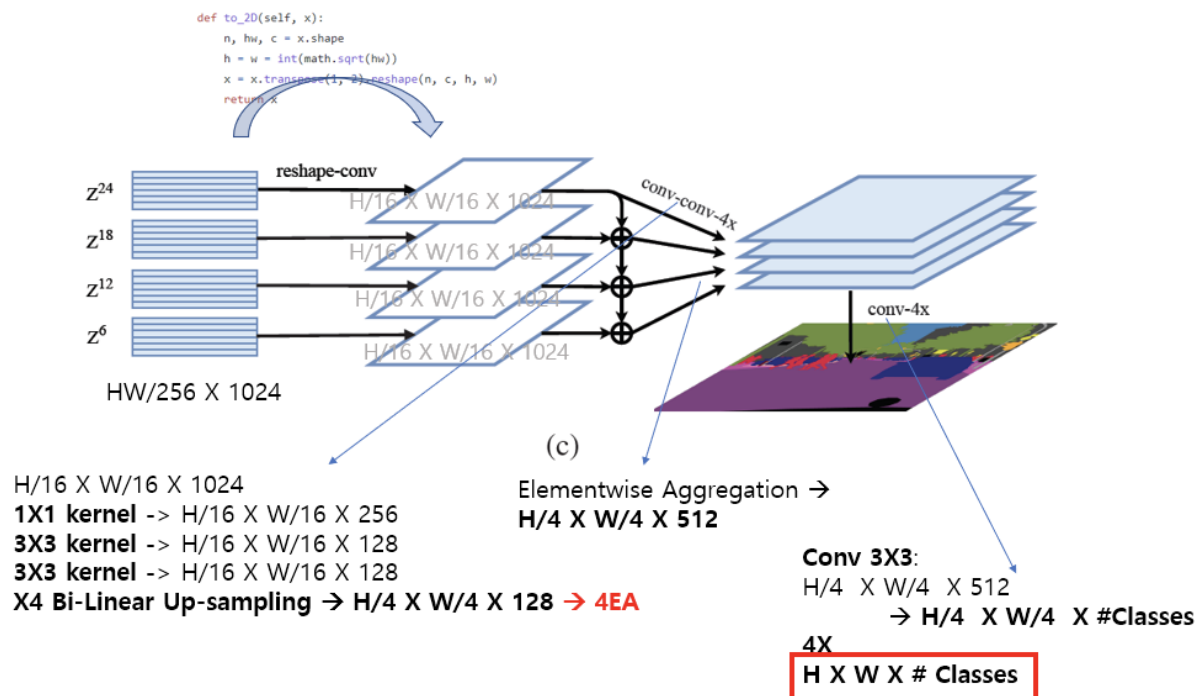
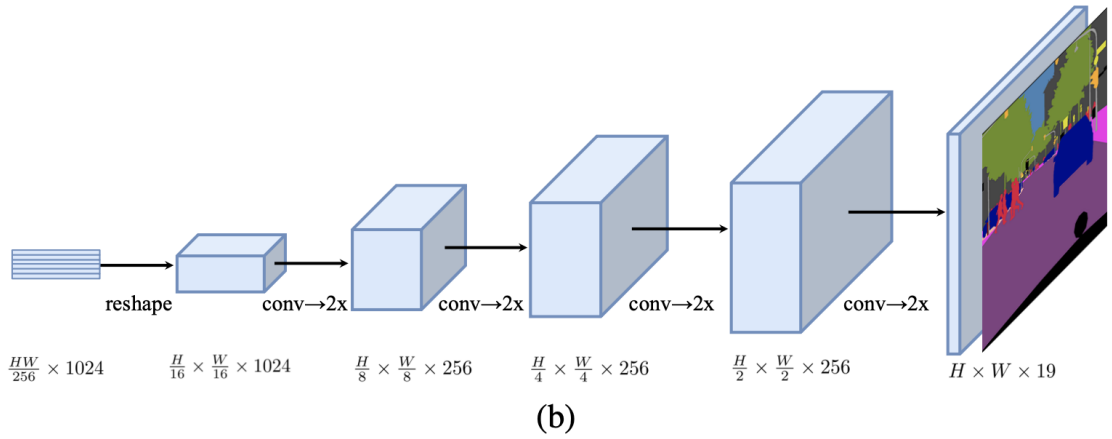


Figure 9. Visualization of output feature of layer $Z^1 - Z^{24}$ and $U^1 - U^5$ of SETR-PUP trained on Pascal Context. Best view in color. **First row:** The input image. **Second row:** Layer $Z^1 - Z^{12}$. **Third row:** Layer $Z^{13} - Z^{24}$. **Fourth row:** Layer $U^1 - U^5$.

Decoder Design



- Pixel Level Segmentation을 위해,
Transformer Encoder의 Output $H/16 \times W/16 \times C$ Feature Map을 $[H \times W \times \text{num. of class}]$ 형태의 Segmentation Map 형태로 Re-representation하는 역할을 합니다.
- 우선 transformer Output Z 을 $\frac{H}{16} \times \frac{W}{16} \times C$ 로 reshape 후, decoder를 적용하여, 최종 Segment map을 도출합니다.
 - Navie UpSampling (Navie)
 - 가장 기본적인 방법으로, bilinear interpolation을 통하여, $[H \times W \times \text{num. of class}]$ segmentation Map을 만들어줍니다.
 - Progressive Up-Sampling(PUP)
 - Noisy를 발생하는 One-step upscaling 방법 대신에, 우리는 progressive upsampling strategy를 사용합니다.
adversarial effect를 최대한 완화하기 위해서, 2배씩 4번 Upscaling하는 방법을 제안합니다.



◦ Multi-Level feature Aggregation (MLA)

- FPN(Feature Pyramid Network)와 유사한 Multi-Level Feature를 사용하는데,
24개의 Transformer Block Layer에서 추출하여, Multi-level Feature개념으로 사용합니다.

Model	T-layers	Hidden size	Att head
T-Base	12	768	12
T-Large	24	1024	16

Table 1. Configuration of Transformer backbone variants.

Method	Pre	Backbone	#Params	40k	80k
FCN [39]	1K	R-101	68.59M	73.93	75.52
Semantic FPN [39]	1K	R-101	47.51M	-	75.80
<i>Hybrid-Base</i>	R	T-Base	112.59M	74.48	77.36
<i>Hybrid-Base</i>	21K	T-Base	112.59M	76.76	76.57
<i>Hybrid-DeiT</i>	21K	T-Base	112.59M	77.42	78.28
<i>SETR-Naïve</i>	21K	T-Large	305.67M	77.37	77.90
<i>SETR-MLA</i>	21K	T-Large	310.57M	76.65	77.24
<i>SETR-PUP</i>	21K	T-Large	318.31M	78.39	79.34
<i>SETR-PUP</i>	R	T-Large	318.31M	42.27	-
<i>SETR-Naïve-Base</i>	21K	T-Base	87.69M	75.54	76.25
<i>SETR-MLA-Base</i>	21K	T-Base	92.59M	75.60	76.87
<i>SETR-PUP-Base</i>	21K	T-Base	97.64M	76.71	78.02
<i>SETR-Naïve-DeiT</i>	1K	T-Base	87.69M	77.85	78.66
<i>SETR-MLA-DeiT</i>	1K	T-Base	92.59M	78.04	78.98
<i>SETR-PUP-DeiT</i>	1K	T-Base	97.64M	78.79	79.45

Experiments

- mmsegmentation 기반으로 구현
- Augmentation:
 - Random cropping (768, 512 and 480 for Cityscapes, ADE20K and Pascal Context respectively)
 - Random horizontal flipping
- Optimization
 - Init. learning rate: 0.001 (on ADE20K, Pascal Context), 0.01 (on Cityscapes)
 - SGD with polynomial learning rate decay schedule
 - Momentum = 0.9, weight decay = 0

Figure 2. **Qualitative results on ADE20K:** SETR (right column) vs. dilated FCN baseline (left column) in each pair. Best viewed in color and zoom in.

Method	Pre	Backbone	#Params	mIoU
FCN (160k, SS) [39]	1K	ResNet-101	68.59M	39.91
FCN (160k, MS) [39]	1K	ResNet-101	68.59M	41.40
CCNet [25]	1K	ResNet-101	-	45.22
Strip pooling [23]	1K	ResNet-101	-	45.60
DANet [18]	1K	ResNet-101	69.0M	45.30
OCRNet [54]	1K	ResNet-101	71.0M	45.70
UperNet [49]	1K	ResNet-101	86.0M	44.90
Deeplab V3+ [11]	1K	ResNet-101	63.0M	46.40
SETR-Naïve (160k, SS)	21K	T-Large	305.67M	48.06
SETR-Naïve (160k, MS)	21K	T-Large	305.67M	48.80
SETR-PUP (160k, SS)	21K	T-Large	318.31M	48.58
SETR-PUP (160k, MS)	21K	T-Large	318.31M	50.09
SETR-MLA (160k, SS)	21K	T-Large	310.57M	48.64
SETR-MLA (160k, MS)	21K	T-Large	310.57M	50.28
SETR-PUP-DeiT (160k, SS)	1K	T-Base	97.64M	46.34
SETR-PUP-DeiT (160k, MS)	1K	T-Base	97.64M	47.30
SETR-MLA-DeiT (160k, SS)	1K	T-Base	92.59M	46.15
SETR-MLA-DeiT (160k, MS)	1K	T-Base	92.59M	47.71

Table 4. **State-of-the-art comparison on the ADE20K dataset.** Performances of different model variants are reported. SS: Single-scale inference. MS: Multi-scale inference.

Figure 3. **Qualitative results on Pascal Context:** SETR (right column) vs. dilated FCN baseline (left column) in each pair. Best viewed in color and zoom in.

Method	Pre	Backbone	mIoU
FCN (80k, SS) [39]	1K	ResNet-101	44.47
FCN (80k, MS) [39]	1K	ResNet-101	45.74
DANet [18]	1K	ResNet-101	52.60
EMANet [31]	1K	ResNet-101	53.10
SVCNet [16]	1K	ResNet-101	53.20
Strip pooling [23]	1K	ResNet-101	54.50
GFFNet [30]	1K	ResNet-101	54.20
APCNet [19]	1K	ResNet-101	54.70
SETR-Naïve (80k, SS)	21K	T-Large	52.89
SETR-Naïve (80k, MS)	21K	T-Large	53.61
SETR-PUP (80k, SS)	21K	T-Large	54.40
SETR-PUP (80k, MS)	21K	T-Large	55.27
SETR-MLA (80k, SS)	21K	T-Large	54.87
SETR-MLA (80k, MS)	21K	T-Large	55.83
SETR-PUP-DeiT (80k, SS)	1K	T-Base	52.71
SETR-PUP-DeiT (80k, MS)	1K	T-Base	53.71
SETR-MLA-DeiT (80k, SS)	1K	T-Base	52.91
SETR-MLA-DeiT (80k, MS)	1K	T-Base	53.74

Table 5. **State-of-the-art comparison on the Pascal Context dataset.** Performances of different model variants are reported. SS: Single-scale inference. MS: Multi-scale inference.

Conclusion

FCN의 기반의 방식과 다르게, FCN에 의존하는 방식을 제거하고, Sequence-to-Sequence기반인 Transformer를 제안합니다. Transformer는 Global Context를 Model이 learning합니다.

Reference

Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers


Most recent semantic segmentation methods adopt a fully-convolutional network (FCN) with an encoder-decoder architecture. The encoder progressively reduces the spatial resolution and learns more abstract/semantic visual concepts with larger receptive fields. Since context modeling is critical for

 <https://arxiv.org/abs/2012.15840>



All about Segmentation

Transformer 모델은 이미 NLP분야에서 성능을 입증하였고, ViT(Vision Transformer)를 통해 Image Classification에서 좋은 성능을 보였다. 이는 이미지 특징 추출을 위해 stacked convolution 구조를 통한 공간 정보는 압축(손해)하며 global context를 학습한다는 종래 방식이 필수가 아님을 증명하였다.

 https://pseudo-lab.github.io/SegCrew-Book/docs/ch1/01_03_01_SETR.html

SETR-Naïve	21K	T-Large	305.67M	77.37	77.90
SETR-MLA	21K	T-Large	310.57M	76.65	77.24
SETR-PUP	21K	T-Large	318.31M	78.39	79.34
SETR-PUP	R	T-Large	318.31M	42.27	-
SETR-Naïve-Base	21K	T-Base	87.69M	75.54	76.25
SETR-MLA-Base	21K	T-Base	92.59M	75.60	76.87
SETR-PUP-Base	21K	T-Base	97.64M	76.71	78.02
SETR-Naïve-DeiT	1K	T-Base	87.69M	77.85	78.66
SETR-MLA-DeiT	1K	T-Base	92.59M	78.04	78.98
SETR-PUP-DeiT	1K	T-Base	97.64M	78.79	79.45

