



[논문 리뷰] Swin Transformer [2021]

Abstract

NLP에서 ComputerVision에 Transformer를 적용하는데 있어 차이점으로는 이미지는 높은 해상도를 가지고 있고, Scale of Visual이 상당히 크다는 것입니다.

이러한 차이점을 다루고자,

본 논문에서는 **Shifted Window 기법을 활용한 Hierarchical Transformer**를 제안합니다.

Shifted Window 방식은 self-attention computation을 cross-window connection을 허용하는 non-overlapping local Windows를 사용함으로써, 효율성을 높입니다.

Hierarchical Architecture는 다양한 스케일에서 유연함을 가지고 있습니다. 그리고, 이미지의 크기별로 선형계산복잡도를 가지고 있습니다.

Introduction

Transformer의 적용가능성을 확장하여,
ComputerVision의 Backbone으로 범용적으로 활용될 수 있도록 하였다.

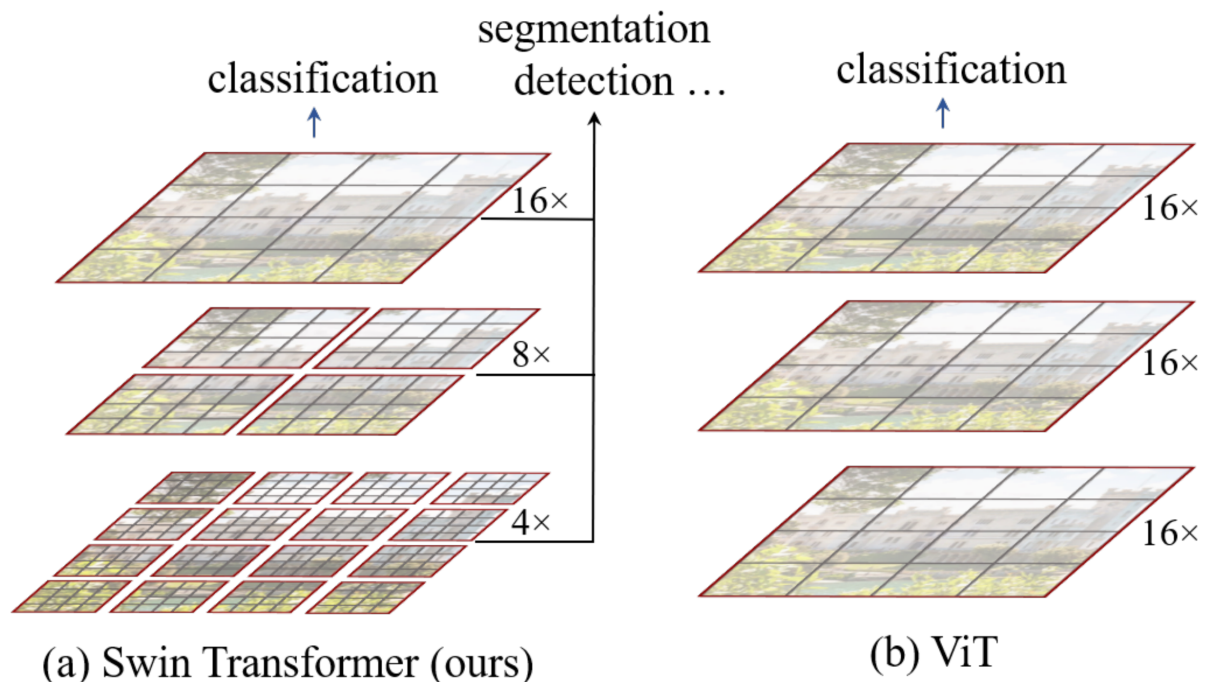
NLP와 다르게 Vision의 element들은 거대하고, 상당한 Scale을 가지고 있습니다.

이전의 Vision Transformer는 고정된 token을 가지고 있기에, Vision application에 적합하지 않은 문제가 있습니다.

Pixel 내에서 예측이 필요한 Semantic Segmentation과 같은, Vision Task는 고해상도 이미지에서 Transformer를 다루기에는 상당한 어려움이 있습니다.

이러한 문제를 극복하기 위해, Transformer를 backbone으로 활용하는 Swin Transformer를 제안하였습니다.

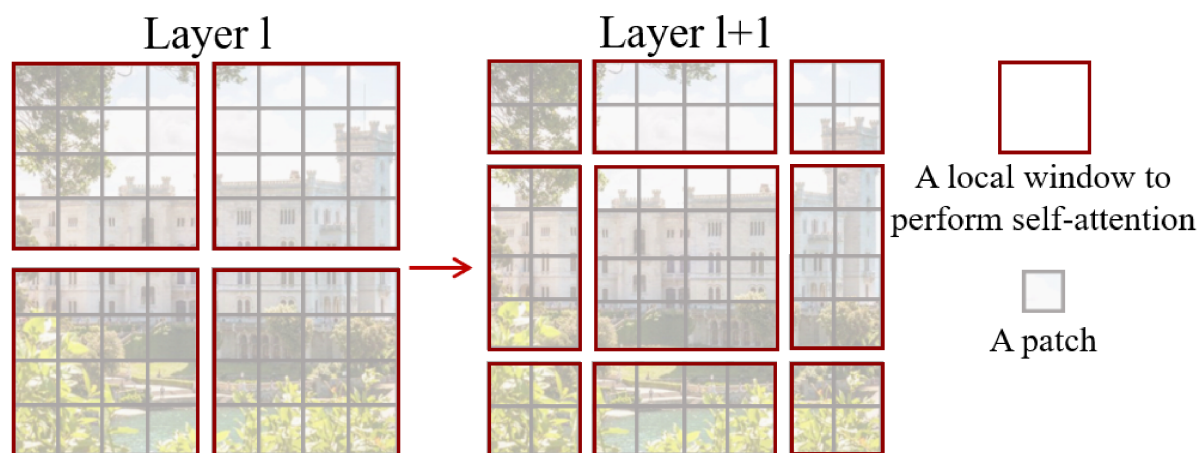
Swin Transformer는 hierarchical feature map으로 구성되어 있으며, image size에 대한 선형계산복잡도를 가지고 있습니다.



Swin Transformer는 작은 Patch부터 시작하여, hierarchical representation으로부터 구성되어 있으며, Transformer의 layer의 깊이가 점점 깊어질 수록, 주변 Patch들과 점차적으로 계속해서 병합합니다.

Linear Computational Complexity는 non-overlapping windows내에서 self-attention을 사용함으로써, 달성합니다.

Swin Transformer의 가장 핵심은 self-attention layer 사이에, shift of the window partition을 적용하는 것입니다.



Shifted Windows는 이전의 Layer의 Window와 Bridge 역할을 함으로써, Modeling power를 향상시키는 연결을 제공합니다. Shifted Windows 기법은 sliding Windows 기법보다 low latency를 가집니다. 또한, Shifted Windows 기법은, all-MLP architecture에서 상당한 benefit을 가져다 줍니다.

Related Work

CNN and Variants

CNN이 main stream이었고, VGG, GoogleNet, ResNet, DenseNet, HRNet, EfficientNet등 많은 논문들이 나오면서 계속해서 발전해왔습니다.

Self-Attention based Backbone Architecture

ResNet기반의 Spatial convolution layer를 self-attention layer로 대체하려는 시도가 있었으나, 앞서 언급한 sliding Window 방식의 memory access 문제가 있어 어려움을 겪었습니다.

Self-Attention / Transformer to Complement CNNs

CNN을 보완하기 위해, self-attention을 사용하려는 방법을 시도하였습니다.

Transformer based Vision backbones

Transformer를 Vision task의 backbone으로 사용하기 위해서 연구가 되고 있습니다. 대표적으로는 Vit, Deit가 존재합니다.

Method

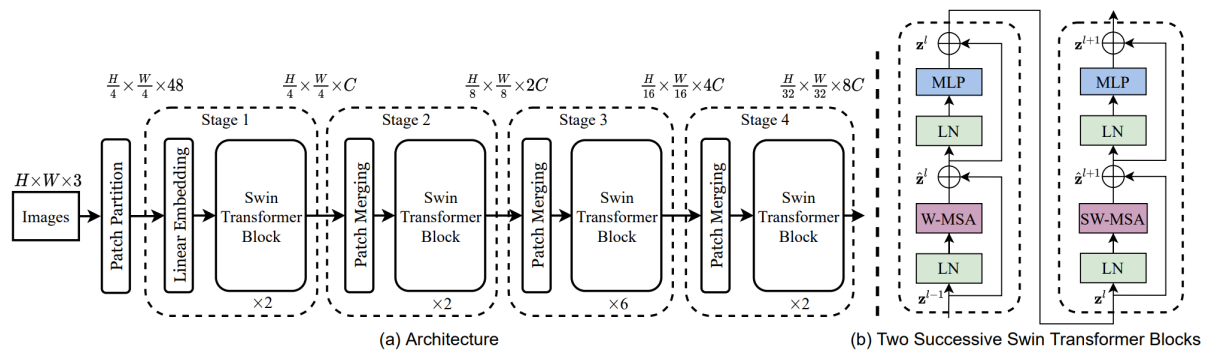


Figure 3. (a) The architecture of a Swin Transformer (Swin-T); (b) two successive Swin Transformer Blocks (notation presented with Eq. (3)). W-MSA and SW-MSA are multi-head self attention modules with regular and shifted windowing configurations, respectively.

Swin Transformer는 4가지 Stage로 구성되어 있습니다.

STAGE 1

ViT와 같은 Patch Splitting Module을 활용하여, non-overlapping patch로 RGB 이미지를 나눕니다.

나뉜 Patch는 Token으로 활용되며, 해당 Feature는 raw Pixel의 RGB값으로 이어 붙입니다.

Patch를 4x4로 나누어, RGB value를 곱하여, 48 Dimension을 가지게 합니다.

이후, Linear Embedding을 활용하여, $(H/4) * (W/4) * C$ 로 텐서를 변환합니다.

(C는 Model의 크기에 따라, 달라집니다.)

STAGE 2 ~ 4

Small Patch들을 합쳐가면서, hierarchical representation을 만들어갑니다. 인접한 2*2개의 patch들을 계속해서, concat하여, 채널 수가 4배가 되게 생성합니다.

이후, linear layer를 활용하여, 최종적으로 2C가 되게 만듭니다.

Swin Transformer Block

처음 Block에서는 Window Multi-head Self Attention(W-MSA)를 통과하고,

이후 Block에서는 Shifted Window based Multi-head Self Attention(SW-MSA)를 통과해야 한다.

각각 2-layer MLP, Layer Norm(LN), GELU를 통과하게 됩니다.

Shifted Window based Self-Attention

Self-attention in non-overlapped windows

효율적인 모델링을 위해, local-windows에 self-attention을 적용합니다. Windows는 이미지가 손상되지 않고, 고르게 만듭니다. M(window size)는 h*w(image size)에 비해 훨씬 작기 때문에 연산량이 적고, image size가 커지더라도, ViT에 비해, 연산량을 줄일 수 있습니다.

하지만 Window가 fix되어 있기 때문에, self-attention시에 고정된 부분만을 수행한다는 단점이 있어, window를 shift하여 한 번 더 self-attention을 수행해서 문제를 해결합니다.

Shifted window partitioning in successive blocks

window를 shift하는 것을 cyclic shift라고 합니다. Window size의 1/2만큼 우측 하단으로 shift한 후 A, B, C구역을 padding합니다. Padding시키는 부분은 좌 상단에서 온 것이므로 A, B, C를 포함해서, Self-attention을 진행하는 것은 의미가 없습니다.

따라서 A, B, C에 mask를 씌운 뒤 self-attention을 수행합니다.

그 뒤에 reverse cyclic shift를 진행하여, 원래 값으로 되돌립니다.

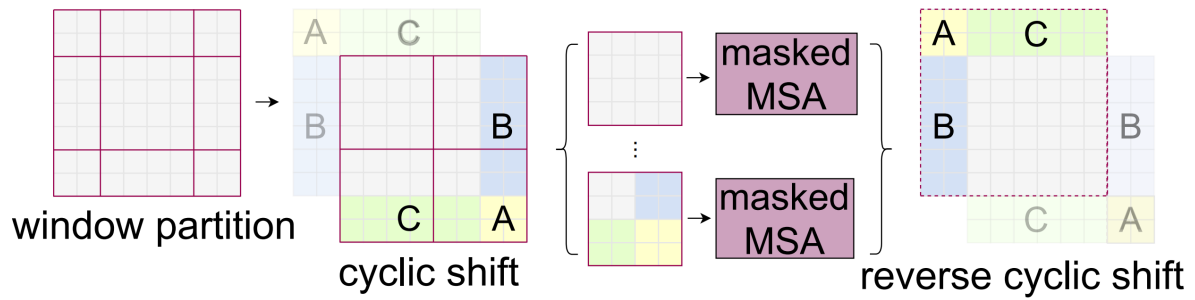


Figure 4. Illustration of an efficient batch computation approach for self-attention in shifted window partitioning.

Relative position bias

Swin Transformer는 ViT와 다르게, Position Embedding을 곧바로 진행하지 않습니다.

Self-attention 과정에서 relative Position bias를 추가함으로써, 그 역할을 대체하였습니다.

Softmax를 활용하기 전, B를 더해주는데 이것이 Relative Position Bias입니다.

Experiment

ViT-B/16 [20]	384^2	86M	55.4G	85.9	77.9
ViT-L/16 [20]	384^2	307M	190.7G	27.3	76.5
DeiT-S [63]	224^2	22M	4.6G	940.4	79.8
DeiT-B [63]	224^2	86M	17.5G	292.3	81.8
DeiT-B [63]	384^2	86M	55.4G	85.9	83.1
Swin-T	224^2	29M	4.5G	755.2	81.3
Swin-S	224^2	50M	8.7G	436.9	83.0
Swin-B	224^2	88M	15.4G	278.1	83.5
Swin-B	384^2	88M	47.0G	84.7	84.5

(b) ImageNet-22K pre-trained models


method	image size	#param.	FLOPs	throughput (image / s)	ImageNet top-1 acc.
R-101x3 [38]	384 ²	388M	204.6G	-	84.4
R-152x4 [38]	480 ²	937M	840.5G	-	85.4
ViT-B/16 [20]	384 ²	86M	55.4G	85.9	84.0
ViT-L/16 [20]	384 ²	307M	190.7G	27.3	85.2
Swin-B	224 ²	88M	15.4G	278.1	85.2
Swin-B	384 ²	88M	47.0G	84.7	86.4
Swin-L	384 ²	197M	103.9G	42.1	87.3

Conclusion

Swin Transformer는 hierarchical feature representation을 가지고 있습니다. 뿐만 아니라, Input Image에 대한 선형 복잡도를 가지고 있습니다. Swin Transformer의 핵심 요소는 self attention 기반의 shifted windows를 가지고 있다는 점입니다. 그리고 이전의 ViT보다 low latency를 가진다는 점이 있습니다.

Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

This paper presents a new vision Transformer, called Swin Transformer, that capably serves as a general-purpose backbone for computer vision. Challenges in adapting Transformer from language to vision arise from differences between the two domains, such

 <https://arxiv.org/abs/2103.14030>



[논문 리뷰] Swin Transformer: Hierarchical Vision Transformer using Shifted Windows (2021)

나의 정리 논문이 지적한 문제점: NLP에서 사용되는 Transformer가 vision task에 적용되는데 큰 문제점이 두 가지 있다. visual entity의 scale이 큰 variation을 가진다. image resolution이 커지면 computation cost가 매우 커진다. 해결 방안: hierarchical feature를 생성하는 transformer를 사용한다. 계층 구조를 사용하기 때문에

 <https://talkto.tistory.com/22>

