Swin Transformer: Hierarchical Vision Transformer using Shifted Windows

Ze Liu, Yutong Lin, Yue Cao, Han Hu

[v2] 17 Aug 2021

Overview

Last time…

We’ve already seen ViT can perform well on classification task

This time…

Transformer can actually perform well on all vision task, including object detection and semantic segmentation

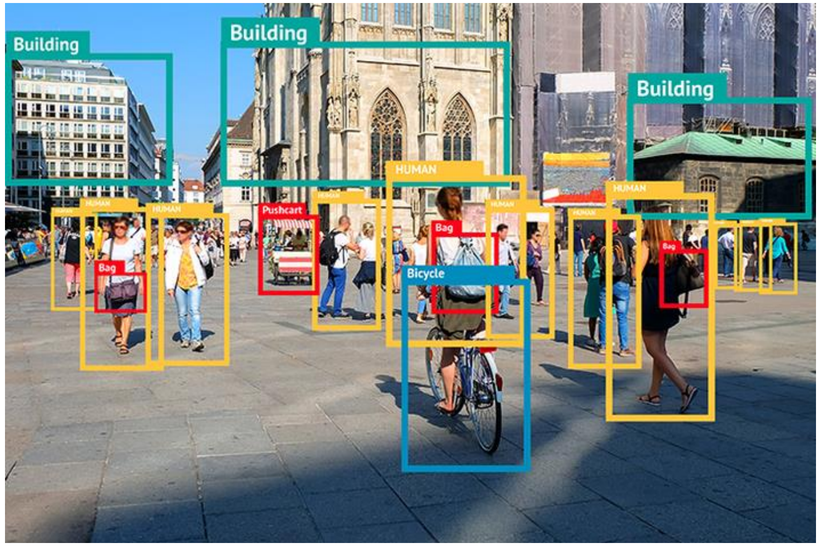
Goal: Replace CNN and become a backbone for computer vision

Introduction & Motivation

Introduction & Motivation

- Visual entity could be in different scales

Ex. People in different size are the same class



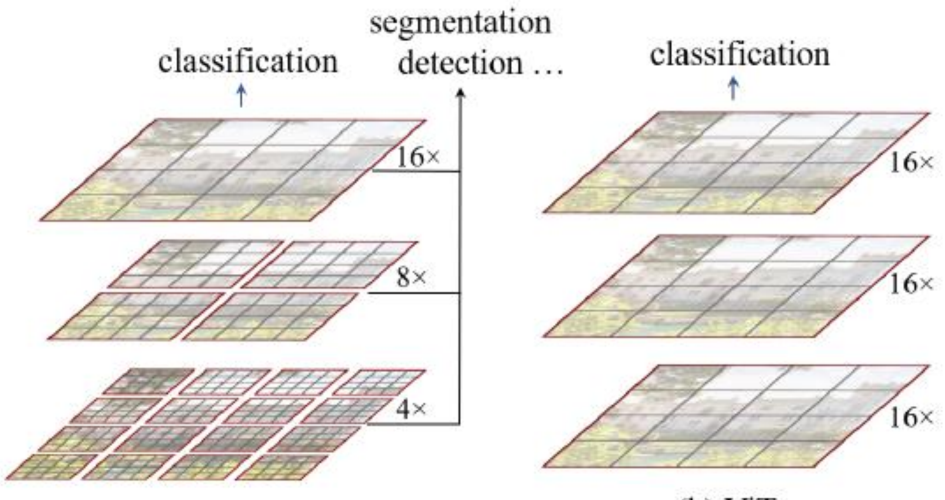
- High computational complexity
- Global self-attention is too expensive for some task

Ex. semantic segmentation

Introduction & Motivation

- Visual entity could be in different scales
- Get feature in different resolution level

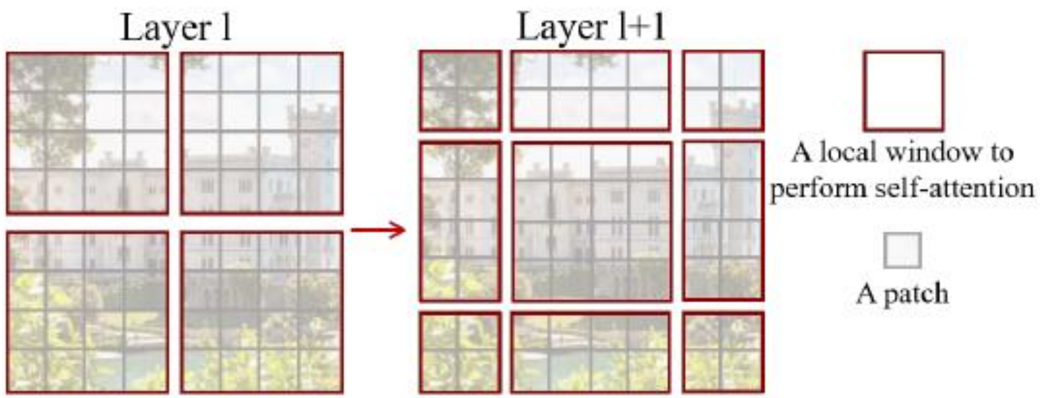
- Hierarchical structure



(a) Swin Transformer (ours)
(b) ViT

- High computational complexity
- Global self-attention is too expensive for some task
- Limit the SA within a window

- Shifted window



Related Work

Related Work

1. Self-attention/Transformers to complement CNNs

- Providing the capability to encode distant dependencies

2. Transformer based vision backbones

- Vision Transformer (ViT) directly applies a Transformer architecture for image classification.

Method

Notation

Image

224
pixels

224*224 image

Patch
(computational unit)

4
pixels

56 *56 patches

Window
(M * M Patches)
(self attention unit)

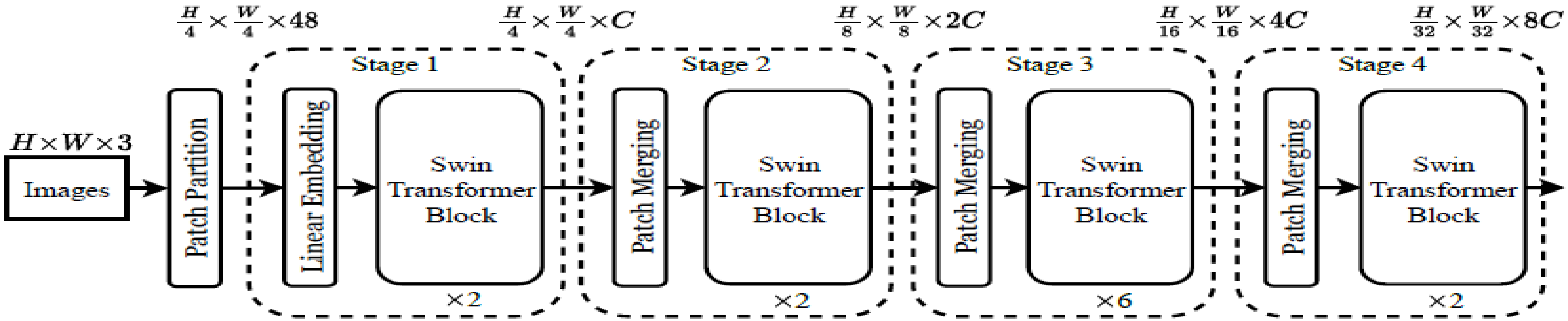
M = 4

14*14 windows

C: Dimension of the embedding vector

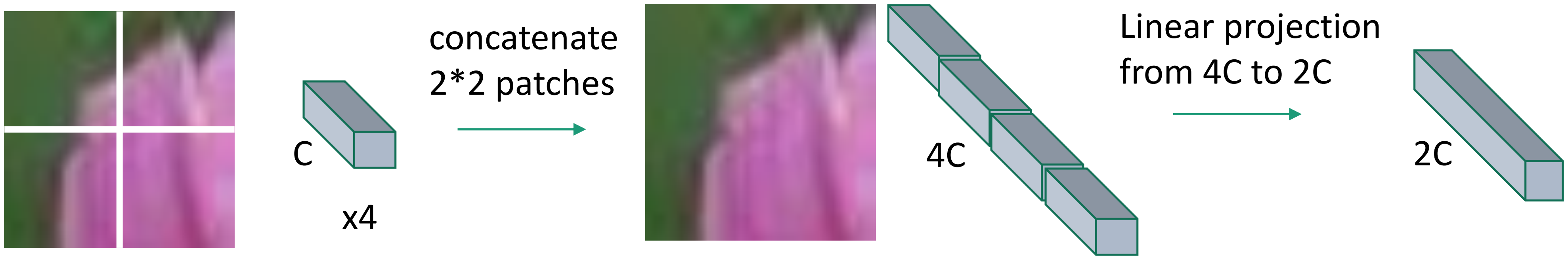
Overall Architecture

Swin-T: 56*56*48 56*56*96 28*28*192 14*14*384 7*7*768



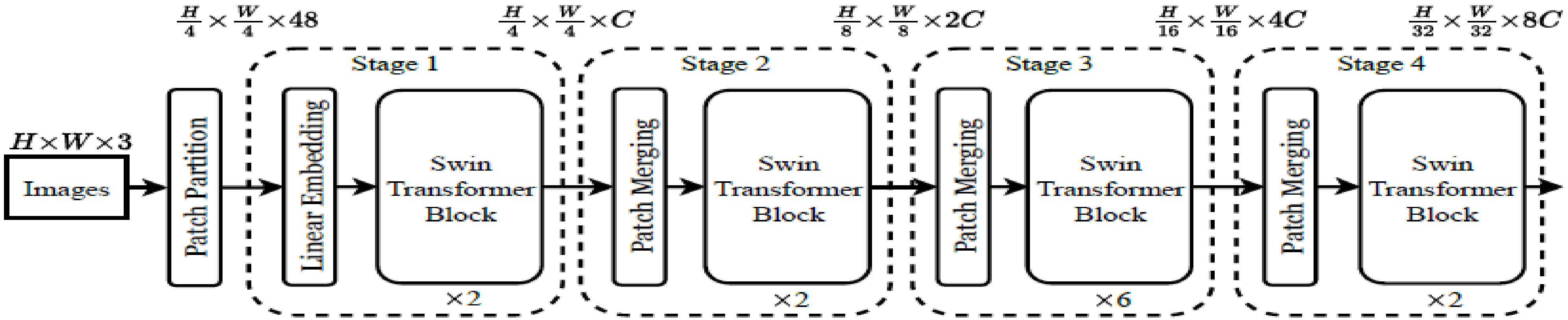
224*224*3 56*56*96 28*28*192 14*14*384 7*7*768

Patch merging:

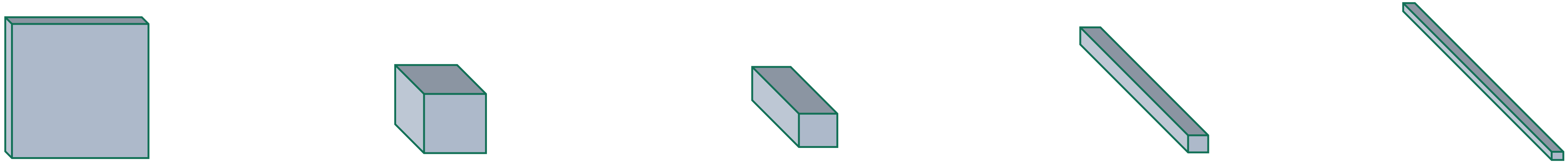


Overall Architecture

Swin-T: 56*56*48 56*56*96 28*28*192 14*14*384 7*7*768



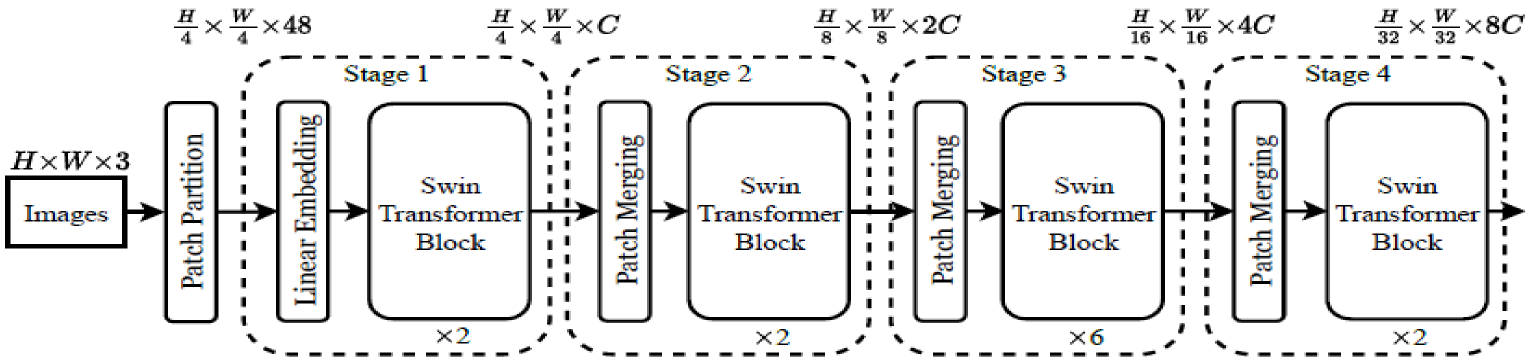
224*224*3 56*56*96 28*28*192 14*14*384 7*7*768

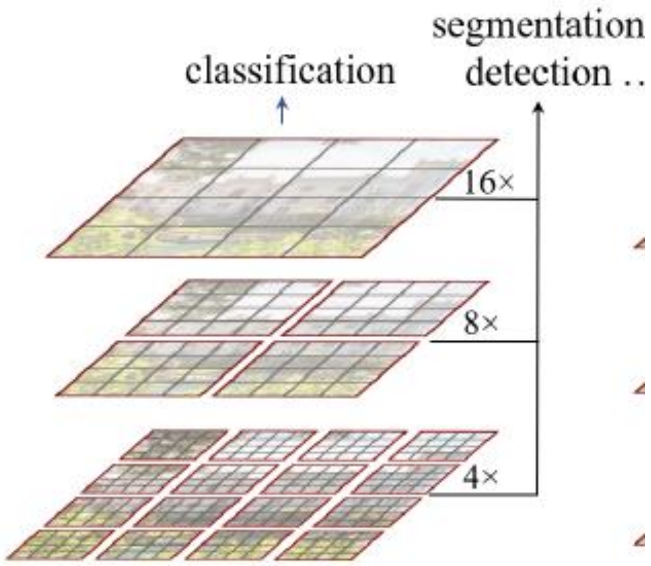


Look like CNN!

Overall Architecture

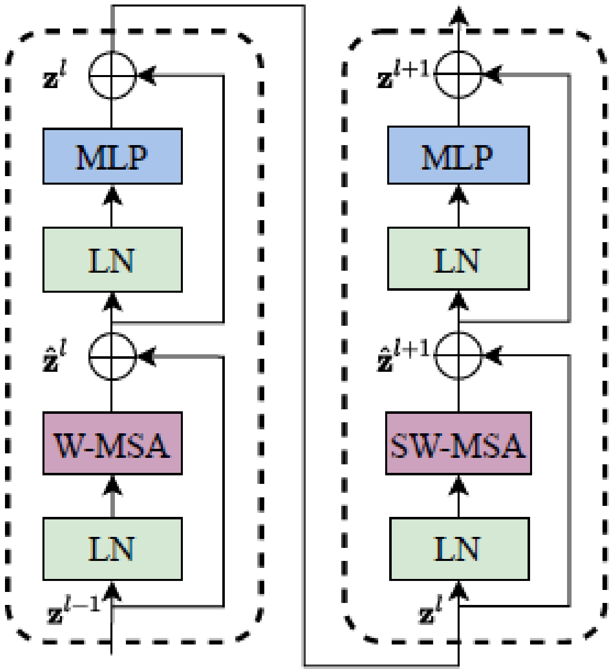
Swin-T:

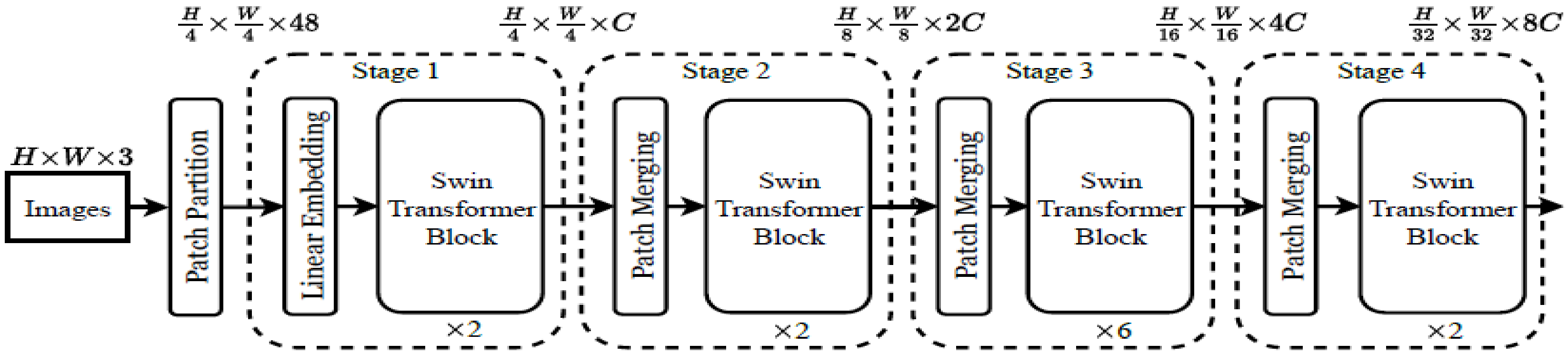




- Patch merging layers capture the hierarchical features
- Exactly the same feature map resolutions as VGG and ResNet
- As a result, the proposed architecture can conveniently replace the backbone networks

Swin Transformer block

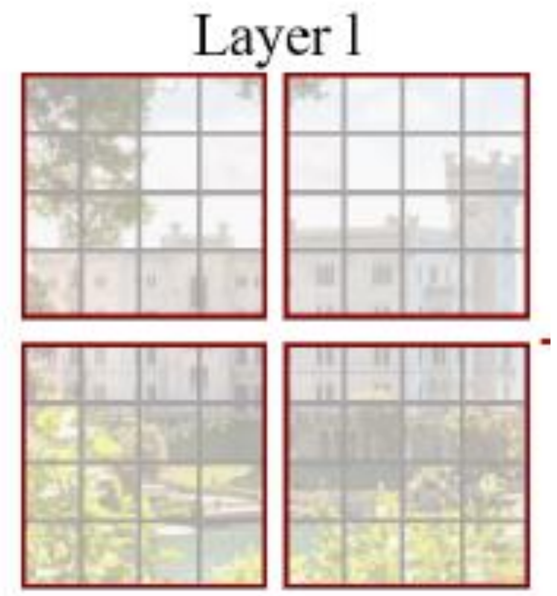




- Overall structure remains similar to the original transformer block
- Replace the multi-head self attention(MSA) module with shifted window based MSA

Window based self attention

If an image is (h x w) patches and a window is (M x M) patches



Computational complexity (naively):
Global:

(h*w)^2 (quadratic)

Window based:

(M*M)^2 * (h/M * w/M)
(M*M) * (h*w) (Linear)

- Limiting self attention computation within local windows can reduce the computational complexity

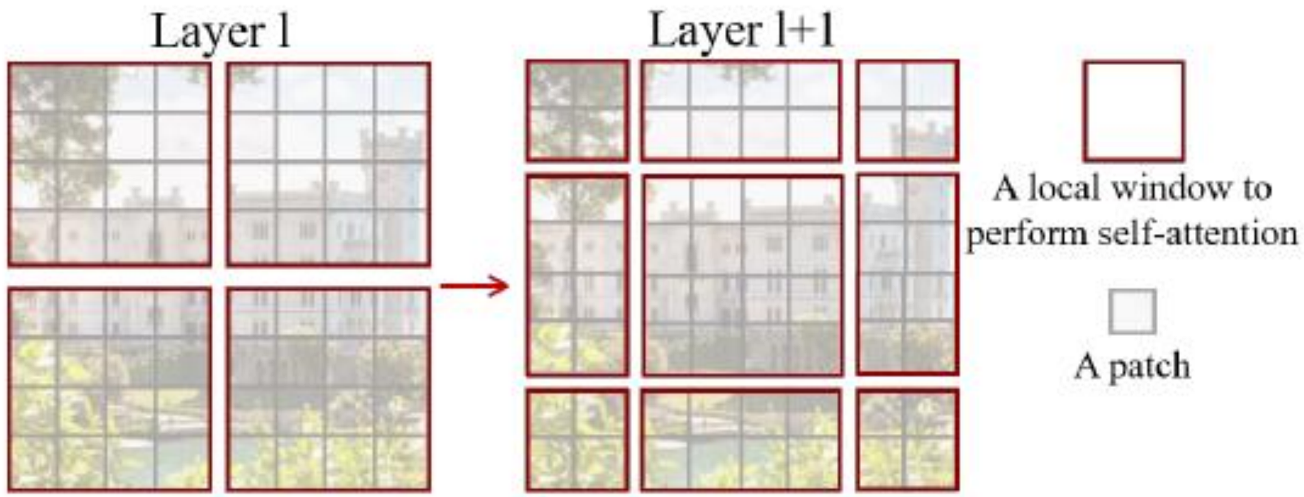
Window based self attention

Computational complexity (from paper):

$$\Omega(\text{MSA}) = 4hwC^2 + 2(hw)^2C,$$
$$\Omega(\text{W-MSA}) = 4hwC^2 + 2M^2hwC,$$

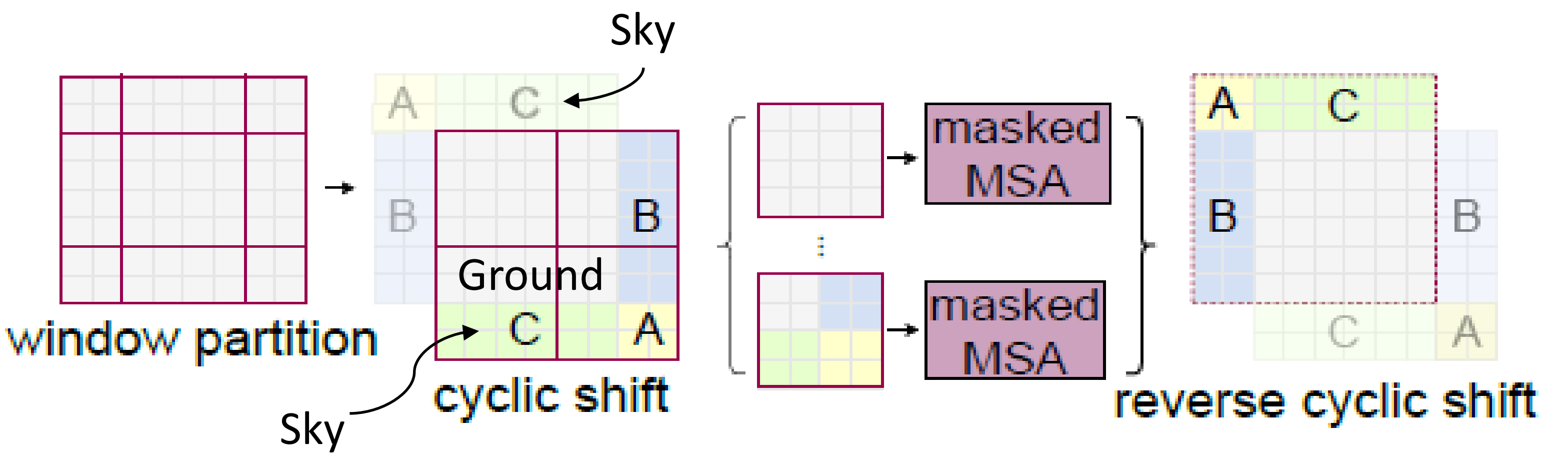
- Reduce to linear computational complexity with respect to image size
- Locality: related objects are usually in the neighborhood area for computer vision
- But still lacks connections across windows => shifted windows

Shifted window



- Shifted bottom right by $(\lfloor\frac{M}{2}\rfloor, \lfloor\frac{M}{2}\rfloor)$ pixels
- Introduces connections between neighboring non-overlapping windows

Some tricky adjustments(skipped)



- After shifting the windows, we have more windows and they are not in the same size

$\Rightarrow$ Can not process in parallel and the computational complexity also increase

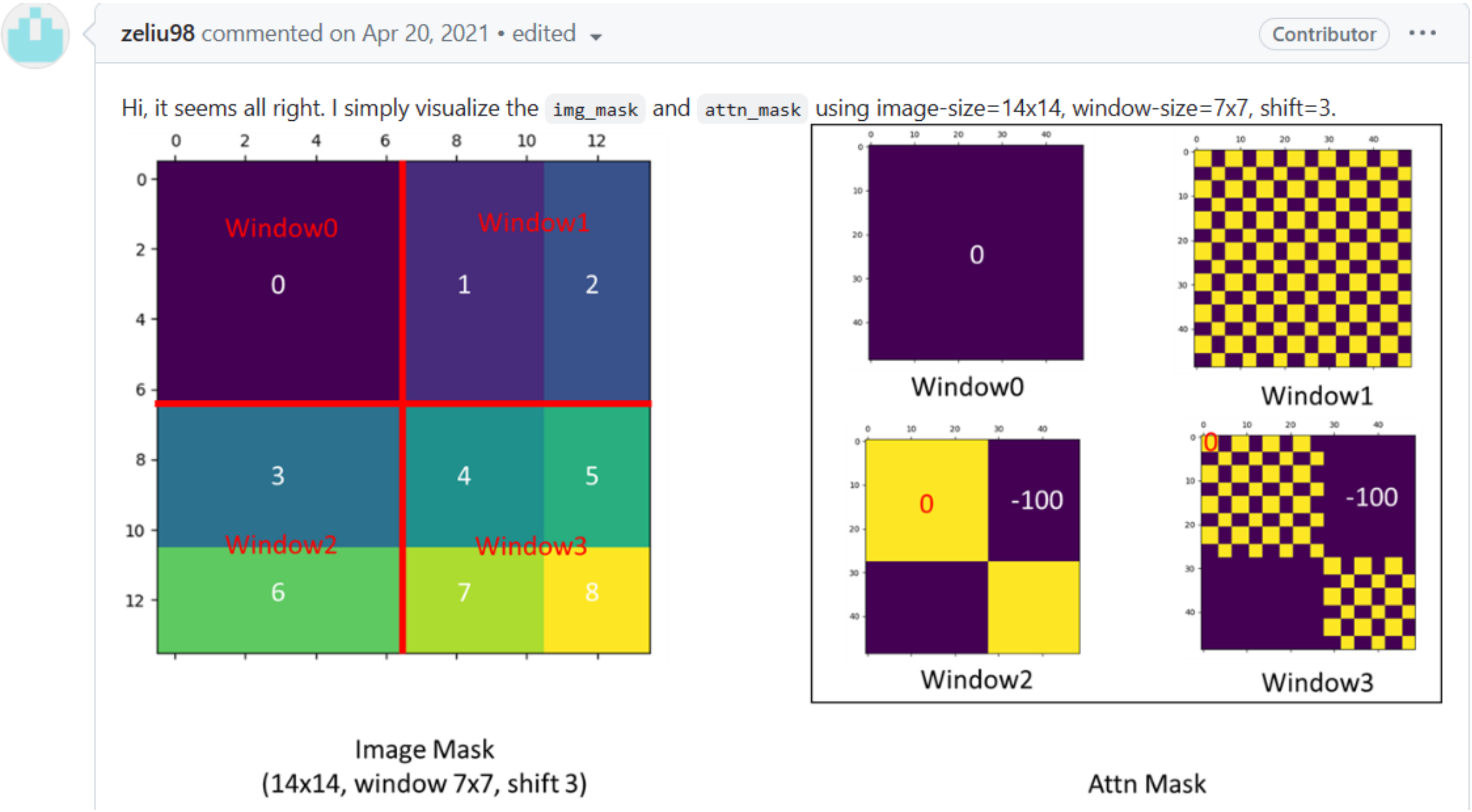
- Do cyclic shift! We have the same number of windows and all of them are in the same size.

- However, we should not compute the self-attention of sub-windows which are not adjacent in the original image

- Masking mechanism limits self-attention computation to within each sub-window

Some tricky adjustments(skipped)

Author's explanation of masked MSA:



The Question about the mask of window attention · Issue #38 · microsoft/Swin-Transformer · GitHub

Experiments

Image Classification on ImageNet1K

- Regular ImageNet-1K training
- With similar complexities, the accuracy: Swin Transformers > DeiT
- Pre-trained on ImageNet-22K & fine tuned on ImageNet-1K
- With similar complexities, the accuracy: Swin Transformers > ViT

(a) Regular ImageNet-1K trained models

method	image size	#param.	FLOPs	throughput (image / s)	ImageNet top-1 acc.
RegNetY-4G [44]	224^2	21M	4.0G	1156.7	80.0
RegNetY-8G [44]	224^2	39M	8.0G	591.6	81.7
RegNetY-16G [44]	224^2	84M	16.0G	334.7	82.9
ViT-B/16 [19]	384^2	86M	55.4G	85.9	77.9
ViT-L/16 [19]	384^2	307M	190.7G	27.3	76.5
DeiT-S [57]	224^2	22M	4.6G	940.4	79.8
DeiT-B [57]	224^2	86M	17.5G	292.3	81.8
DeiT-B [57]	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 [34]	384^2	388M	204.6G	-	84.4
R-152x4 [34]	480^2	937M	840.5G	-	85.4
ViT-B/16 [19]	384^2	86M	55.4G	85.9	84.0
ViT-L/16 [19]	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

Object Detection on COCO

- ResNet-50 and Swin-T as the backbones of different methods

(a) Various frameworks

Method	Backbone	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	#param.	FLOPs	FPS
Cascade Mask R-CNN	R-50	46.3	64.3	50.5	82M	739G	18.0
	Swin-T	**50.5**	**69.3**	**54.9**	86M	745G	15.3
ATSS	R-50	43.5	61.9	47.0	32M	205G	28.3
	Swin-T	**47.2**	**66.5**	**51.3**	36M	215G	22.3
RepPointsV2	R-50	46.5	64.6	50.3	42M	274G	13.6
	Swin-T	**50.0**	**68.5**	**54.2**	45M	283G	12.0
Sparse R-CNN	R-50	44.5	63.4	48.2	106M	166G	21.0
	Swin-T	**47.9**	**67.3**	**52.3**	110M	172G	18.4

- ResNet-50, DeiT and Swin-T as the backbones using the same method
- The lower inference speed of DeiT is mainly due to its quadratic complexity to input image size

(b) Various backbones w. Cascade Mask R-CNN

	AP^{box}	AP^{box}_{50}	AP^{box}_{75}	AP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}	param	FLOPs	FPS
DeiT-S†	48.0	67.2	51.7	41.4	64.2	44.3	80M	889G	10.4
R50	46.3	64.3	50.5	40.1	61.7	43.4	82M	739G	18.0
Swin-T	**50.5**	**69.3**	**54.9**	**43.7**	**66.6**	**47.1**	86M	745G	15.3
X101-32	48.1	66.5	52.4	41.6	63.9	45.2	101M	819G	12.8
Swin-S	**51.8**	**70.4**	**56.3**	**44.7**	**67.9**	**48.5**	107M	838G	12.0
X101-64	48.3	66.4	52.3	41.7	64.0	45.1	140M	972G	10.4
Swin-B	**51.9**	**70.9**	**56.5**	**45.0**	**68.4**	**48.7**	145M	982G	11.6

Semantic Segmentation on ADE20K

- With similar computation cost, the mIoU:
  Swin-S > DeiT-S
  Swin-S > ResNet-101

ADE20K Method	Backbone	val mIoU	test score	#param.	FLOPs	FPS
DLab.v3+ [11]	ResNet-101	44.1	-	63M	1021G	16.0
DNL [65]	ResNet-101	46.0	56.2	69M	1249G	14.8
OCRNet [67]	ResNet-101	45.3	56.0	56M	923G	19.3
UperNet [63]	ResNet-101	44.9	-	86M	1029G	20.1
OCRNet [67]	HRNet-w48	45.7	-	71M	664G	12.5
DLab.v3+ [11]	ResNeSt-101	46.9	55.1	66M	1051G	11.9
DLab.v3+ [11]	ResNeSt-200	48.4	-	88M	1381G	8.1
SETR [73]	T-Large‡	50.3	61.7	308M	-	-
UperNet	DeiT-S†	44.0	-	52M	1099G	16.2
UperNet	Swin-T	46.1	-	60M	945G	18.5
UperNet	Swin-S	49.3	-	81M	1038G	15.2
UperNet	Swin-B‡	51.6	-	121M	1841G	8.7
UperNet	Swin-L‡	**53.5**	**62.8**	234M	3230G	6.2

Effectiveness of Shifted window

- Accuracy of shifted window

	ImageNet		COCO		ADE20k
	top-1	top-5	AP^{box}	AP^{mask}	mIoU
w/o shifting	80.2	95.1	47.7	41.5	43.3
shifted windows	**81.3**	**95.6**	**50.5**	**43.7**	**46.1**

- Real speed of different self-attention computation

- Window(w/o shifting) > shifted window > sliding window

method	MSA in a stage (ms)				Arch. (FPS)		
	S1	S2	S3	S4	T	S	B
sliding window (naive)	122.5	38.3	12.1	7.6	183	109	77
sliding window (kernel)	7.6	4.7	2.7	1.8	488	283	187
Performer [14]	4.8	2.8	1.8	1.5	638	370	241
window (w/o shifting)	2.8	1.7	1.2	0.9	770	444	280
shifted window (padding)	3.3	2.3	1.9	2.2	670	371	236
shifted window (cyclic)	3.0	1.9	1.3	1.0	755	437	278

Conclusion

- Swin Transformer produces a hierarchical feature representation and has linear computational complexity with respect to input image size.

- Strong performance on various vision problems will encourage unified modeling of vision and language signals