CvT: Introducing Convolutions to Vision Transformers

School of Industrial and Management Engineering, Korea University

Lee Kyung Yoo

KOREA UNIVERSITY

DMQA

Contents

Introduction

❖ CvT: Introducing Convolutions to Vision Transformers (arXiv, 2021)

- 저자들은 McGill University 및 Microsoft Cloud + AI 소속
- 2021년 08월 25일 기준으로 57회 인용

CvT: Introducing Convolutions to Vision Transformers

Haiping Wu[1,2*] Bin Xiao[2†] Noel Codella[2] Mengchen Liu[2] Xiyang Dai[2]
Lu Yuan[2] Lei Zhang[2]
[1]McGill University [2]Microsoft Cloud + AI
haiping.wu2@mail.mcgill.ca, {bixi, ncodella, mengcliu, xidai, luyuan, leizhang}@microsoft.com

Abstract

We present in this paper a new architecture, named Convolutional vision Transformer (CvT), that improves Vision Transformer (ViT) in performance and efficiency by introducing convolutions into ViT to yield the best of both designs. This is accomplished through two primary modifications: a hierarchy of Transformers containing a new convolutional token embedding, and a convolutional Transformer block leveraging a convolutional projection. These changes introduce desirable properties of convolutional neural networks (CNNs) to the ViT architecture (i.e. shift, scale, and distortion invariance) while maintaining the merits of Transformers (i.e. dynamic attention, global context, and better generalization). We validate CvT by conducting extensive experiments, showing that this approach achieves state-of-the-art performance over other Vision Transformers and ResNets on ImageNet-1k, with fewer parameters and lower FLOPs. In addition, performance gains are maintained when pretrained on larger datasets (e.g. ImageNet-22k) and fine-tuned to downstream tasks. Pre-trained on ImageNet-22k, our CvT-W24 obtains a top-1 accuracy of 87.7% on the ImageNet-1k val set. Finally, our results show that the positional encoding, a crucial component in existing Vision Transformers, can be safely removed in our model, simplifying the design for higher resolution vision tasks. Code will be released at https://github.com/leoxiaobin/CvT.

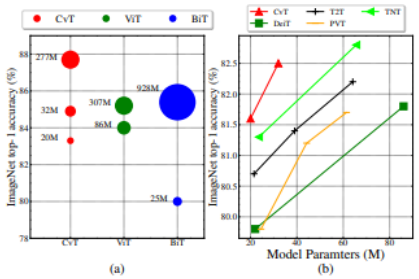


Figure 1: Top-1 Accuracy on ImageNet validation compared to other methods with respect to model parameters. (a) Comparison to CNN-based model BiT [18] and Transformer-based model ViT [11], when pretrained on ImageNet-22k. Larger marker size indicates larger architectures. (b) Comparison to concurrent works: DeiT [30], T2T [41], PVT [34], TNT [14] when pretrained on ImageNet-1k.

architectures [10] from language understanding with minimal modifications. First, images are split into discrete non-overlapping patches (*e.g.* 16×16). Then, these patches are treated as tokens (analogous to tokens in NLP), summed

DMQA

Introduction

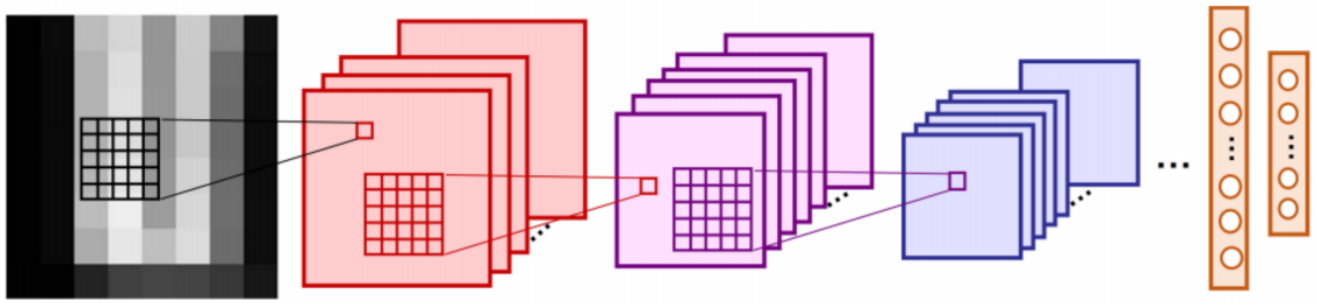
❖ CvT: Introducing Convolutions to Vision Transformers (arXiv, 2021)

- Convolutional Neural Network(CNN)와 Vision Transformer(ViT)를 결합하여, 이미지 분류 문제에서의 새로운 모델 아키텍처 제안
- 기존 ViT 구조에 두 가지 convolution-based operations를 적용
  - ➢ **Convolutional Token Embedding**
  - ➢ **Convolutional Projection for Attention**
- 이를 통해 CNN의 장점(local receptive fields, shared weights, and spatial subsampling)과 ViT의 장점(dynamic attention, global context fusion, and better generalization)을 모두 유지하면서 높은 계산 효율성 및 우수한 성능을 달성

Research Purpose

❖ Vision task에 적합한 구조를 지닌 CNN

- ViT는 막대한 스케일에서의 우수한 성능에도 불구하고, 보다 소량의 데이터에 대해 훈련 시 비슷한 크기의 CNN 모델(e.g., ResNet)보다 낮은 성능을 보임
- 이미지는 인접한 픽셀간 높은 연관성을 가지는 강력한 **2D local structure**로 이루어져 있으며, **CNN은 이에 본질적으로 적합한 속성을 보유한 구조**로 이루어져 있음
  - ➢ Local receptive fields / Shared weights / Spatial subsampling
  - ➢ Hierarchical structure of convolutional kernels



Chu, Jie, et al. "A Novel Bilinear Feature and Multi-Layer Fused Convolutional Neural Network for Tactile Shape Recognition." *Sensors* 20.20 (2020): 5822.

Research Purpose

❖ "이러한 CNN을 기존 ViT에 부분적으로 접목해보면?"

- **ViT 구조에 convolution 연산을 전략적으로 도입함**으로써, 높은 수준의 계산 및 메모리 효율성을 유지함과 동시에 모델의 성능과 강건함을 향상시킬 수 있다고 가정
- 이를 검증하기 위해 새로운 아키텍처인 **Convolutional Vision Transformer(CvT)**를 제안하였으며, parameter 수와 FLOPs 두 가지 측면 모두에서 효율성 확인

Method	Needs Position Encoding (PE)	Token Embedding	Projection for Attention	Hierarchical Transformers
ViT [11], DeiT [30]	yes	non-overlapping	linear	no
CPVT [6]	no (w/ PE Generator)	non-overlapping	linear	no
TNT [14]	yes	non-overlapping (patch+pixel)	linear	no
T2T [41]	yes	overlapping (concatenate)	linear	partial (tokenization)
PVT [34]	yes	non-overlapping	spatial reduction	yes
CvT (ours)	no	overlapping (convolution)	convolution	yes

Convolutional Vision Transformer

- Overview of CvT

❖ Architecture

- 기존 ViT에서 변형된 두 가지 구조
  - ➢ **Patch + Position embedding → Convolutional Token Embedding**
  - ➢ **Linear projection → Convolutional Projection for Attention**
- 이를 여러 단계에 걸쳐 수행함으로써, 계층적 구조 생성

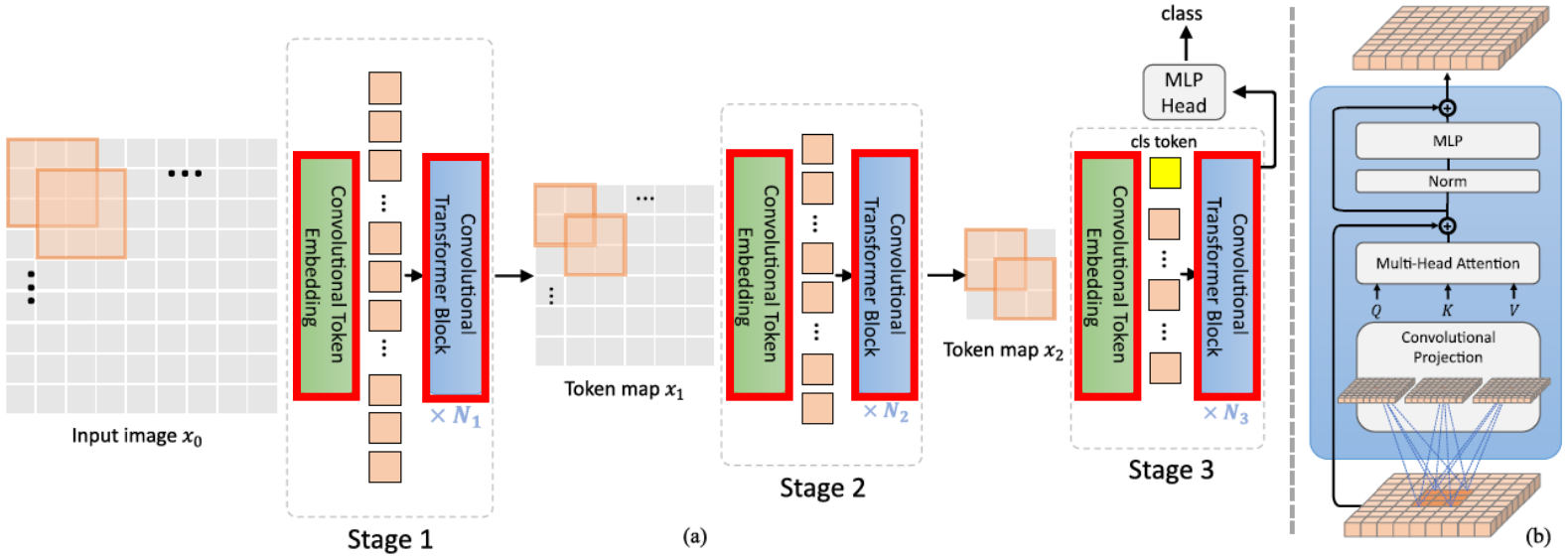


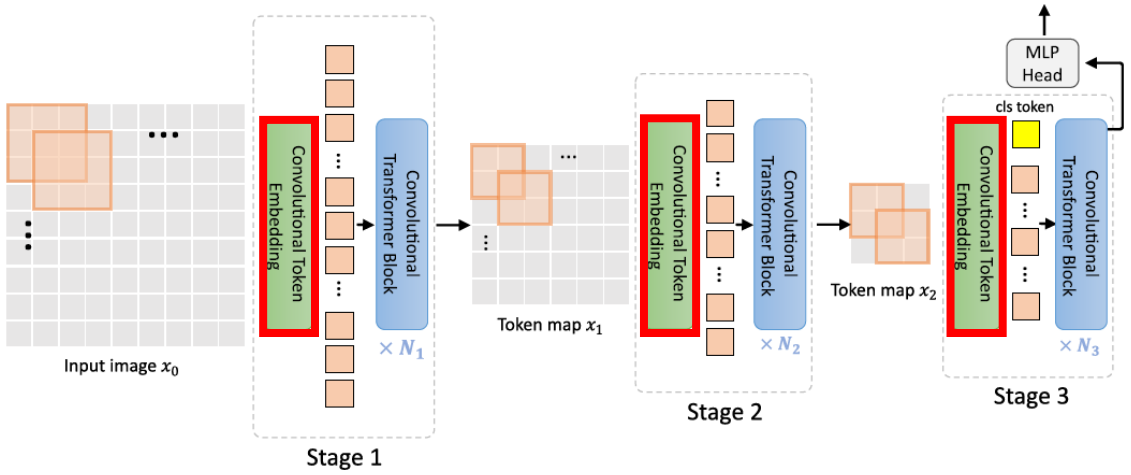
Figure 2: The pipeline of the proposed CvT architecture. (a) Overall architecture, showing the hierarchical multi-stage structure facilitated by the Convolutional Token Embedding layer. (b) Details of the Convolutional Transformer Block, which contains the convolution projection as the first layer.

Convolutional Vision Transformer

- Convolutional Token Embedding

❖ Architecture

- 입력 이미지, 즉 2D token maps를 단계적으로 Convolutional Token Embedding 레이어에 통과
- Convolution 연산 내 매개변수를 변경하여 각 단계에서의 token의 수(= feature resolution)와 너비(= feature dimension)를 조정 가능
- 단계적 convolution 연산으로 token의 수를 점진적으로 줄이는 동시에 token의 너비를 증가시켜, spatial undersampling과 complex visual pattern representation을 가능하게 함



$$x_{i-1} \in \mathbb{R}^{H_{i-1} \times W_{i-1} \times C_{i-1}}$$

= output token map from previous

$$f(x_{i-1}) \in \mathbb{R}^{H_i \times W_i \times C_i}$$

= maps x_{i-1} into new tokens

$$H_i = \left\lfloor \frac{H_{i-1} + 2p - s}{s - o} + 1 \right\rfloor$$

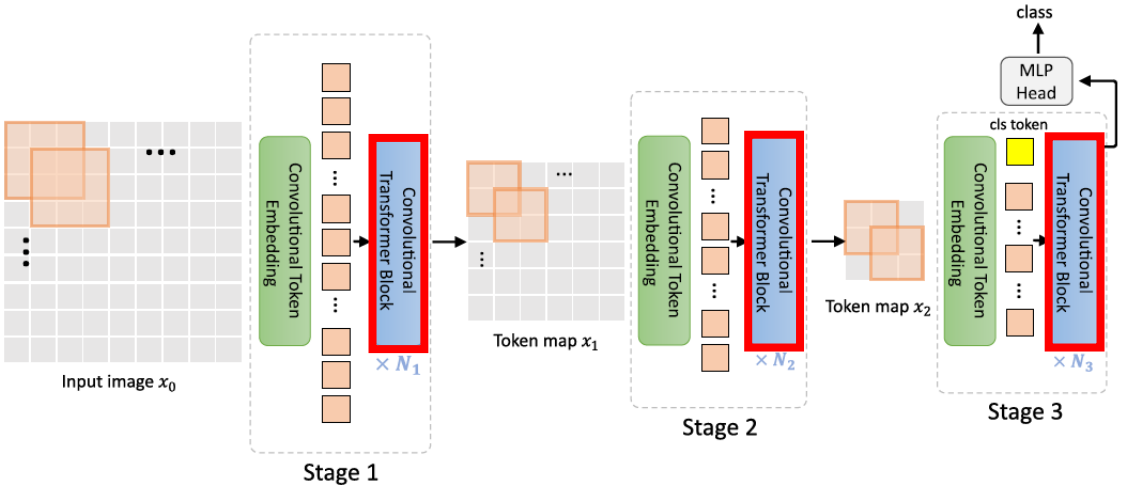
$$W_i = \left\lfloor \frac{W_{i-1} + 2p - s}{s - o} + 1 \right\rfloor$$

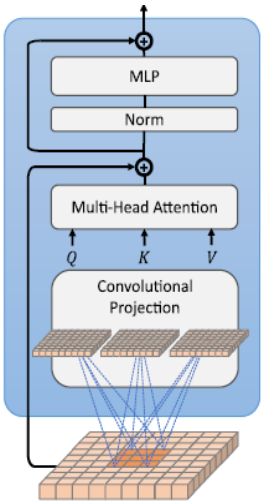
Convolutional Vision Transformer

- Convolutional Projection for Attention

❖ Architecture

- Convolutional Transformer Block의 첫번째 레이어에서 convolutional projection 수행
- 임베딩 값을 fully connected layer에 통과시켜 query, key, value로 변환하던 기존 과정을 depth-wise convolution 연산으로 대체
- Query에는 stride=1, key와 value에는 stride=2를 적용하여 연산량 감소(by spatial undersampling)
- Convolutional Token Embedding과 함께 local spatial context를 더 잘 모델링하게 함





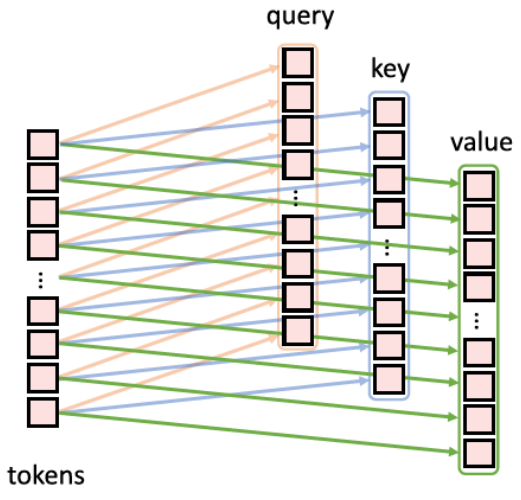
$$x_i^{q/k/v} = \texttt{Flatten}\left(\texttt{Conv2d}\left(\texttt{Reshape2D}(x_i),\ s\right)\right)$$

Convolutional Vision Transformer

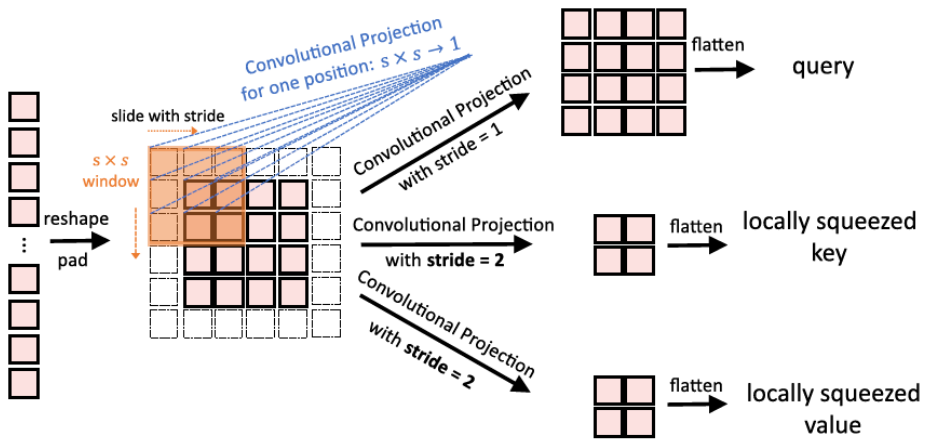
- Convolutional Projection for Attention

❖ Architecture

- Convolutional Transformer Block의 첫번째 레이어에서 convolutional projection 수행
- 임베딩 값을 fully connected layer에 통과시켜 query, key, value로 변환하던 기존 과정을 depth-wise convolution 연산으로 대체
- Query에는 stride=1, key와 value에는 stride=2를 적용하여 연산량 감소(by spatial undersampling)
- Convolutional Token Embedding과 함께 local spatial context를 더 잘 모델링하게 함



Linear projection



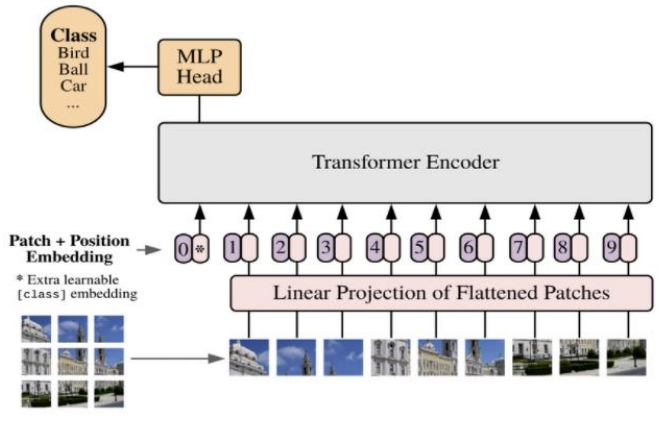
Convolution projection

Convolutional Vision Transformer

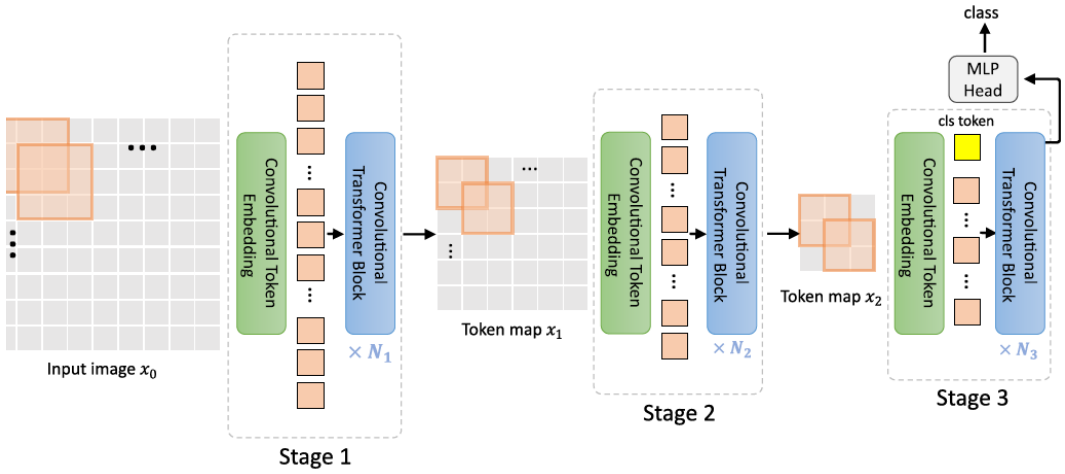
- Extra modifications

❖ Architecture

- 한 stage 내 Convolutional Token Embedding과 Convolutional Projection이 한 번씩 수행되며, 전체 프로세스 내 총 세 개의 stage로 구성
- Position embedding을 거치지 않음
- Classification token은 가장 마지막 stage에서 한 번만 추가됨



ViT architecture



CvT architecture

Experiments

❖ Architecture for model comparisons

- ImageNet classification을 위한 여러 크기(Params / FLOPs)의 모델 구성
- Conv. Embed. = Convolutional Token Embedding / Conv. Proj. = Convolutional Projection
- H_i: number of heads / D_i: number embedding feature dimension in the ith Multi-Head Self-Attention
- R_i: feature dimension expansion ratio in the ith MLP layer

	Output Size	Layer Name	CvT-13	CvT-21	CvT-W24
Stage1	56×56	Conv. Embed.	7×7, 64, stride 4		7×7, 192, stride 4
	56×56	Conv. Proj. MHSA MLP	$\begin{bmatrix} 3 \times 3, 64 \\ H_1 = 1, D_1 = 64 \\ R_1 = 4 \end{bmatrix} \times 1$	$\begin{bmatrix} 3 \times 3, 64 \\ H_1 = 1, D_1 = 64 \\ R_1 = 4 \end{bmatrix} \times 1$	$\begin{bmatrix} 3 \times 3, 192 \\ H_1 = 3, D_1 = 192 \\ R_1 = 4 \end{bmatrix} \times 2$
Stage2	28×28	Conv. Embed.	3×3, 192, stride 2		3×3, 768, stride 2
	28×28	Conv. Proj. MHSA MLP	$\begin{bmatrix} 3 \times 3, 192 \\ H_2 = 3, D_2 = 192 \\ R_2 = 4 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 192 \\ H_2 = 3, D_2 = 192 \\ R_2 = 4 \end{bmatrix} \times 4$	$\begin{bmatrix} 3 \times 3, 768 \\ H_2 = 12, D_2 = 768 \\ R_2 = 4 \end{bmatrix} \times 2$
Stage3	14×14	Conv. Embed.	3×3, 384, stride 2		3×3, 1024, stride 2
	14×14	Conv. Proj. MHSA MLP	$\begin{bmatrix} 3 \times 3, 384 \\ H_3 = 6, D_3 = 384 \\ R_3 = 4 \end{bmatrix} \times 10$	$\begin{bmatrix} 3 \times 3, 384 \\ H_3 = 6, D_3 = 384 \\ R_3 = 4 \end{bmatrix} \times 16$	$\begin{bmatrix} 3 \times 3, 1024 \\ H_3 = 16, D_3 = 1024 \\ R_3 = 4 \end{bmatrix} \times 20$
Head	1×1	Linear	1000		
Params			19.98 M	31.54 M	276.7 M
FLOPs			4.53 G	7.13 G	60.86 G

Experiments

❖ Default training settings (CvT)

- Dataset: ImageNet-1k, 22k
- Optimizer: AdamW
- Batch size / epochs: 2048 / 300
- Learning rate: 0.02
- Learning rate scheduler: cosine learning rate decay scheduler
- ViT와 동일한 data augmentation 및 regularization methods 사용하였으며, 모든 ImageNet model은 224×224 크기의 input으로 훈련됨

Experiments

❖ Results

- 실험 결과 기존 CNN 및 ViT 모델보다 더 적은 Parameter 수와 낮은 FLOPs로 우수한 성능 달성

Method Type	Network	#Param. (M)	image size	FLOPs (G)	ImageNet top-1 (%)	Real top-1 (%)	V2 top-1 (%)
Convolutional Networks	ResNet-50 [15]	25	224^2	4.1	76.2	82.5	63.3
	ResNet-101 [15]	45	224^2	7.9	77.4	83.7	65.7
	ResNet-152 [15]	60	224^2	11	78.3	84.1	67.0
Transformers	ViT-B/16 [11]	86	384^2	55.5	77.9	83.6	–
	ViT-L/16 [11]	307	384^2	191.1	76.5	82.2	–
	DeiT-S [30][arxiv 2020]	22	224^2	4.6	79.8	85.7	68.5
	DeiT-B [30][arxiv 2020]	86	224^2	17.6	81.8	86.7	71.5
	PVT-Small [34][arxiv 2021]	25	224^2	3.8	79.8	–	–
	PVT-Medium [34][arxiv 2021]	44	224^2	6.7	81.2	–	–
	PVT-Large [34][arxiv 2021]	61	224^2	9.8	81.7	–	–
	T2T-ViT$_t$-14 [41][arxiv 2021]	22	224^2	6.1	80.7	–	–
	T2T-ViT$_t$-19 [41][arxiv 2021]	39	224^2	9.8	81.4	–	–
	T2T-ViT$_t$-24 [41][arxiv 2021]	64	224^2	15.0	82.2	–	–
	TNT-S [14][arxiv 2021]	24	224^2	5.2	81.3	–	–
	TNT-B [14][arxiv 2021]	66	224^2	14.1	82.8	–	–
Convolutional Transformers	**Ours:** CvT-13	20	224^2	4.5	81.6	86.7	70.4
	Ours: CvT-21	32	224^2	7.1	82.5	87.2	71.3
	Ours: CvT-13$_{\uparrow 384}$	20	384^2	16.3	83.0	87.9	71.9
	Ours: CvT-21$_{\uparrow 384}$	32	384^2	24.9	**83.3**	**87.7**	**71.9**
	Ours: CvT-13-NAS	18	224^2	4.1	82.2	87.5	71.3

Experiments

❖ Results

- 더 큰 데이터셋(ImageNet-22k)에서 pre-trained된 CvT-W24가 가장 좋은 성능을 보임

Method Type	Network	#Param. (M)	image size	FLOPs (G)	ImageNet top-1 (%)	Real top-1 (%)	V2 top-1 (%)
Convolution Networks$_{22k}$	BiT-M$_{\uparrow 480}$ [18]	928	480^2	837	85.4	–	–
Transformers$_{22k}$	ViT-B/16$_{\uparrow 384}$ [11]	86	384^2	55.5	84.0	88.4	–
	ViT-L/16$_{\uparrow 384}$ [11]	307	384^2	191.1	85.2	88.4	–
	ViT-H/16$_{\uparrow 384}$ [11]	632	384^2	–	85.1	88.7	–
Convolutional Transformers$_{22k}$	**Ours:** CvT-13$_{\uparrow 384}$	20	384^2	16	83.3	88.7	72.9
	Ours: CvT-21$_{\uparrow 384}$	32	384^2	25	84.9	89.8	75.6
	Ours: CvT-W24$_{\uparrow 384}$	277	384^2	193.2	**87.7**	**90.6**	**78.8**

Experiments

❖ Removing the position embedding

- 모델에 convolution 구조를 도입하여 local context를 잡아낼 수 있음에 따라 position embedding이 여전히 필요한지 실험 진행
- 실험 결과, 모델에서 position embedding을 제거해도 성능이 저하되지 않음을 확인
- 따라서 CvT에서 position embedding을 완전히 제거할 수 있음에 따라, embedding을 다시 설계할 필요 없이 vision task에 대한 보다 단순화된 형태로 접근이 가능함

Method	Model	Param (M)	Pos. Emb.	ImageNet Top-1 (%)
a	DeiT-S	22	Default	79.8
b	DeiT-S	22	N/A	78.0
c	CvT-13	20	Every stage	81.5
d	CvT-13	20	First stage	81.4
e	CvT-13	20	Last stage	81.4
f	CvT-13	20	N/A	81.6

Table 5: Ablations on position embedding.

Conclusion

❖ Conclusion

- CNN은 이에 본질적으로 2D local structure인 이미지에 적합한 속성을 보유한 구조로, 이미지 분류 문제에 있어 우수한 성능이 보장되어 있음
- 기존 ViT 구조에 CNN을 결합하여(Convolutional Token Embedding / Convolutional Projection), 이미지 분류를 위한 local & global dependencies를 효율적으로 모델링하는 방법을 제안
- 계층적 구조가 적용됨과 함께, position embedding이 필요 없는 간단한 구조로 변환

❖ Opinion

- CNN과 Transformer를 연결하려는 다양한 시도가 많았음
- 그러한 시도들과 해당 논문이 어떠한 차별점을 가지는지에 초점을 맞춰 흐름을 정리해나가는 것도 좋은 시도라고 생각됨

References

- Wu, Haiping, et al. "Cvt: Introducing convolutions to vision transformers." arXiv preprint arXiv:2103.15808 (2021).
- Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).
- Chu, Jie, et al. "A Novel Bilinear Feature and Multi-Layer Fused Convolutional Neural Network for Tactile Shape Recognition." Sensors 20.20 (2020): 5822.


Thank You