CoAtNet: Marrying Convolution and Attention for All Data Sizes

Zihang Dai, Hanxiao Liu, Quoc V. Le, Mingxing Tan || Google Research, Brain Team (2021.06)
link : https://arxiv.org/pdf/2106.04803.pdf

Gyujun Jeong

2022/01/25
email : jgyujun911@gmail.com
main subject : The aim of CoAtNet is therefore to blend the pros of CNNs and Transformers into a single architecture, in terms of 1) generalization and 2) model capacity.

Positional Encoding : Spatial-relative attention

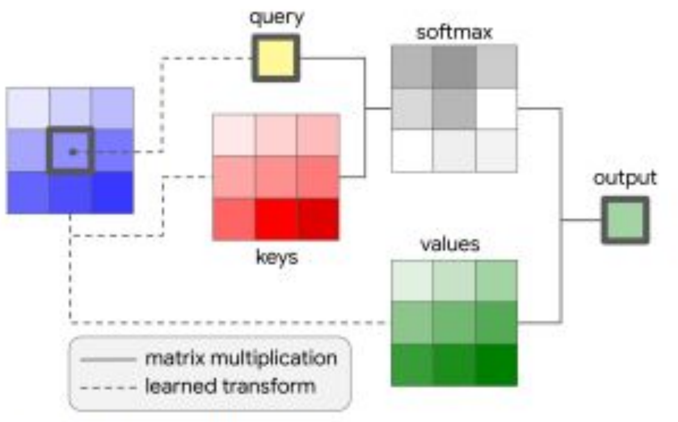


Figure 3: An example of a local attention layer over spatial extent of $k = 3$.

Figure 4: An example of relative distance computation. The relative distances are computed with respect to the position of the highlighted pixel. The format of distances is *row offset*, *column offset*.

Stand-Alone Self-Attention in Vision Models
Parmar, Niki, et al || Google Research, Brain Team(NIPS, 2019)

Contents

Introduction

- None of these ViT variants could outperform the SOTA convolution-only models on ImageNet classification given the same amount of data and computation.
- Many recent works have been trying to incorporate the inductive biases of ConvNets into Transformer models, by imposing local receptive fields for attention layers or augmenting the attention and FFN layers with implicit or explicit convolutional operations.
- In this work, we systematically study the problem of hybridizing convolution and attention from two fundamental aspects in machine learning generalization and model capacity.
- In this paper, we investigate two key insights:

  1) we observe that the commonly used depthwise convolution can be effectively merged into attention layers with simple relative attention

  2) simply stacking convolutional and attention layers, in a proper way, could be surprisingly effective to achieve better generalization and capacity.

Model

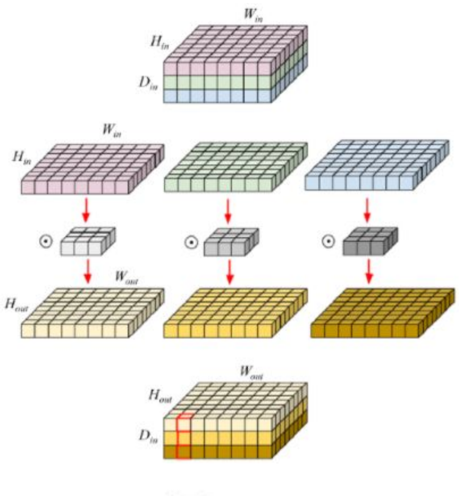
- we focus on the question of how to “optimally” combine the convolution and transformer. Roughly speaking, we decompose the question into two parts:

1. How to combine the convolution and self-attention within one basic computational block?
2. How to vertically stack different types of computational blocks together to form a complete network?

Model : Merging convolution and Self-attention

1. How to combine the convolution and self-attention within one basic computational block?

Depthwise Convolution



$$y_i = \sum_{j \in \mathcal{L}(i)} w_{i-j} \odot x_j$$

$x_i, y_i \in \mathbb{R}^D$ are the input and output at position i respectively,

Self-Attention

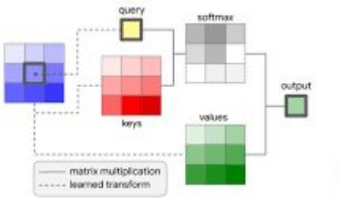


Figure 3: An example of a local attention layer over spatial extent of $k = 3$.

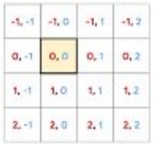


Figure 4: An example of relative distance computation. The relative distances are computed with respect to the position of the highlighted pixel. The format of distances is *row offset*, *column offset*.

$$y_i = \sum_{j \in \mathcal{G}} \underbrace{\frac{\exp\left(x_i^\top x_j\right)}{\sum_{k \in \mathcal{G}} \exp\left(x_i^\top x_k\right)}}_{A_{i,j}} x_j$$

$\mathcal{G}$ indicates the global spatial space.

Table 1: Desirable properties found in convolution or self-attention.

Properties	Convolution	Self-Attention
Translation Equivariance	✓	
Input-adaptive Weighting		✓
Global Receptive Field		✓

$$y_i^{\text{post}} = \sum_{j \in \mathcal{G}} \left(\frac{\exp\left(x_i^\top x_j\right)}{\sum_{k \in \mathcal{G}} \exp\left(x_i^\top x_k\right)} + w_{i-j} \right) x_j$$

relative self-attention

$$y_i^{\text{pre}} = \sum_{j \in \mathcal{G}} \frac{\exp\left(x_i^\top x_j + w_{i-j}\right)}{\sum_{k \in \mathcal{G}} \exp\left(x_i^\top x_k + w_{i-k}\right)} x_j .$$

Model : Vertical Layout Design

However, If we directly apply the relative attention. it to the raw image input, the computation will be excessively slow due to the large number of pixels in any image of common sizes

2. How to vertically stack different types of computational blocks together to form a complete network?

(A) Perform some down-sampling to reduce the spatial size and employ the global relative attention after the feature map reaches manageable level.

(B) Enforce local attention, which restricts the global receptive field G in attention to a local field L just like in convolution.

(C) Replace the quadratic Softmax attention with certain linear attention variant which only has a linear complexity w.r.t. the spatial size.

Model : CoAtNet

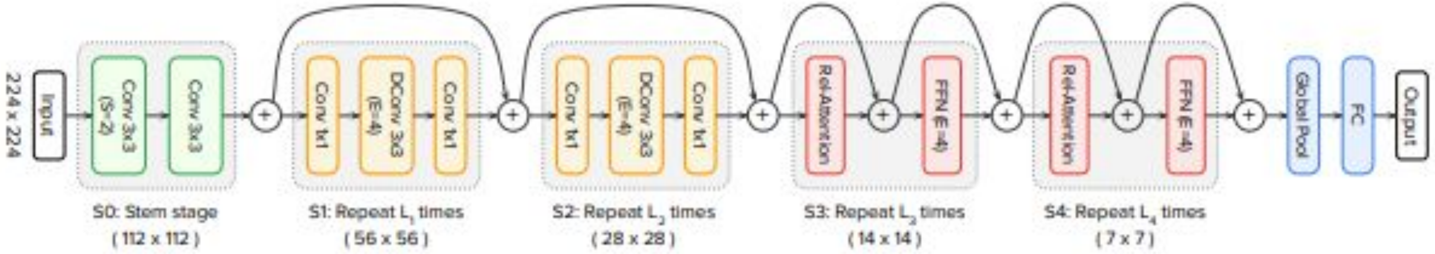


Figure 4: Overview of the proposed CoAtNet.

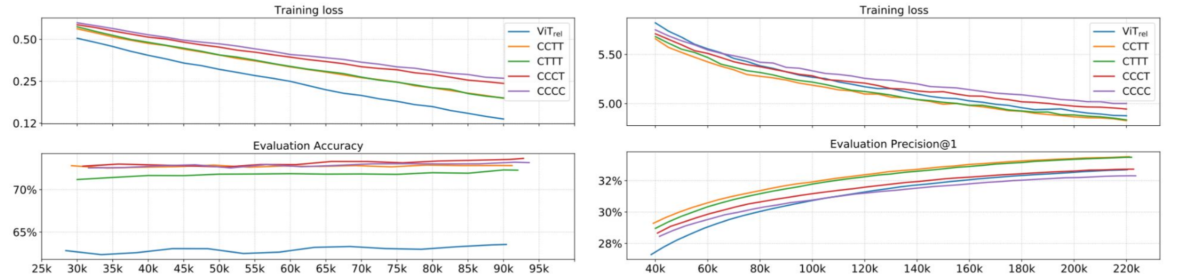
Table 3: L denotes the number of blocks and D denotes the hidden dimension (#channels). For all Conv and MBConv blocks, we always use the kernel size 3. For all Transformer blocks, we set the size of each attention head to 32, following [22]. The expansion rate for the inverted bottleneck is always 4 and the expansion (shrink) rate for the SE is always 0.25.

Stages	Size	CoAtNet-0	CoAtNet-1	CoAtNet-2	CoAtNet-3	CoAtNet-4
S0-Conv	1/2	L=2 D=64	L=2 D=64	L=2 D=128	L=2 D=192	L=2 D=192
S1-MbConv	1/4	L=2 D=96	L=2 D=96	L=2 D=128	L=2 D=192	L=2 D=192
S2-MBConv	1/8	L=3 D=192	L=6 D=192	L=6 D=256	L=6 D=384	L=12 D=384
S3-TFM_{Rel}	1/16	L=5 D=384	L=14 D=384	L=14 D=512	L=14 D=768	L=28 D=768
S4-TFM_{Rel}	1/32	L=2 D=768	L=2 D=768	L=2 D=1024	L=2 D=1536	L=2 D=1536

Table 11: CoAtNet-5 model sizes.

Stages	Size	CoAtNet-5
S0-Conv	1/2	L=2 D=192
S1-MbConv	1/4	L=2 D=256
S2-MBConv	1/8	L=12 D=512
S3-TFM_{Rel}	1/16	L=28 D=1280
S4-TFM_{Rel}	1/32	L=2 D=2048

Model : Systematically study the design choices



(a) ImageNet-1K (b) JFT

Figure 1: Comparison for model generalization and capacity under different data size. For fair comparison, all models have similar parameter size and computational cost.

Table 2: Transferability test results.

Metric	C-C-T-T	C-T-T-T
Pre-training Precision@1 (JFT)	34.40	34.36
Transfer Accuracy 224x224	**82.39**	81.78
Transfer Accuracy 384x384	**84.23**	84.02

- From the ImageNet-1K results, a key observation is that, in terms of generalization capability.

$$\text{C-C-C-C} \approx \text{C-C-C-T} \geq \text{C-C-T-T} > \text{C-T-T-T} \gg \text{ViT}_{\text{REL}}.$$

- As for model capacity, from the JFT comparison, both the train and evaluation metrics at the end of the training suggest the following ranking:

$$\text{C-C-T-T} \approx \text{C-T-T-T} > \text{ViT}_{\text{REL}} > \text{C-C-C-T} > \text{C-C-C-C}.$$

Experiments : Setting

- Evaluation Protocol
  - ImageNet-1K(1.28M images), ImageNet-21K(12.7M images) and JFT(300M images)
  - we first pre-train our models on each of the three datasets at resolution 224 for 300, 90, and 14 epochs respectively.
  - we finetune the pre-trained models on ImageNet-1K at the desired resolutions for 30 epochs and obtain the corresponding evaluation accuracy.
- Data Augmentation & Regularization
  - data augmentations - randaugment, mixup
  - common techniques - stochastic depth, label smoothing and weight decay
  - As a result, for certain runs of the proposed model, we deliberately apply RandAugment and stochastic depth of a small degree when pre-training on the two larger datasets, ImageNet21-K and JFT.

Experiments : Main Result

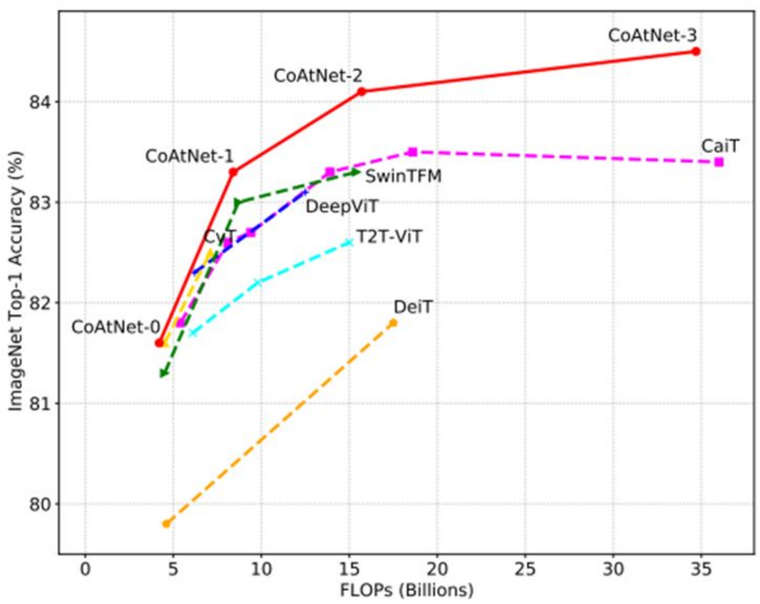


Figure 2: Accuracy-to-FLOPs scaling curve under ImageNet-1K only setting at 224x224.

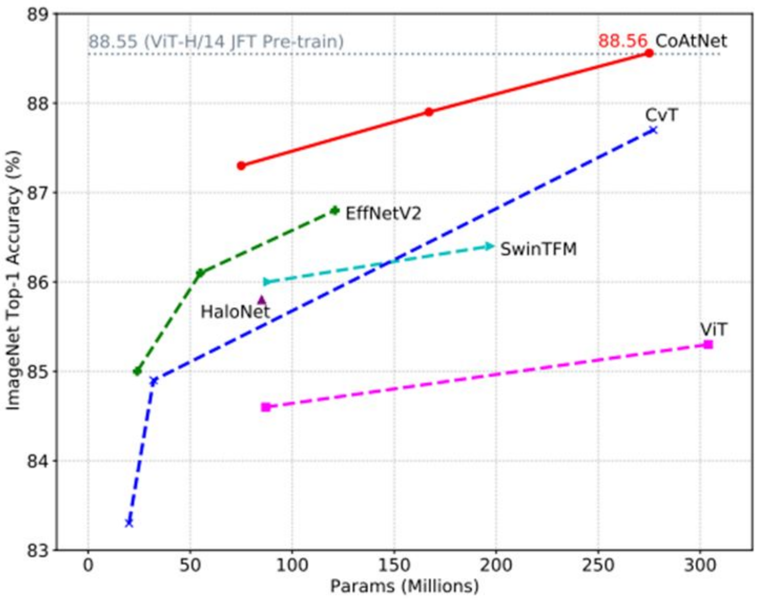


Figure 3: Accuracy-to-Params scaling curve under ImageNet-21K $\Rightarrow$ ImageNet-1K setting.

Main Result

Table 5: Performance Comparison on large-scale JFT dataset. `TPUv3-core-days` denotes the pre-training time, *Top-1 Accuracy* denotes the finetuned accuracy on ImageNet. Note that the last 3 rows use a larger dataset JFT-3B [26] for pre-training, while others use JFT-300M [15]. See Appendix A.2 for the size details of CoAtNet-5/6/7. †: Down-sampling in the MBConv block is achieved by stride-2 Depthwise Convolution. ◇: ViT-G/14 computation consumption is read from Fig. 1 of the paper [26].

Models	Eval Size	#Params	#FLOPs	TPUv3-core-days	Top-1 Accuracy
ResNet + ViT-L/16	384^2	330M	-	-	87.12
ViT-L/16	512^2	307M	364B	0.68K	87.76
ViT-H/14	518^2	632M	1021B	2.5K	88.55
NFNet-F4+	512^2	527M	367B	1.86K	89.2
CoAtNet-3†	384^2	168M	114B	0.58K	88.52
CoAtNet-3†	512^2	168M	214B	0.58K	88.81
CoAtNet-4	512^2	275M	361B	0.95K	89.11
CoAtNet-5	512^2	688M	812B	1.82K	89.77
ViT-G/14	518^2	1.84B	5160B	>30K◇	90.45
CoAtNet-6	512^2	1.47B	1521B	6.6K	90.45
CoAtNet-7	512^2	2.44B	2586B	20.1K	**90.88**

Table 4: Model performance on ImageNet. `1K only` denotes training on ImageNet-1K only; `21K+1K` denotes pre-training on ImageNet-21K and finetuning on ImageNet-1K; `PT-RA` denotes applying RandAugment during 21K pre-training, and `E150` means 150 epochs of 21K pre-training, which is longer than the standard 90 epochs. More results are in Appendix A.3.

Models		Eval Size	#Params	#FLOPs	ImageNet Top-1 Accuracy	
					1K only	21K+1K
Conv Only	EfficientNet-B7	600^2	66M	37B	84.7	-
	EfficientNetV2-L	480^2	121M	53B	85.7	86.8
	NFNet-F3	416^2	255M	114.8B	85.7	-
	NFNet-F5	544^2	377M	289.8B	**86.0**	-
ViT-Stem TFM	DeiT-B	384^2	86M	55.4B	83.1	-
	ViT-L/16	384^2	304M	190.7B	-	85.3
	CaiT-S-36	384^2	68M	48.0B	85.0	-
	DeepViT-L	224^2	55M	12.5B	83.1	-
Multi-stage TFM	Swin-B	384^2	88M	47.0B	84.2	86.0
	Swin-L	384^2	197M	103.9B	-	86.4
Conv+TFM	BotNet-T7	384^2	75.1M	45.8B	84.7	-
	LambdaResNet-420	320^2	-	-	84.8	-
	T2T-ViT-24	224^2	64.1M	15.0B	82.6	-
	CvT-21	384^2	32M	24.9B	83.3	-
	CvT-W24	384^2	277M	193.2B	-	**87.7**
Conv+TFM (ours)	CoAtNet-0	224^2	25M	4.2B	81.6	-
	CoAtNet-1	224^2	42M	8.4B	83.3	-
	CoAtNet-2	224^2	75M	15.7B	84.1	87.1
	CoAtNet-3	224^2	168M	34.7B	84.5	87.6
	CoAtNet-0	384^2	25M	13.4B	83.9	-
	CoAtNet-1	384^2	42M	27.4B	85.1	-
	CoAtNet-2	384^2	75M	49.8B	85.7	87.1
	CoAtNet-3	384^2	168M	107.4B	85.8	87.6
	CoAtNet-4	384^2	275M	189.5B	-	87.9
	+ PT-RA	384^2	275M	189.5B	-	88.3
	+ PT-RA-E150	384^2	275M	189.5B	-	88.4
	CoAtNet-2	512^2	75M	96.7B	85.9	87.3
	CoAtNet-3	512^2	168M	203.1B	**86.0**	87.9
	CoAtNet-4	512^2	275M	360.9B	-	88.1
	+ PT-RA	512^2	275M	360.9B	-	88.4
	+ PT-RA-E150	512^2	275M	360.9B	-	**88.56**

Conclusion

- In this paper, we systematically study the properties of convolutions and Transformers

- Extensive experiments show that CoAtNet enjoys both good generalization like ConvNets and superior model capacity like Transformers, achieving state-of-the-art performances under different data sizes and computation budgets.

- Note that this paper currently focuses on ImageNet classification for model development. However, we believe our approach is applicable to broader applications like object detection and semantic segmentation.

감사합니다.

Related Work

- Convolutional network building blocks.
- Self-attention and Transformers.
- Relative attention.
- Combining convolution and self-attention.

Ablation Studies

Table 6: Ablation on relative attention.

Setting	Metric	With Rel-Attn	Without Rel-Attn
ImageNet-1K	Accuracy (224^2)	84.1	83.8
	Accuracy (384^2)	85.7	85.3
ImageNet-21K	Pre-train Precision@1 (224^2)	53.0	52.8
$\Rightarrow$ ImageNet-1K	Finetune Accuracy (384^2)	87.9	87.4

Table 7: Ablation on architecture layout.

Setting	Models	Layout	Top-1 Accuracy
ImageNet-1K	V0: CoAtNet-2	[2, 2, 6, 14, 2]	84.1
	V1: S2 $\Leftarrow$ S3	[2, 2, 2, 18, 2]	83.4
	V2: S2 $\Rightarrow$ S3	[2, 2, 8, 12, 2]	84.0
ImageNet-21K	V0: CoAtNet-3	[2, 2, 6, 14, 2]	53.0 $\rightarrow$ 87.6
$\Rightarrow$ ImageNet-1K	V1: S2 $\Leftarrow$ S3	[2, 2, 2, 18, 2]	53.0 $\rightarrow$ 87.4

Table 8: Ablation on head size and normalization type.

Setting	Models	Image Size	Top-1 Accuracy
ImageNet-1K	CoAtNet-2	224^2	84.1
	Head size: 32 $\rightarrow$ 64	224^2	83.9
	Norm type: BN $\rightarrow$ LN	224^2	84.1
ImageNet-21K	CoAtNet-3	384^2	87.9
$\Rightarrow$ ImageNet-1K	Norm type: BN $\rightarrow$ LN	384^2	87.8

State-of-the-art : Image Classification on ImageNet